Semantic impact - a novel approach for domain concept selection in ontology learning

Wan, Jizheng

SEMANTIC IMPACT – A NOVEL APPROACH FOR DOMAIN CONCEPT SELECTION IN ONTOLOGY LEARNING

by

JIZHENG WAN

A thesis submitted to the University of Birmingham for the degree of DOCTOR OF PHILOSOPHY

School of Computer Science

College of Engineering and Physical Sciences

University of Birmingham

May 2021

Abstract

One of the remaining challenges of Ontology Learning (OL) is the significant dependence on human interference to decide which of the “learnt” concepts from a training corpus are relevant and/or important to the domain of discourse. Though part of this challenge is deeply rooted in expert knowledge of the application domain, there is no doubt that a good relevance/importance measure with which concepts can be semantically judged serves as a good enhancement to the OL weaponry. A new measure called “Semantic Impact” (SI) is, therefore, proposed to bridge between explicitly defined formal semantics (in the form of ontologies) and the distributional semantics learnt from a vast amount of data.

SI aims to consistently and objectively quantify the semantic importance of a concept by aggregating two different measures: informativeness of a concept and its connectivity (or correlation) with the other concepts. Furthermore, it has been evaluated through two experiments.

The first experiment was conducted within the news domain – using 200 BBC News articles about Donald Trump (between February 2017 and September 2017) to semantically assess the impact of the concepts identified from the corpus/corpora. This experiment successfully learnt, for example, the Date concept is one of the most important concepts in the News domain, even if it has not been included in the BBC Core Concept ontology.

The second experiment was conducted within the biological area – using 2000 documents from PubMed on “Candida” to determine which diseases are more

“semantic impact” in the Candida domain knowledge. The results are promising. The proposed system has identified that the most correlated (connected) concept to Disease_D003645 (Sudden Death) is Disease_D003643 (Death) without any pre-defined knowledge (or symbolic processing of such labels). Furthermore, a semantic analogy has been identified between Disease_D008223 (Lymphoma) and Disease_D008228 (Non-Hodgkin Lymphoma) due to a close SI between the two concepts.

In addition, we have systematically evaluated the result from various angles and demonstrated that each component within the SI can produce a good and consistent result. At the macro-level, the overall SI result shows a strong clustering trend. At the micro-level, the SI results for both semantically important and non-important concepts are reasonable and reproducible. Moreover, we have compared it with a contemporary mainstream method to show the advantages of the SI algorithm together with its reproducibility.

Acknowledgements

I would like to sincerely thank my supervisors, John Barnden and Peter Hancox. Without them, there is no way I could accomplish this research. I would like to send my appreciation to my family and friends. Especially Claire Yinghui Ma, who I love forever until death do us part, and Alice Maya Wan-Ma, who recently joined our family. Their unconditional support is essential to me.

Table of Contents

List of Illustrations

List of Tables

Chapter 1 Introduction

The word “ontology” is a combination of the Greek word “onto-” (means being) and English “-logy”, which is used to denote a subject of study or interest. It is originally a philosophical concept that first appeared in English in Bailey’s dictionary of 1721 [1] to describe “*the nature and organization of being*” [2]. It was adopted in Computer Science (CS) in the 1980s for knowledge representation and reasoning purposes [3]–[5]. Although there are several definitions of “ontology” in CS, it is widely accepted as “*a formal language designed to represent a particular domain of knowledge*.” [6].

As with other knowledge-based studies in computer science research, the dream has been of developing a self-learning mechanism to automate the generation of such formal representations. Since Maedche and Staab coined the term “Ontology Learning” (OL) [7], which refers to extracting ontological information and conceptual knowledge from data sources and building an ontology from it, there have been experiments with various learning approaches. One of the challenges among all these approaches is that the system needs to decide whether or not a particular concept should be included in the domain ontology at some point in the learning process.

Using “Harry Potter” as an example, Horrocks [8] demonstrated how to use Resource Description Framework (RDF) and Web Ontology Language (OWL)[1]

[1] RDF and OWL are languages for ontology construction. Both of them are managed by W3C.
https://www.w3.org/RDF/
https://www.w3.org/OWL/

to describe the text below, which makes it possible for the software agent to discover that there is a `hasPet` relation between `HarryPotter` and `Hedwig`. Additional properties can be defined by analysing additional contents about Harry Potter at a later stage, such that `HarryPotter` is a `Wizard` and a `Student`, and that `Hedwig` is a `MagicalCreature` (or `SnowyOwl`).

"Harry Potter has a pet called Hedwig."

Assuming that we need to build an ontology containing key concepts in the universe of Harry Potter, an immediate question is what concepts could be considered as key, in other words, what makes Harry Potter "Harry Potter"? Three concepts (or ontology classes) have been identified in the above example: `Wizard`, `Student` and `MagicalCreature`. For those who are familiar with the original story (with sufficient domain knowledge), it is easy to understand that `Wizard` and `Student` are more "important" than `MagicalCreature` since the whole (original) story is about how a young wizard studies magic at Hogwarts and fights against an evil senior wizard who graduated from the same school. Without `Wizard` and `Student` (as concepts), Harry Potter would no longer be the "Harry Potter" that we are familiar with. On the other hand, the entire story would still be coherent if he had a different pet or had no pet at all.

However, for those who have only read the Fantastic Beasts series (written by J.K. Rowling, a spin-off of and prequel to the Harry Potter story) without the knowledge of the main Harry Potter story, it will be difficult for them to understand why `Student` is more important than `MagicalCreature`, since

the story they are familiar with is all about the wizards and their magic creatures (pets).

Both the main Harry Potter story and the prequel were written by J.K. Rowling, and share many common elements (e.g. Albus Dumbledore appear in both stories). If we consider the universe of Harry Potter, or the universe of J.K. Rowling, as a domain, then the main story and the prequel describe this domain from two different perspectives. It is relatively easy for a human being to understand that `MagicalCreature` is less important in the former (main story) but plays a significant role in the latter (prequel). The question here is what measure could a system use to reach the same conclusion?

In traditional Natural Language Processing (NLP) or Information Retrieval (IR) studies, various statistical-based approaches measure how important (or relevant) a word is with respect to a document in the corpus. However, the importance or relevance of a word to a document at a shallow semantic level is not quite the same as the importance or relevance of a concept to the domain knowledge at a deeper semantic level. Moreover, these statistical methods do not operate at a deeper semantic level, and only have limited ability to take contextual information into consideration, such as the contextual difference of the `MagicalCreature` concept between the main story and the prequel.

People can, of course, add some pre-defined knowledge (or rules) into the system and specify that `MagicalCreature` is an important concept only in the prequel and not in the main story, or even manually (i.e., by human intervention) remove it from the main story completely (as with the way to

handle stop words[2]). However, in a real-world application that contains hundreds or thousands of concepts, the rule-based approach may not be achievable because a) People may not have the required domain knowledge to make the decision (or judgement), and b) Rules and relationships set up manually for all the concepts will be, in fact, an equivalent manual process to building the ontology.

The root of this challenge is the lack of the ability to measure the "significance" or "importance" of various domain concepts with a consistent and objective approach at a deeper semantic level, although the actual application area of the ontology also plays a critical role in deciding what concepts the ontology should include. As with the semantic primes (like good, bad etc.), "significance" or "importance" is abstract and subjective, and different people have a different definition of it. Even the same definition may have different effects within different contexts (e.g. the `MagicalCreature` example).

Concept selection is one of the first things that all OL systems need to go through. Failing to identify the important concepts or accidentally including non-important concepts will significantly affect the OL process.

Hence, it is the contention of this thesis that an automated method to measure the importance (or relevance) of a concept to the domain knowledge (from a

[2] "Stop words" refer to the most common words appear in the document (e.g. "a", "an", "the"), which carry little information. In NLP, it is a normal approach to filter out those stop words before start processing the content.

specific perspective) is essential in the concept selection process across all OL methodologies.

By leveraging existing and publicly available tools for extracting concepts and relations, this thesis aims to produce a new method to derive a numerical measure that summarises how strongly a concept impinges on the domain of discourse. Moreover, it focuses on answering the following research questions to distinguish this new measure from the other existing methods:

RQ1. How to reduce the level of human intervention required in the concept selection and make the overall OL process less reliant on pre-defined domain knowledge?

RQ2. How to make the measure objective and consistent at a deeper semantic level?

To achieve the goal and provide answers to the above questions, a novel approach called Semantic Impact (SI) has been proposed in this thesis to assess the importance of a concept from two aspects: informativeness and connectivity.

Overall, SI is a predictive-based approach[3] that builds upon Distributional Semantic Model (DSM) [9], in other words, word vectors (refer to Chapter 2 for more details). SI aims to consistently and objectively quantify the semantic importance of a concept by aggregating two different measures: informativeness of a concept and its connectivity with the other concepts. The

[3] Predictive-based approaches are methods that use machine learning/deep learning to predict the outcome. It is opposite to the count-based approach. Refer to Chapter 2 for more details.

informativeness is measured by a new concept introduced in this thesis called Informative Coefficient (IC). The connectivity is measured by another concept introduced in this thesis -- Connectivity Coefficient[4] (CC).

As part of the IC calculation, the system uses a guiding ontology[5] to make an initial choice of informative concepts[6], then employs neural networks in a novel way[7] to test the robustness (or consistency) of the vector representation, which leads to a measure of its informativeness of the concepts -- One of the interesting phenomena discovered in this thesis is that the informative concepts have a more complex semantic distribution (than those non-informative concepts), that can be used to overcome the potential overfitting on the neural networks employed in the system. Hence, the IC calculation process has been designed to leverage this phenomenon and deliberately use the overfitting behaviour of the neural networks to distinguish informative concepts from those non-informative concepts. A more detailed discussion is provided in Section 4.3.3.2 and Section 4.3.3.3.

The CC is calculated by means of a novel and distinctive application of the Maximal Information Coefficient (MIC), a powerful existing, recently developed statistical method for measuring correlations in numerical data [10].

---

[4] It can also be called as Correlation Coefficient.
[5] A guiding ontology an existing and well constructed ontology that is closely related to the domain (or describes the domain from a different perspective). Please refer to Chapter 4 for more information.
[6] It is also possible to make an initial choice based on a seed list which only contains informative concepts. An example is provided in Section 8.2.5.
[7] Also, there is a new terminology coined in this thesis to describe this neural network setup – Neural Complex, which is discussed in Chapter 4.

This thesis is divided into eight chapters (together with ten appendixes). Chapter 2 introduces the background topics together with a discussion of the related work to demonstrate how the research in the present thesis could fit into the overall picture in the field. Chapter 3 highlights some of the specific problems this thesis tries to address.

The detailed methodology (of the Semantic Impact) will be presented in Chapter 4. Two experiments have been conducted in this thesis. Chapter 5 discusses the first experiment (a prototype with limited scale and depth), which is about assessing the importance of the concepts from the “Donald Trump” perspective within the News domain. The result of this experiment has been published at the Human-Centered Computing conference [11]. Chapter 6 discusses the second experiment – assessing the importance of the disease concepts within the Candida[8] domain.

A multi-faceted evaluation is provided in Chapter 7 to suggest the validity and advantages of the Semantic Impact method. This is then followed by a conclusion in Chapter 8, together with a discussion of future work.

[8] Candida is a type of yeast, which can cause various fungal infections. https://en.wikipedia.org/wiki/Candida_(fungus)

Chapter 2 Related Work

Before explaining the concept of the Semantic Impact (SI), it is crucial to clarify how it connects with other mainstream research, e.g. Ontology Learning, Distributional Semantics and Language Modelling, and how it differs from others.

To do so, this chapter will firstly discuss (Section 2.1) the history of Ontology Learning (OL), as one of the main motivations for developing SI as a contribution to the OL process. Then, it will explain which part of the OL process the SI algorithm is aiming to contribute to (Section 2.2). The SI algorithm is an extension of distributional semantics and deep learning. Therefore, Section 3 of this chapter will provide a brief review of the recent development around the Distributional Semantics Model (DSM) and the application of deep learning in NLP (predictive-based approach).

2.1 History of Ontology Learning

An ontology is “a formal language designed to represent a particular domain of knowledge” [6] and, as with other knowledge-based studies in computer science research, people have dreamed of developing a self-learning mechanism to automate the generation of such formal representations.

The term “Ontology Learning” (OL) was coined by Maedche and Staab [7] in 2001. Back then, machine learning was identified as one of the disciplines used to facilitate the construction process. However, it was also suggested that the learning mechanism should be more like a semiautomatic process with human

intervention, since fully automatic machine knowledge acquisition remains in the distant future [7].

This view was supported by one of the earliest ontology learning surveys published by Ding and Foo in 2002 [12], [13]. After reviewing more than six “state-of-the-art” ontology “generation” systems or approaches, they came up with the conclusion that all the existing methods relied on semi-structured data sources (e.g. XML, HTML) with seed-words provided by domain experts. They also suggested ontology learning from free-text or heterogeneous data sources was still within the area of the research laboratory and far from real applications due to the technical limitations. At the same time, they pointed out that other than machine learning, natural language processing techniques should also be considered as the most useful disciplines in this area since they had revealed promising results in the concept extraction process.

Two additional surveys were published around the same time by the OntoWeb Consortium [14] and Shamsfard and Barforoush [15] with a very similar view. Moreover, the latter specifically pointed out that most of the reviewed approaches relied on pre-defined domain knowledge, and again most of them required human intervention.

In summary, from 2002 to 2012, most of the OL systems used pre-defined domain knowledge, e.g. seed-words list, and heavily relied on human intervention as part of the decision making process. Only a few attempts were made to try to limit the demand for pre-defined knowledge and reduce the amount of intervention required from the domain experts, but none of those attempts was considered successful. For example, *Zhou et al., 2007* [16]

introduced a new hypothetical model which claimed to have no user intervention. However, later on, it was criticised by others for overlooking the significance of logic-based techniques in forming axioms [17].

The situation has not changed since then. In one of the most recent surveys published in 2018, the authors of [17] reviewed 140 ontology learning-related papers. There were, indeed, quite a lot of new developments in the previous decade. For example, with the help of machine learning, the accuracy of term extraction has already improved significantly [18]. BERT, the state-of-the-art language model, opened a new way to do simple relation and knowledge extraction [19], which is another main challenge in the OL study. However, these approaches only made minor or flawed contributions in those two areas and remained heavily reliant on pre-defined domain knowledge and human intervention.

Therefore, reliance on pre-defined domain knowledge and human intervention are the main issues or ‘the bottleneck’ of developing a fully automatic ontology learning approach. Since often the reason a system needs human intervention is to provide the necessary domain knowledge to help the decision-making process, it is reasonable to conclude the level of human intervention could be reduced if the designed OL approach were less dependent on pre-defined knowledge.

2.1.1 Additional Challenges

Besides the challenges in the OL discussed above, this section will briefly outline some additional problems that add another layer of difficulty in this field.

2.1.1.1 Ontology Class and Ontology Instance

In philosophy, there is a continuing discussion of "*Being*" and "*Entity*". Quite often, "Being" has been defined as an extremely broad concept encompassing objective and subjective features of reality and existence. It is the most universal of concepts, and as R. Munday explained in the "Glossary of Terms in Being and Time" [20] (in the section about "Being"), it transcends any categorical distinction people care to make in our apprehension of the world, and it does this by existing above and beyond any notion of a category that we can form in our understanding. "Entity" often means a specific item, a concretization of "Being", moreover, a "Thing".

The terminology of "Ontology" was adopted in Computer Science (CS) in the 1980s, and several standard "components" were developed to interpret this philosophy idea. For example, a Class in ontology is an equivalent of Being and used to represent a group, set or collection of objects; an Instance (or also called individual), on the other hand, is an equivalent of Entity to represent a particular object.

Let us use the Wine Ontology [21] shown in Figure 2-1 as an example to illustrate the difference. Each bubble in the figure represents an ontology class (e.g. Fruit, Region), and different classes join together with an "is-a" relation. Ontology instance or individual, in this case, is the "value" of the class. For example, `WineColor` is an ontology class and could be assigned different

"values", like `Red` or `White`[9], to explicitly state an individual member (or members) that can be categorised in this class[10].

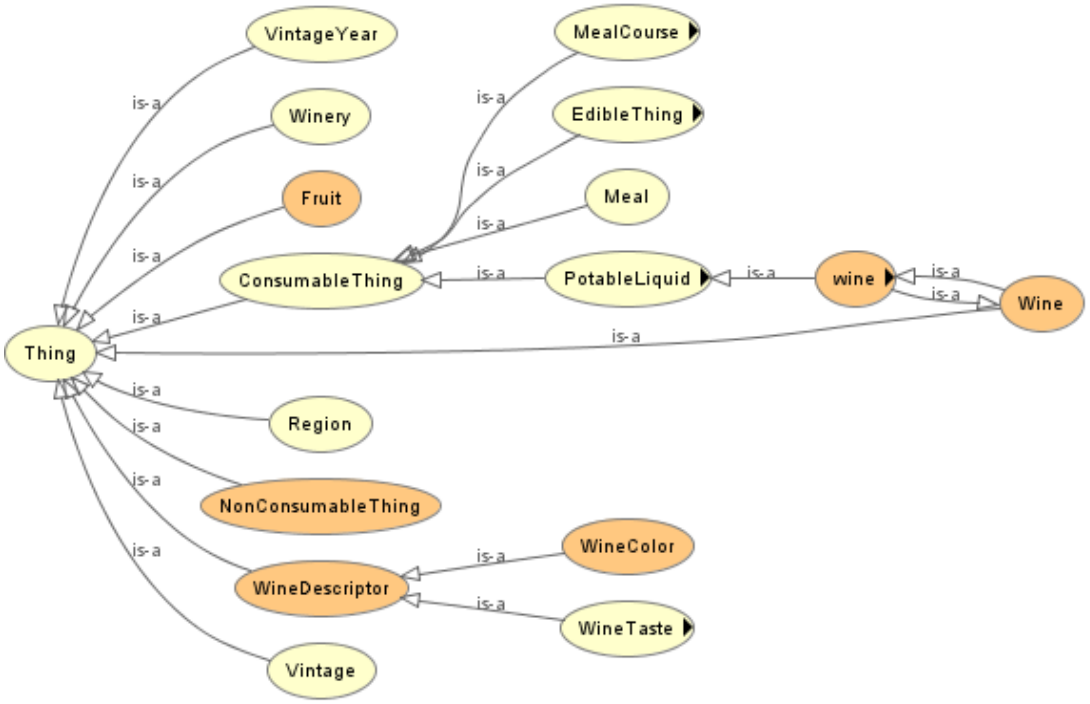


Figure 2-1 Wine Ontology[11], a demo ontology used in the W3C "OWL Web Ontology Language Guide"

It is easy to understand so far, and the distinction between ontology class and ontology instance seems to be precise. However, a well-constructed ontology (by using an ontology language like RDF and OWL) can include both ontology classes and their associated instances. Strictly speaking, those instances are not "essential" (which depend on the usage of the ontology) to the ontology and have no influence on the relation between classes, but it is a common approach for people to include instances/individuals in the ontology. In the Wine Ontology, more than 30 instances/individuals have been declared to populate various

[9] Figure 2-1 is a class-level picture and therefore does not include values of individuals.
[10] For now, please ignore the fact that Red or White can also be classified as a class in a different scenario.
[11] https://www.w3.org/TR/owl-guide/wine.rdf, please ignore there are two different "Wine" concepts and have different colours as they are not related to the topic of this thesis at all.

classes. For example, `Red`, `Rose` and `White` are the instances for `WineColor`; `Dry`, `OffDry` and `Sweet` are `WineSugar` (which is a sub-class of `WineTaste`) instances.

As a consequence, when people use the terminology “ontology”, what they actually refer to could be ontology classes or ontology instances[12] or both. This ambiguity, in fact, leads to a two-fold meaning of “Ontology Learning” (OL) as well: a) extract the related information about the various ontology classes (e.g. People, Organisation) from the corpus and place them into a hierarchy structure (linked with different relations) to form an ontology model, and b) extract various instances (e.g. John, Peter, University of Birmingham) from the corpus and then assign them to identified ontology classes to populate an existing ontology model. It is on the former that this thesis focuses. In the rest of the thesis, the word “ontology” on its own means Ontology Model (e.g. the Wine Ontology); Ontology Class will be denoted as “ontology class” or simply “class”, and Ontology Instance/Individual will be denoted as “ontology instance” or simply “instance”.

Hence, within this thesis, the concepts selected or ranked by the Semantic Impact (SI) algorithm are the ontology class candidates (e.g. Fruit and Winery in the Wine Ontology). The primary purpose of the SI is to provide a quantified assessment to measure how semantically important these candidate ontology classes are to the domain knowledge itself.

[12] It does not mean a well-constructed ontology can consist *only* of instances.

2.1.1.2 Ontological Diversity

In addition to the ambiguity between ontology class and instance, the angle or perspective from which people could build an ontology is not always apparent. As an abstract concept, a domain ontology can be constructed from different perspectives. For example, fruit farmers, who have substantial knowledge about growing apples, may have a different understanding of the “Apple” concept compared with ordinary consumers who probably focus more on where to buy apples. Hence, an `Apple` ontology built by a farmer could be very different from the one built by a consumer because they tend to “describe” this concept from different perspectives.

So, what is the best perspective from which to construct an ontology? It is likely to be an open question without a definite answer since it largely depends on the domain and the actual usage/application of the ontology itself. For example, there are many ways to define the concept of `People`. Within the academic environment (domain), `people` can be a concept constructed with various roles in the university, as Figure 2-2 shows.

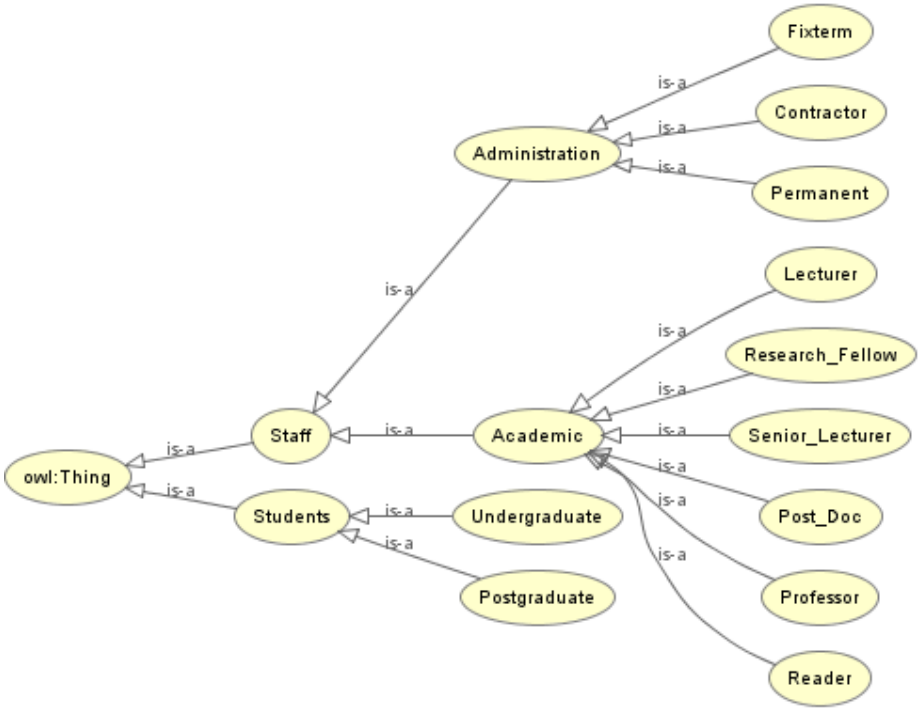


Figure 2-2 People Ontology within the University domain.

While in the social networking domain, `people` focus more on what kind of information is required to describe individuals and their relations (Figure 2-3).

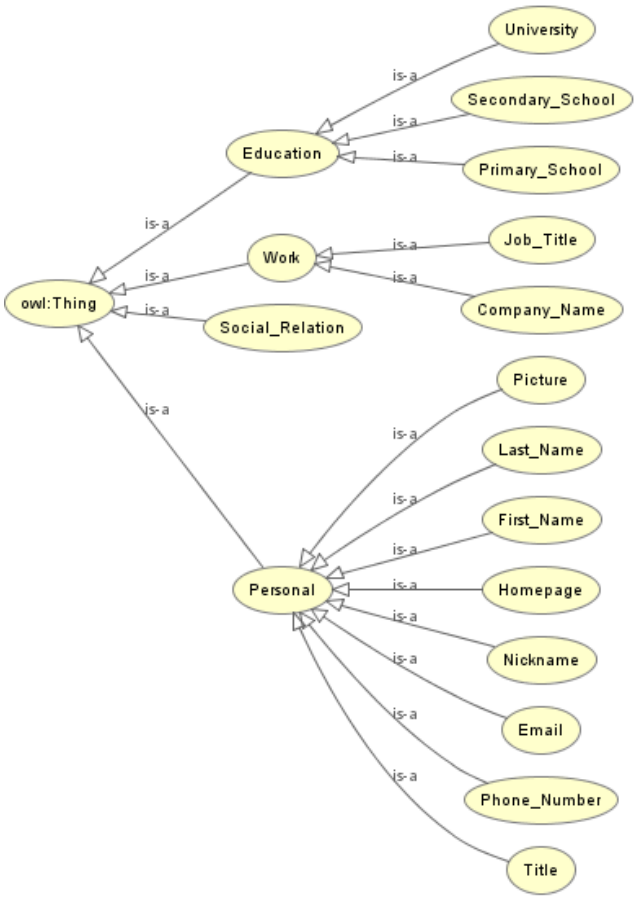


Figure 2-3 Friend of a Friend (FOAF)[13] Ontology example

Although the above two examples have completely different class lists, both of them are, in fact, correct as they describe the `People` concept from two different perspectives. The fact that a concept can be presented differently from various perspectives is denoted as ontological diversity in this thesis.

Unfortunately, ontological diversity adds another layer of complexity to the ontology learning and concept selection process and tends to make it more reliant on human intervention.

Imagine that people are trying to construct (from documents) the ontology shown in Figure 2-2– a people ontology within the academic domain. In an ideal

[13] FOAF project: http://www.foaf-project.org/

situation, the corpus should only contain documents that describe people who work in the higher education sector without any unnecessary/unrelated information.

However, in reality, knowledge is like a dense network, and it is difficult to isolate a specific area of knowledge from others. It is almost guaranteed that there will be concepts and entities that belong to other non-university domains (e.g. the parents of the students). Alternatively, we can consider it as an academic ontology (instead of people ontology), but constructed from the people's perspective. In which case, we can collect a set of documents about the higher education sector to ensure everything mentioned there is academic-related. However, it does not improve the situation since there is no guarantee that the concepts and entities will all associate with people (e.g. the university's course information).

Hence, it requires extra knowledge of both areas (the domain and the perspective) in order to handle these challenges. It is why the concept selection process heavily relies on pre-defined knowledge and human intervention.

However, as also mentioned above, people from different backgrounds (or perspectives) may have a different understanding of the ontology (or some concepts within it). Hence, it is hard to ensure that their involvements are objective and consistent. Section 3.2 will provide a more detailed discussion about it.

2.2 Semantic Impact and its Focus

It has already been described [17] that the OL process contains multiple stages. Different learning approaches use different ways to define these stages, but in general, they should cover four areas, as Figure 2-4 illustrates below.

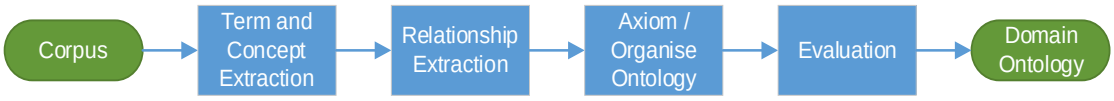


Figure 2-4 Ontology learning step-by-step process.

Each of the steps includes a few sub-tasks. For example, Term and Concept Extraction can be split further into a) pre-processing, b) extraction of all the terms and concepts from the pre-processed text and c) filtering out of the irrelevant concepts and selecting only the domain-related concepts for further processing.

In traditional Natural Language Processing (NLP) study, there are many existing tools and frameworks for Named Entity Recognition (NER), which is a technique that automatically identifies named entities in a text and classifies them into pre-defined categories. Within this research, an existing NER method will be adopted to extract semantic information from the two corpora instead of reinventing the wheel.

As introduced in the last chapter, SI is a distributional-semantics based measure that aims to provide a consistent and objective measurement of the impact (importance and significance) a specific concept could supply to the domain knowledge at a deeper semantic level. It is an alternative to the existing statistics-based measures such as TF-IDF (short for Term Frequency–Inverse Document Frequency) and C/NC-Value (methods introduced in [22]). Hence,

this research is focussed on the last sub-task within the first step (c, mentioned above) of the OL process discussed above – to provide a far more advanced measure to contribute to the domain-related concept selection process.

Moreover, making the concept selection process less dependent on the pre-defined domain knowledge is one of the specific areas that SI contributes to.

Chapter 4 will provide a more detailed introduction about SI, but it is useful to highlight within this section that the main advantages of SI are as follows:

- It does not rely on pre-defined knowledge of the target domain, for which we are aiming to build an ontology, although information about a related domain is required. Hence, it requires less human intervention.
- Unlike the count-based measures (discussed in Section 2.4), the SI operates at a deeper semantic level, which takes into consideration the position of a concept in the sentence and its context, whereas the former ignore these important factors completely, which will limit its ability to assess the semantic importance and relevance for various concepts.
- Instead of being a one-sided measure, SI is a combination of informativeness and connectivity, and it makes the measurement more consistent and objective.

2.3 Distributional Semantic Models and Embedding

Computational linguistics research holds that word meaning can be represented by its contextual information because similar contextual distributions tend to share between semantically similar words [23]. The idea of Distributional Semantic Models (DSM), which have also been referred to as

word space or vector space models, is that the meaning of words can, to a certain extent, be inferred from their usage and therefore the semantics can be encapsulated in high-dimensional vectors based on the nearby co-occurrence of words [9].

One of the most significant benefits of representing words with high-dimensional vectors is that the number-based representation can then be used as the input for further numerical processing, e.g. input for a neural network (NN). Hence, to a certain extent, DSM is simply a vectorisation or encoding process. But unlike the one-hot encoding, which simply assigns a unique number to each word in the vocabulary and the maximum number equals to the size of the vocabulary, the rationale behind DSM is to keep the original contextual and semantic information during the transformation process.

DSM has a much longer history than OL, and it has been widely used in various OL approaches, mainly for two reasons:

Firstly, almost all the OL approaches are associated with Natural Language Processing (NLP) and/or Machine Learning (ML) [24]. And those approaches generally require converting textual data into number-based representation and use as input for further processing purposes.

Secondly, as discussed already, one of the main purposes of human intervention is to provide the related domain knowledge to support various decision-making processes in the OL approach. Since DSM itself contains contextual and semantic information, one of the common approaches is to consider it as a supplemental data source (in addition to pre-defined knowledge

and/or the knowledge inputted through human intervention) for domain knowledge extraction.

Therefore, in order to make the OL process less dependent on pre-defined domain knowledge and human intervention, a good DSM should provide not only an accurate numerical representation but also sufficient contextual and semantic information. The following section will provide an overview of two different DSM construction approaches – the count-based approach, which is a primitive approach purely based on statistical information, and the predictive-based approach, a more advanced approach developed more recently (compare with the count-based).

2.3.1 Count-based Approach

The count-based approach, also referred to as weighting-based, is the most primitive way to build a DSM, because it simply counts the co-occurring words around the defined basis vocabulary list – a list that is normally collected manually to define what vocabularies/words should be included in the DSM.

Consider the following texts as an example [25]:

… and the *small cute* **kitten** *purred* and then …

… the *cute furry* **cat** *purred* and *miaowed* …

… that the *small* **kitten** *miaowed* and she…

… the *loud furry* **dog** *ran* and *bit* …

Example 1 kitten-cat-dog example

Assuming that the basic vocabulary defined in this case is {bit, cute, furry, loud, miaowed, purred, ran, small}, and the words to be analysed are {kitten, cat, dog}.

Let us use **kitten** as an example. By counting how many times each individual word in the basic vocabulary list appears in kitten's context window (3 on each side), we can easily work out a vector $[0,1,0,0,1,1,0,2]^T$ to represent the word **kitten**, because there are 0 occurrences of "bit", 1 occurrence of "cute", and so forth, ending with 2 occurrences of "small". The corresponding vector for **cat** and **dog** can be built in the same way, and finally, we can generate a matrix to represent {kitten, cat, dog} and each column will be the corresponding vector as shown below:

Kitten	**Cat**	**Dog**
0	0	1
1	1	0
0	1	1
0	0	1
1	1	0
1	1	0
0	0	1

The above example, which has also been referred to as bag-of-words [26], is just a demonstration of the simplest way to build a DSM with the count-based approach. In the real application, there are many other more "advanced" methods to improve how they count the co-occurrence and together with additional steps to improve the performance (e.g. dimension reduction).

Use the Latent Semantic Analysis (LSA) as an example. Scott Deerwester and his colleagues first introduced it in 1988 as a technique for improving information retrieval [25]. Similar to the kitten-cat-dog example, the first step of the LSA is to build an original matrix to represent the vocabulary. However, LSA has a different way to construct the matrix, where rows are individual words and

columns are documents or equivalent units. The value of an individual cell is the frequency with which a specific word (row) occurs in a specific document (column), as shown below (in some research, TF-IDF value has been used instead).

$$\begin{array}{c} \\ \text{Word 1} \\ \\ \text{Word } m \end{array} \begin{array}{c} \begin{array}{ccc} \text{Document 1} & & \text{Document } n \end{array} \\ \begin{bmatrix} x_{1,1} & \cdots & x_{1,n} \\ \vdots & \ddots & \vdots \\ x_{m,1} & \cdots & x_{m,n} \end{bmatrix} \end{array}$$

Since the order of words in the sentences has no influence on the representation itself, it can be considered as another version of the “bag-of-words”. However, compared with the first example, which normally produces a matrix with a large dimension, LSA quite often uses Singular Value Decomposition (SVD) to reduce the dimension number and make the downstream tasks (e.g., train a neural network) less computationally expensive.

Besides LSA, there are other count-based approaches, but it is safe to say that they all follow the same principle – simply count (or use a simple statistical model to calculate) the co-occurrence. The advantage of such an approach is quite obvious and easy to understand and implement. So are the disadvantages -- it does not take into consideration the position of words in the sentences and assumes that all words are entirely independent of each other. Moreover, some common challenges within this approach limit its usage in model NLP or OL tasks. For example, words (or terms) with high counts do not necessarily mean they are informative. It may well be the case that they are just independently frequent contexts that do not contain much information in themselves [23]. Thus,

quite often, there will be various transformations (re-weighting, normalisation etc.) attached to these count-based approaches. Even so, count-based approaches are still not ideal for OL for two reasons.

Firstly, from the OL perspective, a good DSM should contain sufficient contextual and semantical information to assist the domain knowledge extraction. However, as suggested by previous research [19], [28], the count-based approach only contains limited semantic information and is often outperformed in related evaluation tasks by the predictive-based approach, which will be discussed in the next section.

Secondly, the various transformations mentioned above will add another complex layer to the problem and will make the process even more reliant on domain knowledge or human intervention. For example, the same word can have a different semantic meaning in different domains/contexts (e.g. "*bank* account" and "*bank* of the river" [29])and therefore will require domain knowledge in order to assign a proper weight to it. This counteracts the idea of reducing the level of human intervention an OL system requires to make it more automatic.

2.3.2 Predictive-based Approach

The history of the count-based approach goes back to almost a half-century ago. In comparison, the predictive-based approach is still very young and relies on some of the techniques developed in Deep Learning (DL) research. Many remarkable methods or frameworks have been developed in this area. For example, Word2Vec [30]–[32], ELMo [33] and BERT [34]. We will use Word2Vec and BERT as examples in this section because the former has been

adopted in this research to produce the word-level representation, and the latter is state-of-the-art within this field.

As briefly mentioned before, the predictive-based approaches generally contain more enriched contextual and semantical information than the count-based approach. It is mainly because of the usage of the Language Model (LM).

Language Modelling is one of the foundations in modern Natural Language Processing. Essentially, it is a method used to calculate the probability of a given sequence of words, $P(w_1, w_2 \ldots, w_n)$ arising in texts of the genre of interest. As Phil Blunsom pointed out in one of his lectures [35], quite a lot of NLP tasks can be classified as (conditional) language modelling related. For example, probability measures of the following sorts are used in the tasks indicated:

- Translation: P_{LM}*(Les chiens aiment les os* ||| *Dogs love bones)*
- Question answering: P_{LM}*(What do dogs love?* ||| *bones* | β*)*
- Dialogue: P_{LM}*(How are you?* ||| *Fine thanks. And you?* | β*)*

The predictive-based approach is, in fact, a side effect or an intermediate product of a bigger neural network-based “deep learning” NLP architecture for language modelling [28].

In the traditional count-based approach, the DSM matrix (vectors for all the words in the vocabulary) is collected based on the co-occurrence and then re-weighting of each word vector based on various criteria (e.g. TF-IDF or domain knowledge). Within this new approach, a word vector is the weight of the hidden layer in a neural network and can be optimised to maximise the probability of

its context that has been observed in the corpus (as the training dataset) [36], [37]. In other words, use the corpus as the training dataset to train a neural network and get a language model that can best align with the context in the training dataset. Once the neural network has been adequately trained, the weights of the nodes in its hidden layer will provide the word vectors that can best fit into the existing context in the corpus.

When *Bengio et al*. first introduced this predictive-based approach in 2003 [36], the overall computational performance was one of the major drawbacks. For example, using 40 CPUs, it took over three weeks to run only five training epochs on the Associated Press (AP) News corpus. Hence, this approach was considered too computationally expensive to implement on a large scale, and one of the top priorities for "future" research was to improve speed-up techniques as well as ways to increase capacity without increasing training time too much [36], [38].

Mikolov et al. made a significant contribution in this area by introducing the Word2Vec framework/toolkit and its application in 2013 [30]–[32]. By using techniques like negative sample and hierarchical softmax, which have been introduced in those papers, it managed to speed up the overall performance considerably. Moreover, it provided two different training models: Continuous Bag of Words (CBOW) and Skip-gram to support different scenarios.

CBOW aims to predict a target word by using its surrounding context. In other words, combine the representations of surrounding words to predict the word in the middle. Its architecture is illustrated in Figure 2-5 [39].

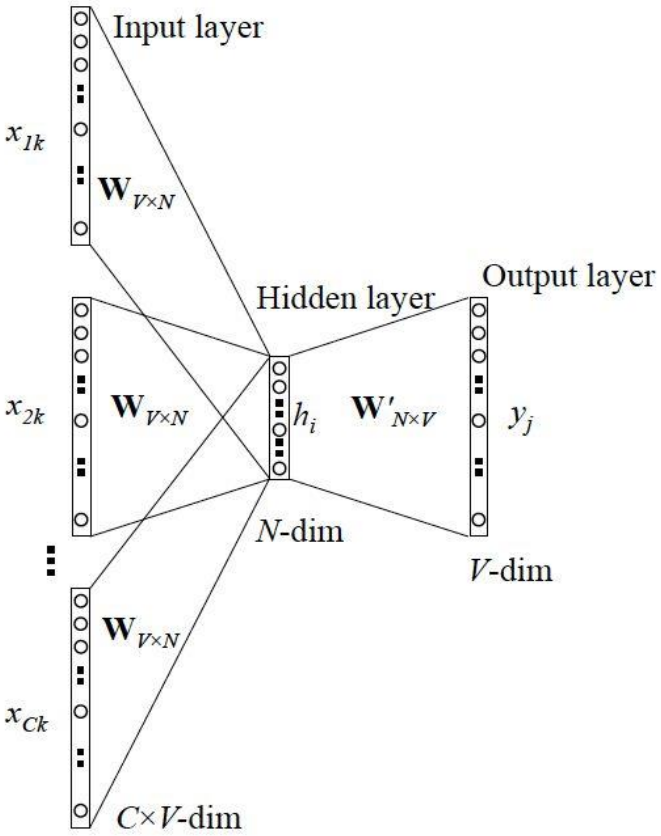


Figure 2-5 The CBOW model

The $\{x_{1k}, x_{2k}, \ldots, x_{Ck}\}$ are the initial V dimension vectors for the words in the vocabulary list that are within the target word's context window. There are many ways to generate these initial vectors, and Mikolov selected the Huffman Tree method [40] as part of the hierarchical software implementation in the Word2Vec model [41]. Each $W_{V\times N}$ is the initial weight ($V \times N$ is the dimension size) of the associated word in the vocabulary (the context word). h is a N dimensional vector that represents the output of the hidden layer and is calculated as:

$$h = \frac{1}{C} W^T (x_1 + x_2 + \cdots + x_C) \qquad 2\text{-}1$$

$$= \frac{1}{C} \left(v_{w_1} + v_{w_2} + \cdots + v_{w_c}\right)^T \qquad 2\text{-}2$$

where C is the total number of the context words, $w_1, \ldots, w_C$ are words in the context and v_w.is their associated vector. Then it will follow the measure neural network process to continually update its parameters to maximise the conditional probability of the actual output context word w_O given the input context words $w_{I1}$to w_{IC} with regards to the weights, by using the loss function shown below:

$$E = -logP(w_O|w_{I1}, w_{I2}, \ldots, w_{IC}) \quad \text{2-3}$$

The Skip-gram model, which was introduced in [32], is the opposite of the CBOW model, where the vector of the target word now becomes the input layer and the context word vectors become the output layer. In other words, the purpose of this model is to learn word vector representations that are good at predicting word context in the same sentence. The model's architecture shown in Figure 2-6 [39].

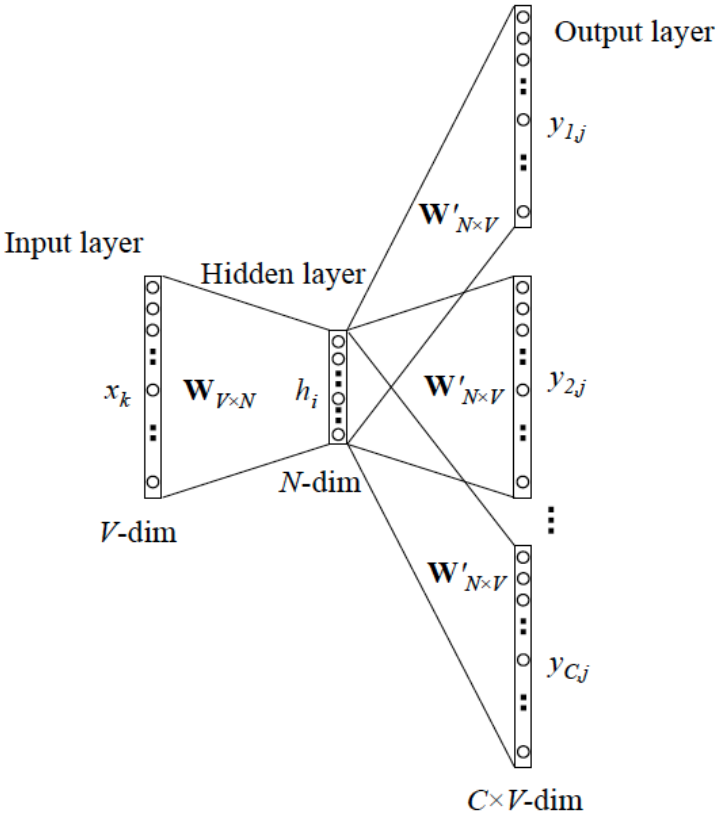


Figure 2-6 The skip-gram model [38]

The rest of the process is very similar to CBOW, except that the loss function is changed to the formula below to maximise the average log probability:

$$E = -logP(w_{O1}, w_{O2}, \dots, w_{OC}|w_I) \quad 2\text{-}4$$

$$= -log \prod_{C=1}^{C} \frac{\exp(u_{C,j_C^*})}{\sum_{j'=1}^{V} \exp(u_{j'})} \quad 2\text{-}5$$

As explained in [39], where w_{OC} is the c-th word in the output context words; w_I is the input word; y_{cj} is the output of the j-th unit on the c-th panel of the output layer; $u_{C,j}$ is the net input of the j-th unit on the c-th panel of the output layer and j_C^* is the index of the actual c-th output context word in the vocabulary list.

Both CBOW and Skip-gram have a similar architecture as well as a low computational complexity compared with the other traditional neural network-

based language models. According to Mikolov, both of them can be used on large datasets. In practice, however, Skip-gram provides a better word representation for infrequent words, although it is slower than CBOW [31].

There are some extensions of the original Word2Vec method. For example, Le and Mikolov presented a modified version called Doc2Vec [42]. Instead of creating vector representation at the word level, the aim of Doc2Vec is to generate a numeric representation at the document level.

In comparison with the count-based approach, there are many advantages of using a predictive-based model like Word2Vec. For example, in 2014, *Baroni et al.* presented a systematic comparative analysis between Word2Vec and count-based approaches on five different benchmarks: semantic relatedness, synonym detection, concept categorisation, selection preferences and analogy. The conclusion of this survey indicates that "*a neural word representation method like Word2Vec outperformed count-based distributional methods on the majority of the considered tasks*" [19] p.2465. They then recommend that anybody interested in using DSMs for theoretical or practical applications should opt for the predictive models instead of count-based methods [28]. These five benchmarks are not task or application specified but have been used in the NLP related study in a generic way. Since NLP is one of the underpinning studies of the OL, it is common to see that these benchmarks have also been applied to OL.

One of the most significant disadvantages (in some applications) of Word2Vec is that it has a 1-to-1 mapping relationship between a word and its associated vector. In other words, it cannot handle polysemy - the fact that a word may

have different meanings in different contexts. For example, the word “bug” has a very different meaning in the biology domain compared with the computer science domain. By using a method like Word2Vec, it can only generate a static vector to represent the word “bug” as a whole instead of producing different vectors within different contexts.

It would be unfair to claim that a method like Word2Vec does not take into consideration the context. In fact, it does, since there is a window size parameter to define how many contextualised words it will process on each side of the target word. However, it can only take account of context to a very limited extent and in scenarios where people need to bring context-awareness to the next level and produce context-dependent word representation, then they will need to use a different approach, such as BERT.

BERT, short for Bidirectional Encoder Representations from Transformers, is the state-of-the-art method designed to produce contextualised word representation introduced by *Devlin et al.* in 2019 [34]. It contains two steps: pre-training and fine-tuning, as shown in Figure 2-7. The pre-training is not task or domain-specific. It is a general-purpose language model trained on BooksCorpus [43], which contains 800M words, and English Wikipedia, which contains 2500M words.

In order to apply this pre-trained representation to the downstream tasks (e.g. question answering and natural language inference), there is a fine-tuning process in BERT to adjust the model with additional task-specific parameters. The following section will only provide a brief introduction to the pre-training process and skip the latter.

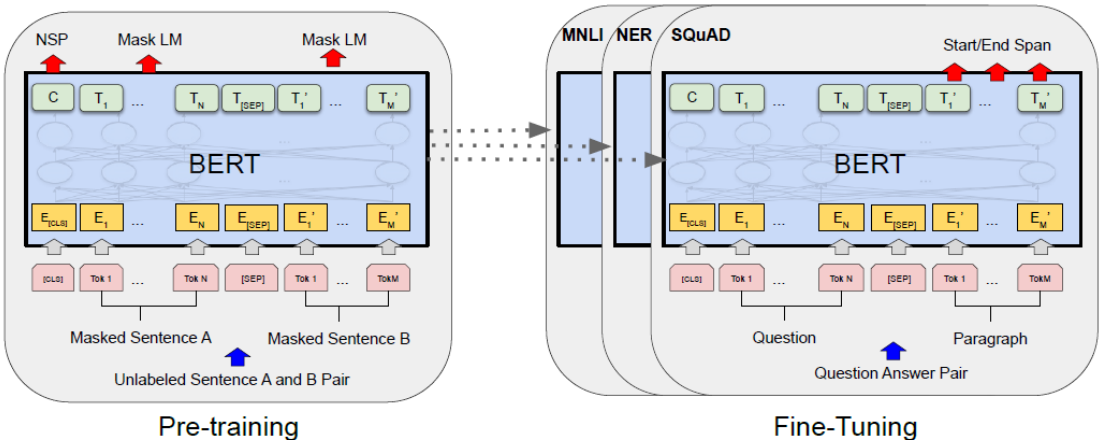


Figure 2-7 Overall BERT procedures [34]

One of the most significant differences between BERT and other methods is the way it handles the input and output representations (other than the fact that BERT is a bidirectional approach). The input/output layer in a traditional method like Word2Vec is a sequence of individual word representation/vector, whilst in BERT, the input embeddings map the words or phrases from the original input sentence to vectors of real numbers, which are the sum of three different embeddings targeted at the token level, segmentation level and the position level as the figure shows below [34].

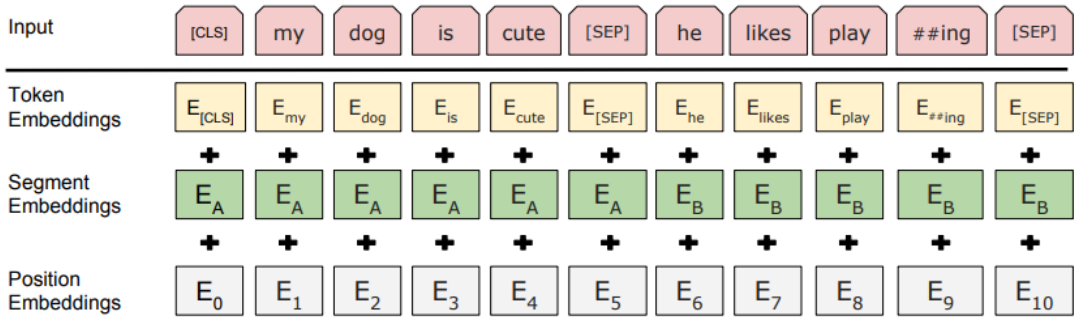


Figure 2-8 BERT input representation [34]

In the above example, the input contains two sentences: “My dog is cute. He likes playing.”. Using a method called WordPiece, these two sentences will be tokenised into the format shown in the “Input” layer. WordPiece is a data-driven subword segmentation algorithm introduced in [44] to achieve a balance

between vocabulary size and out-of-vocab words. The vocabulary is initialised with individual characters in the language, then the most frequent combinations of symbols in the vocabulary are iteratively added to the vocabulary. There is an existing WordPiece vocabulary in BERT that contains 30,000 tokens, and each of them is represented by a 768-dimensional vector. The original words that are not part of the vocabulary are represented as subwords and characters with a "##" mark in front of them. This is why the word "playing" in the original sentence has been split into "play" and "##ing". Moreover, BERT will add two special tokens to determine the beginning of every input example ([CLS], short for classification) and the non-consecutive sentence ([SEP], short for separator). Hence the embedding of those tokens from the input sentences are simply the value (768-dimensional vector) of the associated token in the defined/existing 30,000 vocabularies and is denoted as E shown in the "Token Embeddings" layer in the above picture.

The Segment Embeddings layer is a straightforward process. Assign value 0 to those embeddings that belong to the first sentence (denoted as E_A in the above figure), and value 1 to those second sentence embeddings (E_B). Then build the tensor accordingly, and again the shape should be (1,11,768). And the Position Embedding is simply the position (the order) of each embedding within the given input denoted as $E_n, n \in \{0,10\}$ in the above picture.

BERT is pre-trained by using two unsupervised tasks: Masked Language Model (Masked LM) and Next Sentence Prediction (NSP).

The Masked LM task is straightforward: randomly replace 15% of the WordPiece embeddings with "[MASK]" and then feed the corresponding

vectors into an output SoftMax over the vocabulary as in a standard LM [34]. Unlike CBOW or the denoising auto-encoders [45], it only predicts the masked words rather than going through the entire input.

The purpose of the NSP is to train a language model that can understand sentence relationships. More specifically, select sentences A and B, and then predict if B is the next sentence that follows A or not.

BERT has made many remarkable achievements and is able to store vast amounts of linguistic knowledge [46]–[48]. It has several advantages compared with the structured knowledge base [19]. For example, it does not require schema engineering to produce a structured representation (e.g. the triple structure in the knowledge graph) to store and query factual knowledge (an example shown in Figure 2-10 at the later part of this chapter).

It is challenging to compare BERT with Word2Vec directly. Word2Vec produces word-level representations, while BERT is more likely to be sentence-level, where the same word will be associated with a different vector in different contexts. In the next section, a more detailed analysis will be provided to show how these embedding methods link with the overall ontology learning process and whether they could make a direct contribution to the domain concept selection challenge.

2.4 Existing measures for Concept selection

Essentially, various DSM and embedding technologies allow the creation of a more accurate word representation with richer semantic and contextual information. However, it does not directly tell us which concept is more closely

related or more important to the domain knowledge. Therefore, other methods need to be used to “re-weigh” them. This section will focus on introducing the popular measures that have been used previously.

By cross-referencing the development of these measures and the development of DSM & Embedding, it is interesting to find that they take a very similar path -- from the count-based approach to the predictive-based approach.

Prior to the count-based era, there was a time when the seed-word list, which was defined by the domain experts, was the only approach to select domain-related concepts at the very early stage of the OL development (before 2002). For example, in an early OL survey [12], all the reviewed systems used seed-word as the primary method to extract and select the related concepts from the structured data source.

Seed-word could be an efficient method to identify the related terms and concepts with the same root word. However, it usually requires domain experts (human intervention) to provide the initial list with the limited ability to expand itself to cover new terms and concepts. Hence, from the OL perspective, it is a primitive and sub-optimised method for term and concept selection since it does not limit the demand for pre-defined knowledge or reduce the amount of intervention required from the domain experts.

It is important to highlight here that a good measure should not only operate at a deeper semantic level (in order to align with the DSM development), but also contribute to the overall objective – make the system less reliant on the pre-defined domain knowledge and human intervention.

From 2002 to 2013, research slowly moved into an area where unstructured free text was used as the data source for ontology learning instead of structured data (e.g. databases and XML-annotated documents). Therefore, other than seed-word, people began using those count-based or statistics-based methods/measures for the concept selection task. Below is a summary of the related OL tasks and the common techniques covered by *Wong et al.* in their survey of 2012 [15].

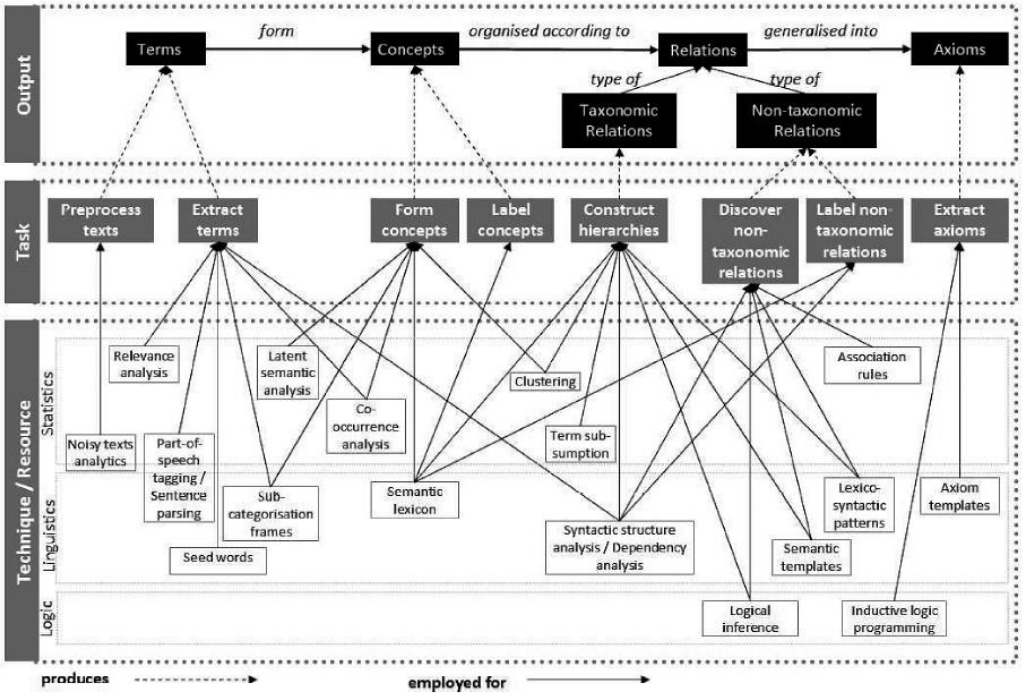


Figure 2-9 Overview of OL tasks and common techniques[15]

Text2Onto is probably one of the most well-known OL systems developed during the period 2002-2013 [49]. For concept selection, it implemented a few different measures (or standards) to assess their relevance: Relative Term Frequency (RTF), Term Frequency Inverted Document Frequency (TF-IDF), Entropy and the C/NC-Value. Before explaining these measures in detail, it is important to point out that the Text2Onto system, as well as the other similar systems, only use them as an indication and it is still down to humans to make

the final decision. For example, when using the TF-IDF method, Text2Onto will simply list all the concepts in the corpus with a high TF-IDF value and ask the user to provide appropriate feedback (True, False or Don't know).

Some of the popular measures include:

Relative Term Frequency (RTF). The idea for this measure is very simple: in a single document, the frequently recurring words are more significant/informative than the others.

Let tf_i be the number of occurrences of the i-th item in the document, then $\{tf_i\}_{i=1}^{n}$ denotes all the term frequencies of a single document, relative term frequency is calculated as [50]:

$$tfw_i = c_1 + c_2 \frac{tf_i}{\max\limits_{1 \le i \le n} \{tf_i\}} \qquad \text{2-6}$$

where c_1 and c_2 are two constant parameters within [0,1].

TF-IDF. One of the most popular methods used in the information retrieval area. There are several variants of TF-IDF, one of the widely used versions is calculated as below:

$$tfidf(t,d,D) = \begin{cases} \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \times log_{10} \frac{N}{|\{d \in D : t \in d\}|}, & |\{d \in D : t \in d\}| \neq 0 \\ 0, & |\{d \in D : t \in d\}| = 0 \end{cases} \qquad \text{2-7}$$

Where $f_{t,d}$ is the raw count of a term t in a document d, D is the document set (corpus) and N is the total number of documents in the corpus, $|\{d \in D : t \in d\}|$ is the number of documents where the term t appears.

BM25. BM is short for Best Matching, and is an improved method (compared with TF-IDF) that addresses some of the limitations in TF-IDF, e.g. taking term saturation and document length into consideration.

Given a query Q that contain keywords $q_1, q_2, \ldots, q_n$, the BM25 score of a document D is calculated as below:

$$score(D,Q) = \sum_{i=1}^{n} \ln\left(\frac{N - n(q_i) + 0.5}{n(q_i) + 0.5} + 1\right) \cdot \frac{f(q_i, D) \cdot (k_1 + 1)}{f(q_i, D) + k_1\left(1 - b + b \cdot \frac{|D|}{avgdl}\right)} \quad \text{2-8}$$

where

- N is the total number of documents in the corpus.
- $n(q_i)$ is the number of documents that contain q_i.
- $f(q_i, D)$ is the term frequency of q_i in the document D.
- $|D|$ is the length of the document D.
- $avgdl$ is the average document length in the corpus.
- k_1 and b are parameters. Normally $k_1 \in [1.2, 2.0]$ and b=0.75.

C/NC-Value. It is an approach introduced in [22] for multi-word terminology extraction. C-value is a domain-independent method for multi-word automatic term recognition, which aims to improve the extraction of nested terms.

Whereas NC-value is a modification of the C-value that considers the context of multi-word term and tries to find longer strings that appear more frequently in the corpus. The author also introduced a method to assign weight to different terms to create a list of “important” term context words (those that appear in the vicinity of terms in texts) from a set of terms extracted from a specialised corpus.

They are calculated based on the formulas below [22]:

$$Cvalue(a) = \begin{cases} \log_2|a| \cdot f(a) & a\ is\ not\ nested, \\ \log_2|a|\,(f(a) - \frac{1}{P(T_a)}\sum_{b\in T_a} f(b)) & otherwise \end{cases} \quad \text{2-9}$$

$$Weight(w) = \frac{t(w)}{n} \quad \text{2-10}$$

$$NCvalue(a) = 0.8 \times Cvalue(a) + 0.2 \times \sum_{b\in C_a} f_a(b) weight(b) \quad \text{2-11}$$

where

- a is the candidate string.
- $f(.)$ is its frequency of occurrence in the corpus.
- T_a is the set of extracted candidate terms that contain a.
- $P(T_a)$ is the number of these candidate terms.
- $f(b)$ is the total frequency of b in the corpus.
- w is the context word to be assigned a weight as a term context word.
- $Weight(w)$ the assigned weight to the word w.
- $t(w)$ the number of terms the word w appears with.
- n the total number of terms considered.
- $f_a(b)$ is the frequency of b as a term context word of a.

- 0.8 and 0.2 have been assigned by the author based on the result from a series of experiments.

Domain Relevance and Domain Consensus. As an improvement of the traditional contrastive analysis, which aims to filter out the irrelevant terms, *Navigli et al.* [51] introduced these new methods in the domain of ontology learning. The basic idea here is to compare the related statistics information in two different corpora: a relevant corpus based on the target domain, and a non-relevant corpus based on a different contrastive domain.

For a specific term t, its domain relevance is calculated as:

$$DR(t,k) = \frac{P(t \mid D_k)}{\sum_{i=1}^{m} P(t \mid D_i)} \quad \text{2-12}$$

$$Est(P(t \mid D_k)) = \frac{f_{t,k}}{\sum_{t' \in D_k} f_{t',k}} \quad \text{2-13}$$

where $P(t \mid D_k)$ and $P(t \mid D_i)$ are the probabilities of finding term t in the target domain D_k and the contrastive domain D_i.

The domain consensus is calculated as below, where $P_t(d)$ is the probability that document d includes t:

$$DC(t,k) = \sum_{d \in D_k} \left(P_t(d) \cdot log \frac{1}{P_t(d)} \right) \quad \text{2-14}$$

So far, we have introduced some of the most popular measures that have been widely adapted. There are other similar measures as summarised by *Asim et al.* [17], but all of these methods are solely based on statistics of the underlying corpora without considering the semantic and context.

It might be useful to point out that some of these methods are also used in other NLP tasks like term/keywords extraction and text summarisation. However, these tasks are not directly aligned with the main focus of this research, therefore additional discussion (about these tasks) will not be covered in this thesis.

One of the open challenges in OL study, which this research is aiming to address, is that the process heavily relies on pre-defined domain knowledge or human intervention to provide the necessary knowledge to make the decision (reason explained below).

When it comes to the concept extraction and selection process, the pre-defined knowledge and/or human intervention are mainly required to decide whether or not a specific concept should be included in the target ontology. In other words, to determine the closeness or significance or importance of a concept to the domain knowledge at a deeper semantic level.

DSM and related embedding methods could effectively produce a representation to make the words “computable”, and then there are various measures to re-weigh the individual word representations/vectors in the DSM. As discussed in the last section, the traditional count-based representation incorporates very little contextual information at a deeper semantic level. Hence, the predictive-based approach is also the direction of the mainstream DSM development, and many achievements have already been made in this area.

However, the measures discussed above are still count-based or statistics-based at a shallow semantic level (only contain very little semantic information).

In other words, we are leveraging various semantic representations with some measures that do not consider the context and do not operate at a deeper semantic level.

Moreover, these count-based measures are not always consistent with each other and often have other associated conditions. For example, in some scenarios, RTF could be a more suitable method than TF-IDF. Therefore, it is difficult to find a count-based approach that would precisely fit into all of the different scenarios. Rather, it would require humans, based on their own domain knowledge, to decide which measure needs to be applied and how. Chapter 3 will discuss these challenges and issues in detail. This section will only point out that these challenges clearly suggest that new measures need to be developed in order to align with the direction of DSM development.

In fact, since 2013 a few attempts have been made to build a predictive-based measure to support the concept selection process. For example, instead of using a count-based method, in [52] the authors used Word2Vec model to gather similar concepts from the corpus based on a seed-word list. Word2Vec itself is, indeed, a predictive-based approach. However, it is an approach that has been designed for DSM and word embedding. The actual measure used to compare the similarity between terms is cosine similarity, which is no more than just a simple vector operation.

Similar attempts have been made in using Word2Vec to support the OL process [53][54]. However, they all follow the same principle: provide a seed list first which contains key domain concepts, then use Word2Vec to compare its similarity with the other concepts. Hence, another concept with a high similarity

value could be considered as an important domain concept. In principle, they still use a seed-word based approach and are therefore not optimised methods as discussed at the beginning of this section.

Another major issue, which has not yet been discussed, is that Word2Vec is very sensitive to the initial corpus. With a limited number of documents in the corpus, it may not be possible to retrieve the similarity information for rare concepts even if these rare concepts have a significant influence on domain knowledge. One of the evaluations of this research is to compare SI with the Word2Vec. To summarise, the evaluation suggests that the semantic similarity relation between two concepts, in this case, Lymphoma and Non-Hodgkin Lymphoma, can only be discovered by Word2Vec if the corpus is specifically constructed for the cancer domain, which is different from the real target domain that we are aiming to process. Please refer to the related section in the Evaluation chapter for more details.

Since there are vast amounts of linguistic knowledge embedded in BERT as discussed above, it is reasonable to believe that BERT could be used to contribute to the overall OL process. Previous research suggests that these well pre-trained language models could act as a knowledge base to perform certain downstream NLP tasks. For example, in [19] the authors illustrated how to query neural language models for relational data by filling in masked tokens in the sequences, as Figure 2-10 shows below. This could be a useful approach to identify relations between concepts, which is another major task in the OL process, but it is difficult to see how it could contribute to the concept selection process.

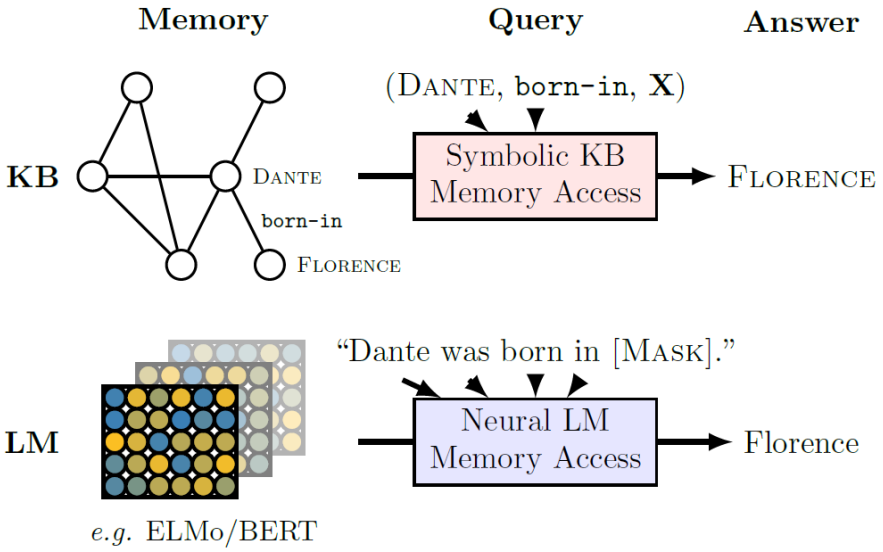


Figure 2-10 Querying knowledge bases and language models for factual knowledge [19]

Moreover, in recent years most of the research in the NLP field appears BERT related. BERT is indeed a very successful model, but people have already argued that the current NLP related research should not only focus on language modelling, as it will restrict the power of the pre-trained representations [55].

2.5 Summary

Within this chapter, we reviewed the history of Ontology Learning (OL) and highlighted that reliance on pre-defined domain knowledge and human intervention are the main issues or 'the bottleneck' in the field. Then we explained that Semantic Impact (SI) aims to provide a far more "advanced" measure to contribute to the domain-related concept selection process.

Since the SI is based on (and also an extension of) the Distributional Semantic Models (DSM), we have also reviewed the development history of DSM – from the count-based approach to the predictive-based approsach.

Finally, we have reviewed the other existing measures for concept selection. Overall, as with DSM and embedding development, people have tried to develop a predictive-based measure that could assess the relevance or the importance of a specific concept to the domain knowledge. The Word2Vec-based approach certainly operates at the semantic level, but it is a different version of the seed-word based approach. Hence it is still heavily reliant on pre-defined knowledge and human intervention. BERT seems to be able to reduce human intervention because of the vast embedded linguistic knowledge, but it focuses more on relational knowledge instead of contributing to the concept selection process.

In this research, we produce a new method called the Semantic Impact (SI) to further analyse (in a predictive way) and uncover the semantical and contextual information embedded in the DSM which is created by Word2Vec. This chapter briefly discussed the issues and challenges of using the existing count-based and predictive-based measures. The next chapter will continue this discussion on a more detailed level and proceed to explain how to use SI to overcome these challenges.

Chapter 3 Specific Problems to be Addressed

In the last chapter, we have briefly reviewed the development of Ontology Learning (OL) and related studies (DSM etc.). It has also been discussed multiple times that reliance on pre-defined domain knowledge (which we are trying to build an ontology for) and human intervention are two open challenges within this field. One of the main reasons is that the OL process itself has a limited ability to retrieve sufficient domain knowledge from the corpus. Therefore, the main challenge within this research is the fact that the OL process relies on extra information and human input to make the related decision.

This chapter will have a more in-depth discussion and explain why the concept selection process in OL needs more underpinning domain knowledge (compared with other generic NLP tasks), either pre-defined or manually inputted by the human being.

Overall, there are two main reasons: a) ambiguity around the definition of “word”, “term/entity name”, and “concept” — moreover, the way to vectorise various concepts. And b) existing measurements may not be objective and consistent at a deeper semantic level. This chapter will discuss them separately.

3.1 Word, Term/Entity Name and Concept

In order to help with the discussion, some basic terminologies need to be clarified first. This thesis uses free text as the data source, so “Word” will be the basic component. Every single word that appears in the corpus will add to a

vocabulary list to produce an associated universal Word2Vec module (which will be discussed in the next chapter).

Most of the corpus words are generic words that do not carry much information related to domain knowledge. However, some of them are used to represent a specific entity/individual. For example, the name of a specific winery (Bancroft, Beringer etc.) or the name of a country (France, Italy etc.) in the Wine Ontology discussed above. Such type of words is denoted as “Entity Name” in this thesis.

In addition to the entity names, another group of words represents abstract ideas instead of concrete individuals, e.g. `People`, `Organisation` and `Winery`. It is denoted as “Concept” in this thesis, although it may differ from how psychology defines it.

It is easy to understand that entity names are more likely associated with ontology instances, and concepts are linked with ontology classes. Hence, it is the concepts that need to be extracted from the corpus for the concept selection purpose, not the entity names. With this idea in mind, there are several challenges here.

The first challenge within this area is the vague boundary between entity name and concept. It is not only because of the ambiguity of an individual word, which is a common challenge in NLP related research (e.g. the word “apple” can either be an entity name that associated with the “fruit” concept or a concept to distinguish iOS from Google’s Android system in the smartphone domain), but also because of the ontological diversity which will be discussed in the next section.

As a result, even within the same domain, a specific word can be classified differently depending on context and application. Hence, it usually requires extra domain knowledge to distinguish entity names from concepts and identify what concept they belong to. This links to the following challenge.

The next challenge within this area is how to generate a single vector to represent the collection of individual word vectors that preserve the semantic meaning of the concept in a high-dimension space. With an appropriate Distributional Semantic Models (DSM) approach discussed in the last chapter, words in the corpus (regardless of whether they are entities or concepts) can be transformed (vectorised) into numerical representations. Supposing that there are three words in a Word2Vec model associated with the "Wizard" concept: `Harry_Potter` (using the "_" character to concatenate Harry and Potter into one word), `Voldemort` and `Wizard`, where $\vec{V}_H$ in Figure 3-1 is the vector for the word "Harry_Potter", $\vec{V}_V$ is the vector for "Voldemort" and $\vec{V}_W$ is the vector for the word "Wizard" (the order of these vectors in the model does not matter).

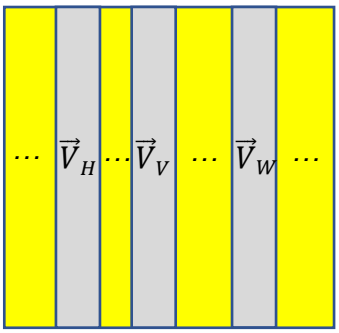


Figure 3-1 Example of a Word2Vec model with Harry_Potter, Voldemort and Wizard. The yellow square represents the Word2Vec model, and the grey rectangles represent the vectors of individual words.

From the Word2Vec perspective, all words in its vocabulary should be equally important regardless of whether they are concepts or not. As part of the concept

selection process, the system will, at some point, start assessing the `Wizard` concept to determine if it is a relevant concept. To do so, a vectorised representation (of the concept) is needed to act as the input for further processing, and the question here is how to generate this vector representation.

The most direct approach is simply using $\vec{V}_W$ as the representation of the "Wizard" concept, and use various methods to analyse its context and identify the semantic distribution pattern. However, $\vec{V}_W$ itself is a representation of `Wizard` as an individual word in the vocabulary instead of a concept. The concept of `Wizard` should be a combination of all the associated words (which be considered as entities), in this case, `Harry_Potter`, `Voldemort` and `Wizard`. Therefore its representation should be a combination of $\vec{V}_H$, $\vec{V}_V$ and $\vec{V}_W$. Just because one of the associated words has the same name as the concept does not mean the representation of that specific word can also be used to represent the concept. How to effectively combine $\vec{V}_H$, $\vec{V}_V$ and $\vec{V}_W$ without losing the existing semantic information embedding in these high-dimension spaces is another challenge here.

Within this thesis, this specific problem/challenge is handled by using a word-replacement process discussed in Section 4.2.2.

3.2 Objective and Consistent Measurement at a Deeper Semantic Level

As introduced in the previous chapter, there are various ways to measure how important (or relevant) a word is with respect to a document in the corpus. However, the importance or relevance of a word to a document is not quite the same as the importance or relevance of a concept to the domain knowledge.

Using TF-IDF and the Harry Potter (mentioned in the previous chapter) as an example, even if we can solve the problem that, in fact, a concept, such as `Wizard`, is implied by multiple words (e.g. Harry_Potter[14], Voldemort etc.), it is still difficult to reach a high *tf-idf* weight to compute its relevance to the corpus, simply because it is almost guaranteed that this concept will exist in every chapter/document about Harry Potter and therefore will have a low, if not 0, *idf* value which suggests that it is not very informative at all. Moreover, statistical approaches like TF-IDF never take into consideration the position of a word in the sentence and completely ignore its context. Therefore, it is reasonable to declare that such methods do not operate at a deeper semantic level.

The predictive-based approach discussed in the previous chapter was supposed to operate at a deeper semantic level. For example, people can train a neural network to distinguish important and unimportant concepts.

One of the most direct ways to do so is by labelling the training dataset (various concepts extracted from the corpus) with “important” or “unimportant” and then designing a neural network for learning the pattern. Hopefully, the neural network will be able to handle the rest of the work at a deeper semantic level. Alternatively, people can use some unsupervised learning methods to automatically place concepts in different groups (e.g. important, less-important and unimportant). These traditional neural network based approaches may address the issue at a deeper semantic level, but they are not consistent and objective.

[14] In this case, Harry_Potter is a proper noun identified by the NER tool.

For example, people will have a different view on whether a concept is important or not. Hence, they may end up with a very different labelling result. For the unsupervised learning approach, again, people may have different views on how many groups they need to create initially (which will have a direct impact on the result). Also, there is no way to compare the importance level of the concepts within the same group (each concept will have a probability value, but a high probability value does not mean it is more important than those with a lower probability value).

These challenges lead to the need to develop a new approach to a) make the concept selection process less reliant on pre-defined knowledge and human intervention, and b) use a predictive-based approach to measure concepts objectively and consistently. The solution proposed here is called Semantic Impact (SI), and we will start a detailed introduction in the next chapter.

Chapter 4 Semantic Impact

Now that we have explored various challenges (and specific problems to be addressed) within the concept selection process, this chapter will provide a detailed discussion of the proposed solution – Semantic Impact (SI), a novel approach to derive a numerical measure that summarises how strongly a concept impinges on the domain of discourse.

More specifically, by taking into consideration the semantic representation of a concept that appears in documents and its connectivity with other concepts in the same document corpus, SI measures the importance of a concept with respect to the knowledge domain at a semantic level. Here, the "semantic" importance of a concept is two-fold. Firstly, the concept needs to be informative. Secondly, it should be well connected (strong correlation) with other concepts in the same domain. There is a full mathematical definition towards the end of this chapter (Equation 4-10, p. 104).

In order to produce the SI value, two contributing ideas need to be introduced first: the Informative Coefficient (IC) and the Connectivity Coefficient (CC). As suggested by the name, the IC is a value that represents the semantic richness a concept (identified from the corpus) has within the domain. Furthermore, the CC is a value that measures how strongly it is connected (or correlated) with the other corpus concepts. Here is an intuitive explanation: to be considered as "important", a concept must be a) meaningful and contain sufficient information about the domain knowledge, and b) well connected (strong correlated) with the other concepts within the domain to be able to have an influence on the domain knowledge.

The SI value is a simple combination of the IC and CC value, although the process to calculate them is extremely complex and time-consuming.

Let $IC_{<ConceptName>}$ be the *informative coefficient* for a specific concept, $CC_{<ConceptName>}$ be the *connectivity coefficient* the specific concept has with the other concepts. λ_1 is a constant that can be used to adjust the weight of the $IC_{<ConceptName>}$. Subsequently, λ_2 is a constant that adjusts the weight of the $CC_{<ConceptName>}$. A more detailed explanation about λ_1 and λ_2 is given at the end of Section 4.5. Then the Semantic Impact value ($SI_{<ConceptName>}$) of a specific concept can be calculated as follows:

$$SI_{<ConceptName>} = \lambda_1 Normalised(IC_{<ConceptName>}) + \lambda_2 Normalised(CC_{<ConceptName>}) \quad 4\text{-}1$$

where $\lambda_1 + \lambda_1 = 1$, $\lambda_1 \in [0,1]$, $\lambda_2 \in [0,1]$, $IC \in [-1,1]$, $IC \in [0,n]$, n is the number of the class pairs (refer to Section 4.4 for more details), $SI_{<ConceptName>} \in [-1,1]$. The normalisation function will be discussed in Section 4.5.

The rest of this chapter will provide a detailed discussion on calculating the IC and CC value.

4.1 Overall Architecture

Essentially, IC is calculated by leveraging the overfitting mechanism of a series of neural networks to distinguish informative concepts from non-informative concepts; CC is obtained by assessing the correlation strength between each

concept with the Maximal Information Coefficient (MIC) algorithm [10]. To achieve these ends, the SI algorithm is divided into four steps, as shown in Figure 4-1 below.

In fact, there is another step before step 1 to build two corpora by randomly splitting all the collected documents into two groups with a 10% overlap. For convenience, one of them is denoted as "Source Corpus" and the other is called "Target Corpus". Both of them refer to the same domain[15], although the word "source" and "target" may imply they (source corpus and target corpus) are different.

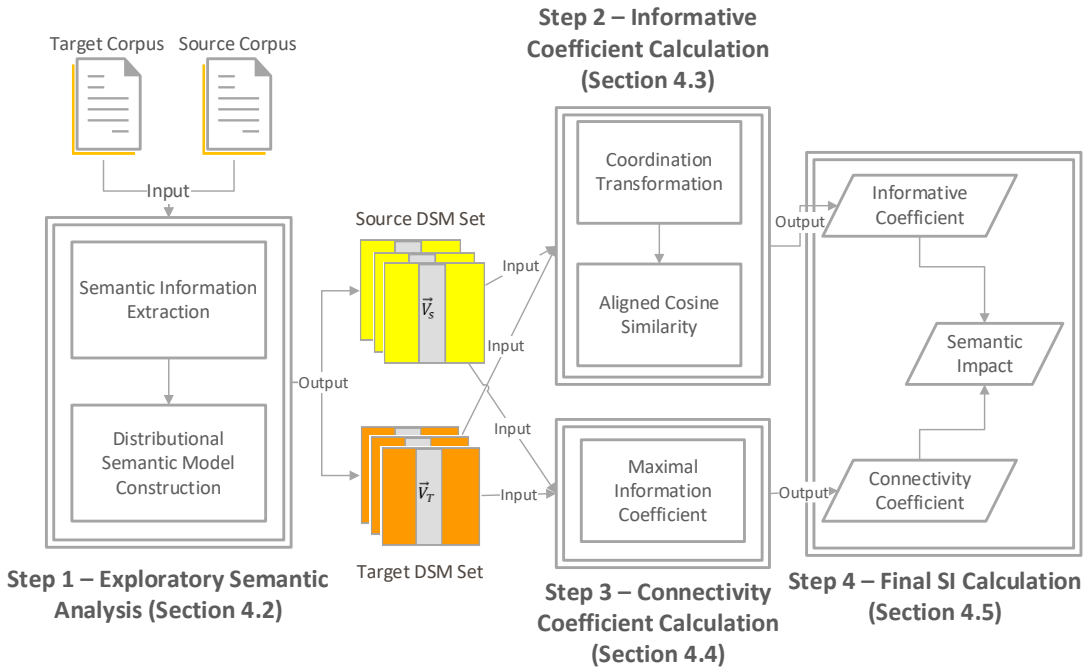


Figure 4-1 Process Overview. Four steps to calculate the Semantic Impact value. Step 1 is the Exploratory Semantic Analysis (ESA) process, which aims to extract various semantic information from the Source and Target corpus and then build Distributional Semantic Models. Detailed information is discussed in Section 4.2. Step 2 is used to calculate the Informative Coefficient and is discussed in Section 4.3. Step 3 is for Connectivity Coefficient Calculation and is discussed in Section 4.4.

[15] It is possible to make the Source Corpus and Target Corpus focus on two difficult but closely related domain, or two different perspectives of the same domain. An example will be the Harry Potter main story and its prequel. A more detailed discussion is given in Section 8.2.5.

Step 4 is how to merge the Informative Coefficient and the Connectivity Coefficient to produce the final Semantic Impact value, and is discussed in Section 4.5.

As explained already in Chapter 2, the idea of Distributional Semantic Models (DSM) [9] is that the meaning of words can (at least to a certain extent) be inferred from their usage. Therefore, the semantics can be encapsulated in high-dimensional vectors based on the nearby co-occurrence of words.

By adopting and expanding the DSM theory, this research is based on two assumptions: a) a high-dimensional vector can be used to infer the semantic representation of a concept, which extensionally is a set of words that belong to the same semantic group, and b) with sufficient data, for any concept in a domain, the distribution of its semantic representation is consistent.

There will be a whole section in this thesis (Section 4.3) explaining how to measure informativeness by using the consistency of the semantic representations. It is, in fact, considered as one of the most important contributions in this thesis. This section will provide a brief discussion to help readers better understand the overall process of the SI.

As will be explained in the next section, the semantic distribution of a specific concept is represented by the associated vector (by using a word-replacement process discussed in Section 4.2.2) that is produced by the Word2Vec method. Since there are two separate corpora (Source and Target Corpus) about the same domain, it is easy to generate two vectors (for a specific concept), one from the source corpus and the other from the target corpus. Based on the second assumption, these two vectors should be the same (the cosine similarity between those vectors should be 1) in an ideal situation. However, as

discussed in Section 4.3.2, there are some randomisations in the Word2Vec (or any word embedding) method. As a result, the numerical value of the vectors could be shifted from run to run and end up falling into different coordinate systems. Moreover, the randomisation of the Word2Vec process itself is not the only reason those vectors fall into different coordinate systems. A deeper issue is because the Source Corpus and Target Corpus contain different documents. As a consequence, the cosine similarity between those two vectors cannot be calculated directly, and therefore we cannot easily measure or observe the consistency of the semantic distribution.

Hence, we have designed a neural network-based approach (Section 4.3.2 and Section 4.3.3) to align those different coordinate systems. An interesting phenomenon discovered in this thesis is that concepts' informativeness has an impact on the overfitting of the neural network(s). This means that the more informative a concept is, the less possible the neural network will overfit itself (reasons will be explained later). In other words, the more informative a concept is, the more accurate its predicted value (of the semantic distribution) will be, and as a consequence, the more close to 1 the cosine similarity (between the predicted value and the "observed" value) will be (more information is given in Section 4.3.2 and Section 4.3.3). Subsequently, by assessing how close to 1 the cosine similarity is (it is, in fact, aligned cosine similarity which will be explained later), we can measure how informative a concept is. Essentially, this is how we are going to produce the Informative Coefficient (IC) (Step 2) in this thesis.

It is also possible to use the semantic distribution of a specific concept to measure the impact or influence that a particular word (or a list of words) could bring to this concept itself. By doing so for all the domain concepts on all the words in the corpora, the system will then be able to measure the correlation between each concept pair to generate the CC value (Step 3).

The following sections will discuss these steps in detail.

4.2 Step 1 (Figure 4-1) - Exploratory Semantic Analysis (ESA)

There are two experiments conducted in this thesis. The actual implementations of Step 1 in each is slightly different. For example, in the first experiment, which is about “Donald Trump” in the News domain, the Semantic Information Extraction has been done by the IBM Natural Language Understanding (NLU) service and the BBC Core Concepts Ontology; while the second experiment is about the various diseases in the Candida domain, and uses PubTator as the tool to extract entities and concepts.

This chapter only focuses on the overall design and explains why it has been designed in this way. The detailed implementation will be discussed in the later chapters.

In general, ESA (Step 1) aims to extract various semantic information from the Source and Target corpus and then build associated Distributional Semantic Models (DSMs). This process can be further split into two subprocesses: a) identify various entities and concepts within the corpus together with their relationship; b) generate a separate DSM for each individual concept identified from a) in both Source Corpus and Target Corpus, together with two Universal

DSMs (`W2V_Universal_Source` and `W2V_Universal_Target`), which will be discussed later.

4.2.1 Semantic Information Extraction

This subprocess aims to identify all the entities and concepts within the Source and Target Corpus (by using an existing NER tool as mentioned in Chapter 2) and then convert them into a lightweight ontology [16] format called the Document-based Ontology (DbO). As introduced in [56], DbO operates on the document level without concern for the broader context.

One of the open challenges discussed in Chapter 2 (and that this research aims to address) is to reduce the reliance on pre-defined knowledge of the domain that people are trying to build an ontology for. Within this research, we fully accept that knowledge about a specific domain must come from somewhere. However, instead of using pre-defined knowledge of that specific domain (or a specific perspective of the domain), this research proposes a way to "transfer" the required knowledge from a related domain (or a related perspective of the same domain). This is achieved by adapting an existing ontology that is closely related to the domain (or describes the domain from a different perspective). This ontology is denoted as the "Guiding Ontology", and there is an extra mapping process to link the corpus concepts with the classes defined in it.

[16] Concepts are connected by general associations rather than strict formal connections.

Hence, there are three tasks within this subprocess, as shown in Figure 4-2 below (the "End" of the process will be the beginning of the DSM Construction shown in Figure 4-1).

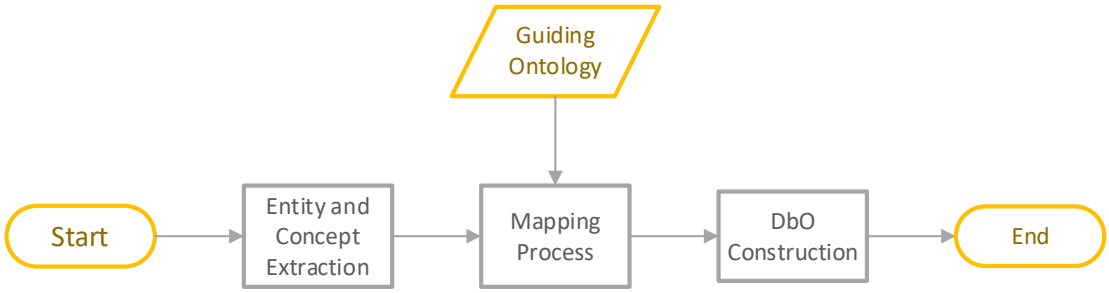


Figure 4-2 Tasks in the Semantic Information Extraction Process

Each document in the corpora will pass through these tasks individually. Supposing a document only contains the following text:

"*Hogwarts is a school of Witchcraft and Wizardry at Scotland and was*
[Organisation] [location]

founded by Godric Gryffindor, Helga Hufflepuff, Rowena Ravenclaw, and
[people] [people] [people]

Salazar Slytherin."
[people]

The entity and concept extraction task will be used to identify that "Hogwarts" is a type of `Organisation`; "Godric Gryffindor", "Helga Hufflepuff", "Rowena Ravenclaw" and "Salazar Slytherin" are `people`, and "Scotland" is a type of `location`. These different types usually are pre-defined classes within the selected NER method. Since the guiding ontology is closely related to the current domain, it is very likely that some of these types will have an equivalent in the guiding ontology, although the actual definition might be different. In other words, they may have a different name in the guiding ontology. For example, "`organisation`" can be called "`school`" in the guiding ontology.

Hence, after the entity and concept extraction task, there is a mapping process to map the corpus concepts (`organisation`, `people` and `location`) to the guiding ontology concepts, which is an ontology we use to generate the benchmark (best neural network structure) for further analysis. More details are discussed in Section 4.3.3.

The system will then generate the associated DbO based on the mapping information. For instance, if a corpus concept has a mapped ontology class (e.g. `organisation` is mapped to `school`), then the system will automatically inherit the properties and relations that are defined in the guiding ontology and use this inherited information to construct the DbO. If a mapping does not exist, the system produces a new empty `Class` and adds it into the DbO.

Finally, by going through all the documents in the Source and Target Corpus, the system generates an individual DbO for each document and groups them into two sets: Source DbO Set and Target DbO Set. The reason to convert the extracted information into the DbO format is that it is more convenient to analyse the ontological relations between each of the concepts using the DbO.

4.2.2 Distributional Semantic Models (DSMs) Construction

The second subprocess in Step 1 is DSM(s) construction. Based on the extraction results, it is easy to get a list of shared concepts (by cross-referencing two DbO Sets discussed above) between two corpora, together with their associated entity names. A separate DSM can then be built (by using Word2Vec) for each of the shared concepts, together with their vectorised representation. Only shared concepts are selected here because the SI algorithm needs to cross-compare the semantic representation information of

the same concept between two corpora in order to identify the patterns of distribution.

The semantic distribution of a concept is, in fact, represented by a vector obtained from the vectorisation process, which is handled by the Word2Vec approach. It is easy to get the semantic distribution for any single word in the corpus. However, as discussed in the previous chapter, a concept contains multiple words. There are many ways to “merge” multiple vectors into one. For example, simply average individual word vectors for a collection of words to produce a single vector, but such a primitive mathematic operation may lead to a change of the semantic representation which is embedded in those vectors. As a result, this newly created single vector may not be able to accurately reflect the semantic information of the concept itself, and therefore there is no guarantee that the semantic distribution of a specific concept is consistent between the Source and Target Corpus. In other words, the Neural Complex approach (discussed in Section 4.3.3.1) may not be able to generate an accurate Informative Coefficient (IC) value. So, the challenge here is to generate a single vector to represent a collection of individual word vectors that preserve the semantic meaning of the concept in a high-dimension space.

This is achieved by a word-replacement process. The rationale is to replace all the relevant entities (keywords) of a specific concept (based on the NER extraction result, for example the concept of `Wizard` includes keywords like `Harry_Potter` and `Voldemort`) from the relevant corpus (the relevant entities may differ between Source and Target Corpus, as the example shown in Figure 4-3) with a unique string (an invented string that is different from any

word) and re-run the vectorisation process to generate a new Word2Vec model for this specific concept. Then the vector of this invented unique string could be considered as a projection of all the vectors of the replaced words on this newly created Word2Vec model and considered to be tantamount to a semantic distribution vector for the original concept (denoted as $\vec{V}_{<ConceptName>\_Source}$ or $\vec{V}_{<ConceptName>\_Target}$).

By repeating this process, the system will generate a separate Word2Vec model for all the concepts in both Source and Target Corpora respectively. Moreover, the system will create two additional Word2Vec models by using the original text from Source and Target Corpus without replacing anything.

The Word2Vec models created via this word-replacement process are denoted as `W2V_<ConceptName>_Source` and `W2V_<ConceptName>_Target`, and the Word2Vec models generated from the original corpora (without replacing any words) are denoted as `W2V_Universal_Source` and `W2V_Universal_Target`.

The Source DSM Set shown in Figure 4-1 contains all the `W2V_<ConceptName>_Source` models and the `W2V_Universal_Source` model. Correspondingly, the Target DSM consists of all the `W2V_<ConceptName>_Target` models and the `W2V_Universal_Target` model.

There is a good reason to generate separate models for the different concepts instead of replacing all the relevant words from the corpus with all the invented unique strings in one go. It is because by the nature of how Word2Vec (or any

word embedding method) works, replacing too many words may significantly change the grouping structure, and therefore the new model will not be able to represent the same semantic distribution as the old model does. Hence, it is essential to minimise the number of words that need to be replaced in each model in order to maximise the consistency of the semantic representation between different models (which is key to the success of the IC calculation discussed in the next section).

Figure 4-3 is an illustrated example of this process. Using the Hogwarts example used above. Assume that “Hogwarts is a school of Witchcraft and Wizardry at Scotland and was founded by Godric Gryffindor, Helga Hufflepuff, Rowena Ravenclaw, and Salazar Slytherin.” is the only content in the target corpus. By going through the semantic information extraction process, “Hogwarts” has been identified as a keyword or entity name of the `organisation` concept, “Godric Gryffindor”, “Helga Hufflepuff”, “Rowena Ravenclaw” and “Paul Allen” belong to the `people` concept. In the source corpus, the only sentence is “Durmstrang is a school for young witches and wizards, and was funded by Nerida Vulchanova.”. Here “Durmstrang” belongs to `organisation`, and “Nerida Vulchanova” belongs to `people`.

Then the word-replacement process will start from the source corpus and use the unmodified text as the input to generate the `W2V_Universal_Source` model, as step 1.1 shows. Then step 1.2 will replace the word “Durmstrang” in the original text with an invented unique string “xoxovvOrganisationvvoxox” and use the modified text as the input to generate a new Word2Vec model – `W2V_Organisation_Source`. In this case, the vector of the word

“xoxovvOrganisationvvoxox” will be the semantic representation for the `organisation` concept ($\vec{V}_{Organisation\_Source}$). In this simplified example, $\vec{V}_{Organisation\_Source}$ and $\vec{V}_{Durmstrang\_Source}$ are the same due to the fact that there is only one entity (keyword) replaced. However, in practice, more entities (keywords) will be replaced from the corpus.

Using the same method, step 1.3 will replace the entity covered by the `people` concept—“Nerida Vulchanova” with “xoxovvPeoplevvoxox” to generate `W2V_People_Source` and the associated $\vec{V}_{People\_Source}$.

Since there is no `location` concept identified within the source corpus, the system will move to the target corpus and follow the same principle to produce `W2V_Universal_Target` (step 1.4), `W2V_Organisation_Target` (step 1.5) and `W2V_People_Source` (step 1.6).

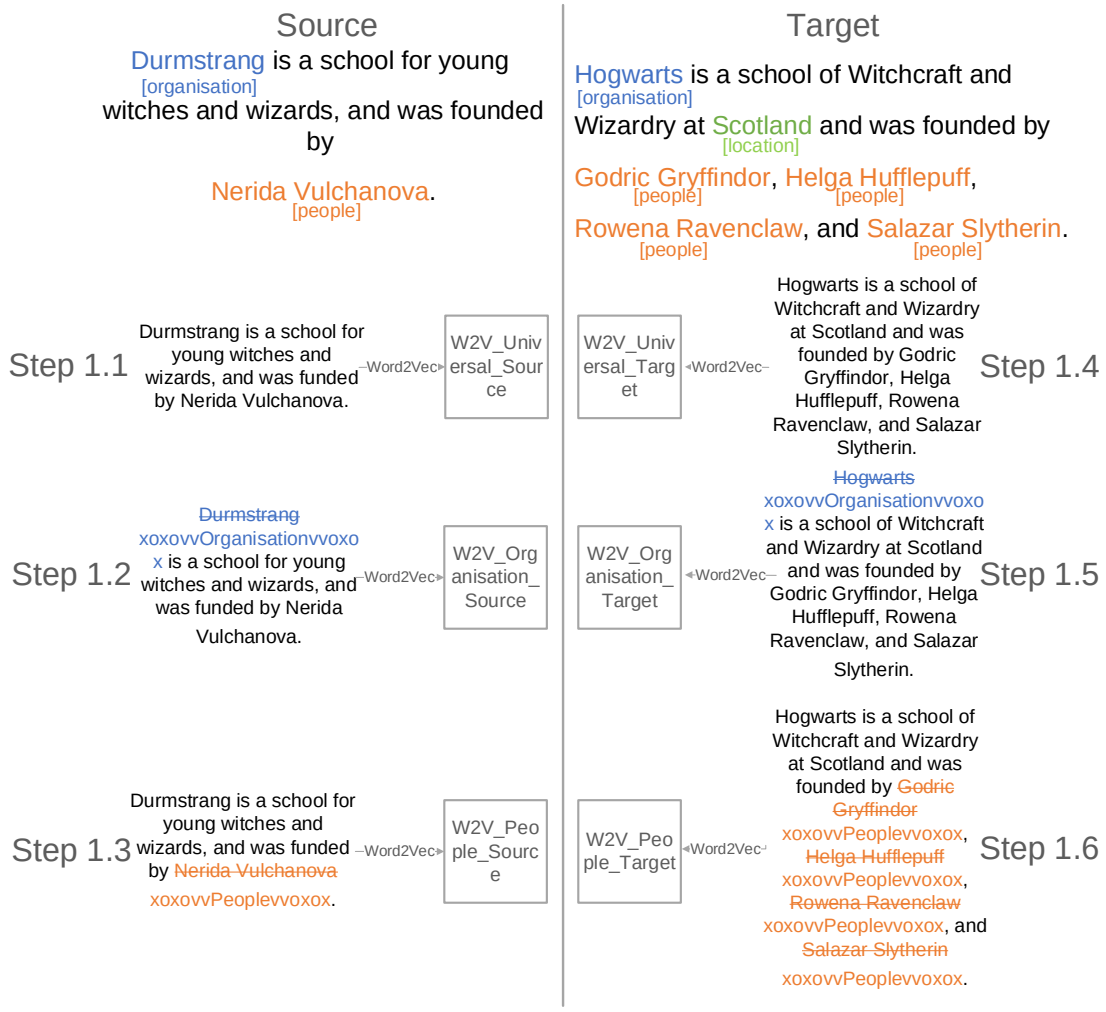


Figure 4-3 Example of the word-replacement process. Source corpus is on the left side, and Target corpus is on the right side. There are six steps in total in this example to generate the required Word2Vec models. Please refer to the above discussion for more information.

4.3 Step 2 (Figure 4-1) - Informative Coefficient Calculation

In this section, a new predictive-based approach will be introduced to consistently and objectively measure how informative a concept is within the given domain. This innovative approach is one of the main contributions of this thesis.

4.3.1 Basic Philosophy

By going through the Exploratory Semantic Analysis (ESA) step (Section 4.2), the system will generate a Word2Vec model for each individual concept in the

Source and Target Corpus (`W2V_<ConceptName>_Source/Target`). Each model contains a single vector to represent the semantic distribution of an associated concept ($\vec{V}_{<ConceptName>\_Source/Target}$).

As mentioned in Section 4.1, both the Source and Target Corpus have the same amount of documents (and with a 10% overlapping) about the same topic, then based on the two assumptions discussed in Section 4.1, the same concept should have the same semantic distribution within the Source and Target Corpus. Following the example shown in Figure 4-3, $\vec{V}_{Organisation\_Source}$, which is included in the `W2V_Organisation_Source` model calculated in step 1.2, should be "equal" to $\vec{V}_{Organisation\_Target}$ which is calculated in step 1.5; and similarly, $\vec{V}_{People\_Source}$ should be "equal" to $\vec{V}_{People\_Target}$.

There is a standard way to measure the closeness of two words within a Word2Vec model – Cosine Similarity (CS), which is calculated by the following formula:

$$CS = \frac{\vec{V}_1 \cdot \vec{V}_2}{\|\vec{V}_1\|\|\vec{V}_2\|} \qquad 4\text{-}2$$

where $\vec{V}_1$ is the vector of the first word and $\vec{V}_2$ is the vector of the second word. The range of CS is from -1 (exactly opposite) to 1 (exactly the same). Hence, the word "equal" used above means the CS value between two vectors is equal or close to 1 instead of being identical vectors.

So, the overall strategy is to find a way to use the CS value between $\vec{V}_{<ConceptName>\_Source}$ and $\vec{V}_{<ConceptName>\_Target}$ to represent how informative a

concept is. Within SI, this is achieved by using a new concept coined in this research -- the Neural Complex (NC), which is discussed in Section 4.3.3.1 and Section 4.3.3.2.

Essentially, those informative concepts should have a more complex semantic representation, and therefore should have a more stable and more consistent distribution across the Source and Target Corpus ($CS = 1$). The Neural Complex is basically designed to measure this consistency, so the more the aligned CS value is close to 1, the more informative the concept is.

However, it is essential to use a coordinate transformation process to place both vectors into the same coordinate system first to produce the aligned CS to measure informativeness.

4.3.2 Coordinate Transformation (CT) Process

The CS approach mentioned in the previous section only works if those two words are within the same Word2Vec model. Otherwise, the CS value is meaningless since those words are in two different coordinate systems.

$\vec{V}_{<ConceptName>\_Source}$ and $\vec{V}_{<ConceptName>\_Target}$ are from two different Word2Vec models based on two corpora. Hence, the CS value between them cannot be calculated directly.

However, it can be argued that since the Source and Target Corpus are about the same domain, the distributional semantic information represented within the `W2V_<ConceptName>_Source` and `W2V_<ConceptName>_Target` models should ideally be the same. As a result, if there are sufficient documents

within the Source and Target Corpus, then they should share the same (or very similar) coordinate system – for a specific word that appears in both corpora, its vector representations in the two models should be the same.

In practice, this is not quite the case. With sufficient training data (documents within the corpora), these two Word2Vec models should have the same (or very similar) distributional semantic information for the shared words (this is also one of the assumptions discussed in Section 4.1). However, the same distributional semantic information means the distances and directions of a specific word from the other words co-trained inside the same model should be the same, instead of having the same numerical vector value.

This is caused by the randomisation behaviours within the Word2Vec training process [57]. For example, as discussed in the original Word2Vec paper's algorithm description, the training windows are randomly truncated as an efficient way of weighting nearer words higher, and the negative examples in the default negative-sampling mode are chosen randomly. Even when all this randomness comes from a fixed seed to give a reproducible stream of random numbers, the usual case of multi-threaded training can further change the exact training-order of text examples, and thus the result in the final model. Hence, even trained on the same corpus, the model could be different and the numerical value of the vectors could be shifted from run to run and end up falling into different coordinate systems.

To address this issue, a Coordinate Transformation (CT) process has been designed to align different Word2Vec models and make the vectors between

them comparable to each other. The key idea here is anchoring on common words appearing in both models.

Using Figure 4-4 as an example, $X_1Y_1Z_1$ and $X_2Y_2Z_2$ are two Word2Vec models. Both have the words “Word_A” and “Word_B”. “Word_C” is a unique word within the former, and “Word_D” is a unique word within the latter. Let $\vec{V}_1^A$ and $\vec{V}_1^B$ be the vectors of the word “Word_A” and “Word_B” in the first model respectively, $\vec{V}_2^A$ and $\vec{V}_2^B$ be the corresponding vectors in the second model. $\vec{V}_1^C$ and $\vec{V}_2^D$ are the vectors for the two unique words in the associated model. By default, there is no direct way to calculate the CS value between $\vec{V}_1^C$ and $\vec{V}_2^D$ since they are in two different models. Hence, we need to align the $X_2Y_2Z_2$ model with the $X_1Y_1Z_1$ model, and the goal is to make the shared words ($\vec{V}_1^A$ & $\vec{V}_2^A$ and $\vec{V}_1^B$ & $\vec{V}_2^B$) as close to each other as possible by maximising the following quantity: $\frac{\vec{V}_1^A \cdot \vec{V}_2^A}{\|\vec{V}_1^A\|\|\vec{V}_2^A\|} + \frac{\vec{V}_1^B \cdot \vec{V}_2^B}{\|\vec{V}_1^B\|\|\vec{V}_2^B\|}$

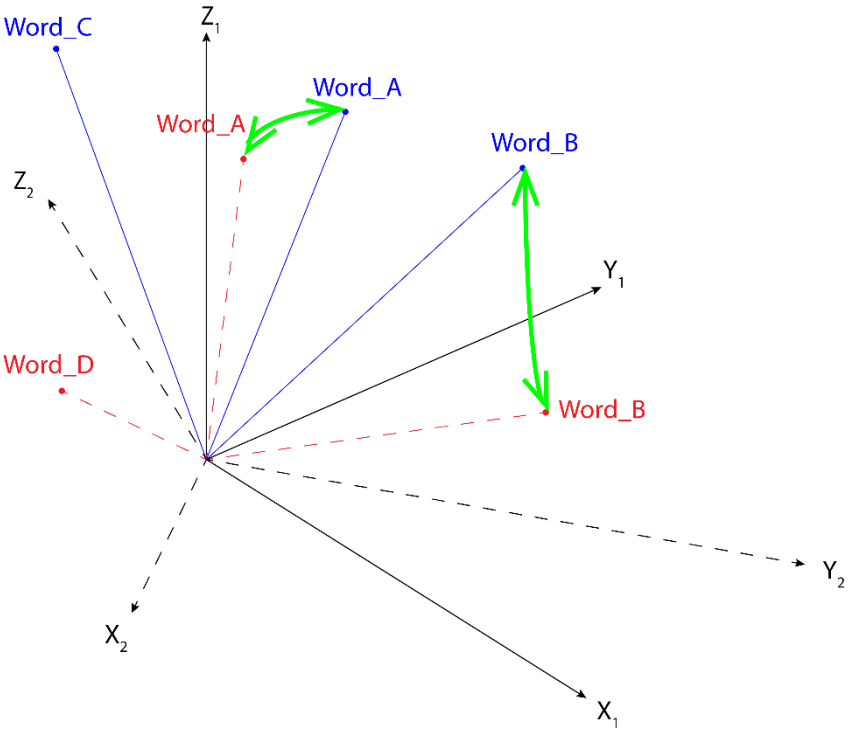


Figure 4-4 Coordinate Transformation Example

This thesis simplifies the solution to the above problem to a classic supervised learning problem with a neural network. As Table 4-1 is shown below, using $\vec{V}_2^A$ and $\vec{V}_2^B$ from the $X_2Y_2Z_2$ as the input of the neural network, and $\vec{V}_1^A$ and $\vec{V}_1^B$ as their associated label, it is then possible to construct a neural network, as shown in Figure 4-5, to automatically align the $X_2Y_2Z_2$ model with the $X_1Y_1Z_1$model. Moreover, it is also possible to use $\vec{V}_2^D$ as the input of this neural network to predict its value in the $X_1Y_1Z_1$model (since the $X_1Y_1Z_1$model does not contain the “Word_D”). In other words, project the vector representation of Word_D from the $X_2Y_2Z_2$ model to the $X_1Y_1Z_1$model to produce $\vec{V}_1^D$. Then the CS between $\vec{V}_1^C$ and $\vec{V}_2^D$ would be the equivalent of the CS between $\vec{V}_1^C$ and $\vec{V}_1^D$, which can be calculated directly. Calculating the cosine similarity between Word_C and Word_D may look strange, but it will make sense after introducing the Neural Complex idea in the next section. The point here is that we can

calculate the cosine similarity for words that may not exist in the current coordinate system after a CT process.

Input	Label
$\vec{V}_2^A$	$\vec{V}_1^A$
$\vec{V}_2^B$	$\vec{V}_1^B$

Table 4-1 Training Set Example

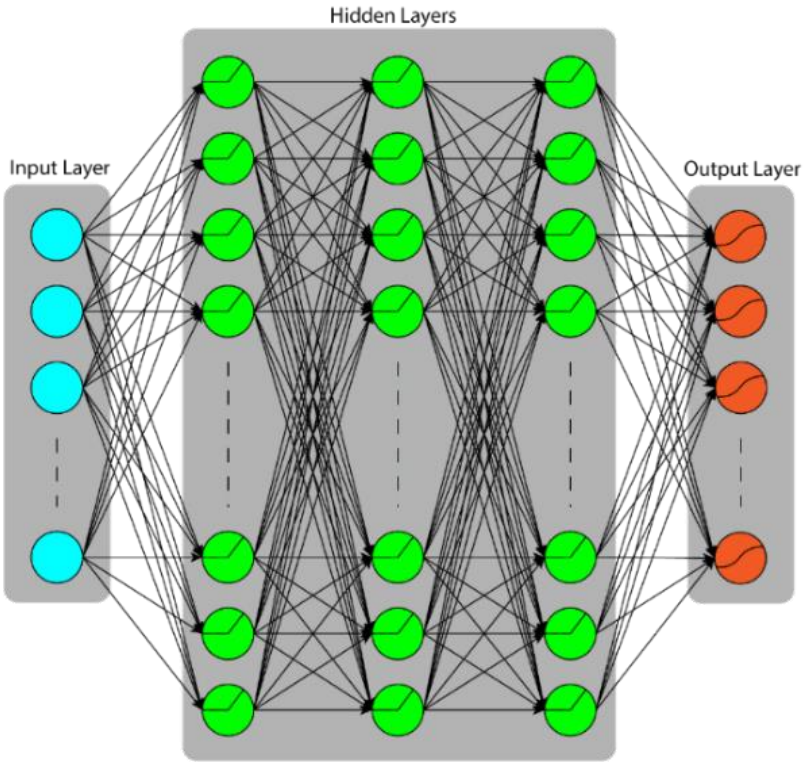


Figure 4-5 Neural Network for the CT Process

There are two experiments conducted for research, and each has a different implementation of the neural network in Figure 4-5 (e.g. different in the number of the hidden layers and the nodes). More detailed information, together with the way to evaluate the neural network, will be provided within the related chapters that focus on those experiments (Chapter 5 and Chapter 6). However, some shared (or unchanged) configurations are:

- Use of Cosine Similarity as the overall loss function, since the goal is to ensure the input vector and label vector as “close” to each other as

possible, which is represented by the CS value instead of numerical vector value as discussed above, between the input layer and the output layer.

- Use of Tanh as the activation function on the Output Layer to scale the output to between -1 and 1.
- Use of Rectified Linear Unit (ReLU), a popular activation function in Deep Learning study [58] as the activation function on the hidden layers.
- Use of XAVIER, a method introduced in [59] for the weight initialisation.
- Use of ADAM, a method introduced in [60],as the method for stochastic optimisation.
- BatchSize set to 100.

In theory, this proposed CT process can be used to align any Word2Vec model pairs. However, in order to ensure the success of this process, there are two conditions here. Firstly, intuitively, the semantic distribution information represented by these two models should be similar (e.g. created from a similar corpus). Secondly, to ensure there are sufficient training data, there should be a large number of shared words between the vocabularies within these models.

In the next section, a more detailed discussion will be provided to explain how to embed this CT process into a much wider process to make the CS value (in fact, aligned CS value) between $\vec{V}_{<ConceptName>\_Source}$ and $\vec{V}_{<ConceptName>\_Target}$ be able to represent how informative a concept is.

4.3.3 Neural Complex and the Implementation Plan (Training Method)

There are four parts included in this section. The first part (Section 4.3.3.1) focuses on the introduction of the Neural Complex (NC), which aims to calculate the aligned CS value (denoted as CS') between $\vec{V}_{<ConceptName>\_Source}$ and $\vec{V}_{<ConceptName>\_Target}$. Then use the CS' to calculate concepts' informativeness, which is the Informative Coefficient (IC).

NC is a new concept coined in this research. A more detailed introduction will be provided within the first part of this section, but intuitively it can be considered as a way to "polymerise" multiple neural networks and combine them into a higher-order neural network structure, where each node on the hidden layer is another independent neural network.

The second part (Section 4.3.3.2) of this section focuses on the training process of the NC. Section 4.3.3.3 will further discuss why the Mapped Subset (Step A, Figure 4-14) can be used as the training data and why a consistent NN structure is required during the NC training process.

The last part (Section 4.3.3.4) will provide a quick summary of the NC and the training process.

4.3.3.1 Neural Complex (NC) and the Aligned Cosine Similarity Calculation

As Equation 4-1 introduced at the beginning of this chapter, the Semantic Impact (SI) is a combination of the Informative Coefficient and the Connectivity Coefficient. Overall, the Informative Coefficient (IC) for a specific concept is calculated as:

$$IC_{<ConceptName>} = CS'_{<ConceptName>} \times \overline{Conf}_{<ConceptName>} \quad \text{4-3}$$

where $CS'_{<ConceptName>}$ is the Aligned Cosine Similarity value, as explained in the last section, vectors within two different Word2Vec models cannot be calculated directly, and therefore need to go through the CT process to align the two models first, between $\vec{V}_{<ConceptName>\_Source}$ and $\vec{V}_{<ConceptName>\_Target}$, and $\overline{Conf}_{<ConceptName>}$ is a confidence score that will be explained later.

The Neural Complex is designed to produce this Aligned Cosine Similarity (CS'). Before providing an overview about the NC and explaining how it works, it is essential to start with an individual concept and see how to calculate its CS' value by using the Coordinate Transformation (CT) process introduced above.

Using the scenario presented in Figure 4-3 (p. 65) as an example: Towards the end of the word-replacement process, six individual Word2Vec models will be produced: `W2V_Universal_Source` and `W2V_Universal_Target`, which are generated by using the original text in the Source and Target Corpus without replacing any keywords; `W2V_Organisation_Source` and `W2V_Organisation_Target` are generated by replacing the `Organisation` related keywords with an invented unique string "xoxovvOrganisationvvoxox" from the associated corpus; subsequently, produce `W2V_People_Source` and `W2V_People_Target` by replacing the `People` related keywords.

Assuming that we are trying to calculate the Aligned Cosine Similarity for the `People` concept (CS'_{People}), then the system will use the related Word2Vec

models (`W2V_Universal_Source/Target` and `W2V_People_Source/Target`) and follow the below steps to do so:

Step 2.1. Use the Coordinate Transformation (CT) process, as discussed in the previous section, to align the `W2V_Universal_Target` model with the `W2V_Universal_Source` model to produce a trained neural network.

More specifically, select all the shared/overlapped words (vocabularies) between the `W2V_Universal_Source` model and the `W2V_Universal_Target` model, in this case {*is, a, school, and, was, founded, by*}.

Let

W_{label} ={ $\vec{V}_{is\_Universal\_Source}$, $\vec{V}_{a\_Universal\_Source}$, $\vec{V}_{school\_Universal\_Source}$, $\vec{V}_{and\_Universal\_Source}$, $\vec{V}_{was\_Universal\_Source}$, $\vec{V}_{founded\_Universal\_Source}$, $\vec{V}_{by\_Universal\_Source}$ } be the associated word vectors in the `W2V_Universal_Source` model.

Correspondingly,

W_{input} ={ $\vec{V}_{is\_Universal\_Target}$, $\vec{V}_{a\_Universal\_Target}$, $\vec{V}_{school\_Universal\_Target}$, $\vec{V}_{and\_Universal\_Target}$, $\vec{V}_{was\_Universal\_Target}$, $\vec{V}_{founded\_Universal\_Target}$, $\vec{V}_{by\_Universal\_Target}$ } be the associated word vectors in the `W2V_Universal_Target` model. Then use W_{input} as the input and W_{label} as the label to train a neural network, which is denoted as NN_{ST} as Figure 4-6 shown below.

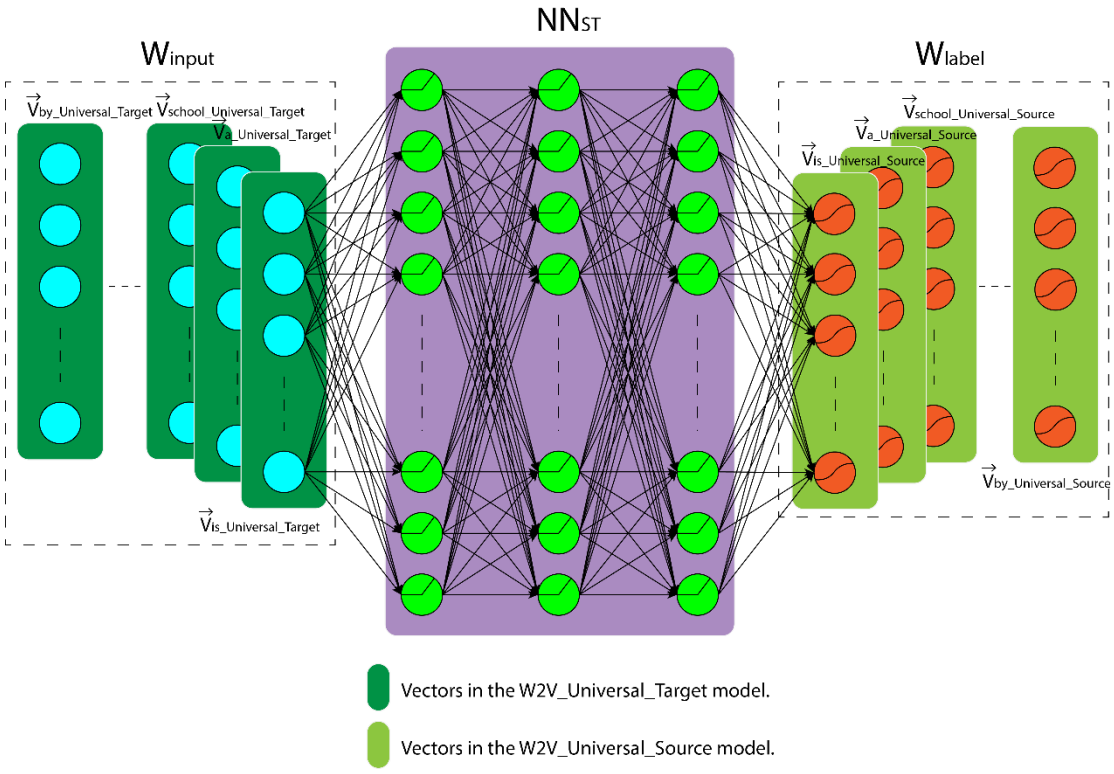


Figure 4-6 Step 2.1. Use the shared/overlapped words to train a neural network to align the W2V_Universal_Target with W2V_Universal_Source.

Step 2.2. Following the same process to align the `W2V_People_Source` model with the `W2V_Universal_Source` model to produce another trained neural network denoted as NN_{People_Source}. In this case, W_{label} ={$\vec{V}_{durmstrang\_Universal\_Source}$, $\vec{V}_{is\ldots}$, $\vec{V}_{a\ldots}$, $\vec{V}_{school\ldots}$, $\vec{V}_{for\ldots}$, $\vec{V}_{young\ldots}$, $\vec{V}_{witches\ldots}$, $\vec{V}_{and\ldots}$,$\vec{V}_{wizards\ldots}$,$\vec{V}_{was\ldots}$,$\vec{V}_{founded\ldots}$,$\vec{V}_{by\ldots}$} are the associated word vectors in the `W2V_Universal_Source` model, and "…" is used to replace "*_Universal_Source*".

The W_{input} ={ $\vec{V}_{durmstrang\_People\_Source}$, $\vec{V}_{is\ldots}$, $\vec{V}_{a\ldots}$, $\vec{V}_{school\ldots}$, $\vec{V}_{for\ldots}$,$\vec{V}_{young\ldots}$,$\vec{V}_{witches\ldots}$,$\vec{V}_{and\ldots}$,$\vec{V}_{wizards\ldots}$,$\vec{V}_{was\ldots}$,$\vec{V}_{funded\ldots}$,$\vec{V}_{by\ldots}$} are the associated word vectors in the `W2V_People_Source` model, and "…" is used to replace "*_People_Source*".

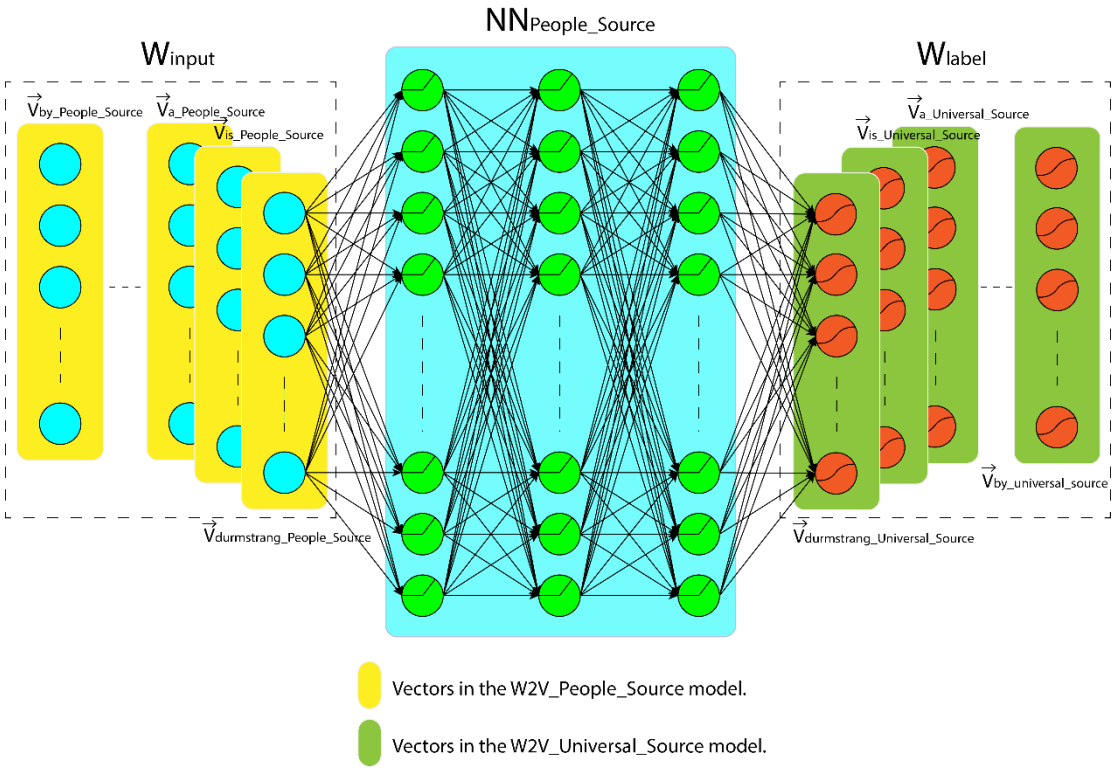


Figure 4-7 Step 2.2. Use the shared/overlapped words to train a neural network to align the W2V_People_Source with W2V_Universal_Source.

Step 2.3. Similar to the last step, we now need to train another neural network (NN_{People_Target}) to align the `W2V_People_Target` model with the `W2V_Universal_Target` model. In this case, W_{label} ={ $\vec{V}_{hogwarts\_Universal\_Target}$, $\vec{V}_{is\ldots}$, $\vec{V}_{a\ldots}$, $\vec{V}_{school\ldots}$, $\vec{V}_{of\ldots}$, $\vec{V}_{witchcraft\ldots}$, $\vec{V}_{and\ldots}$, $\vec{V}_{wizardry\ldots}$,$\vec{V}_{at\ldots}$,$\vec{V}_{scotland\ldots}$,$\vec{V}_{was\ldots}$,$\vec{V}_{founded\ldots}$,$\vec{V}_{by\ldots}$} are the associated word vectors in the `W2V_Universal_Target` model and "…" is used to replace "*_Universal_Target*".

The W_{input} ={ $\vec{V}_{hogwarts\_People\_Target}$, $\vec{V}_{is\ldots}$, $\vec{V}_{a\ldots}$, $\vec{V}_{school\ldots}$, $\vec{V}_{of\ldots}$, $\vec{V}_{witchcraft\ldots}$, $\vec{V}_{and\ldots}$, $\vec{V}_{wizardry\ldots}$,$\vec{V}_{at\ldots}$,$\vec{V}_{scotland\ldots}$,$\vec{V}_{was\ldots}$,$\vec{V}_{founded\ldots}$,$\vec{V}_{by\ldots}$} are the associated word vectors

in the `W2V_People_Target` model, and "…" is used to replace "*_People_Target*".

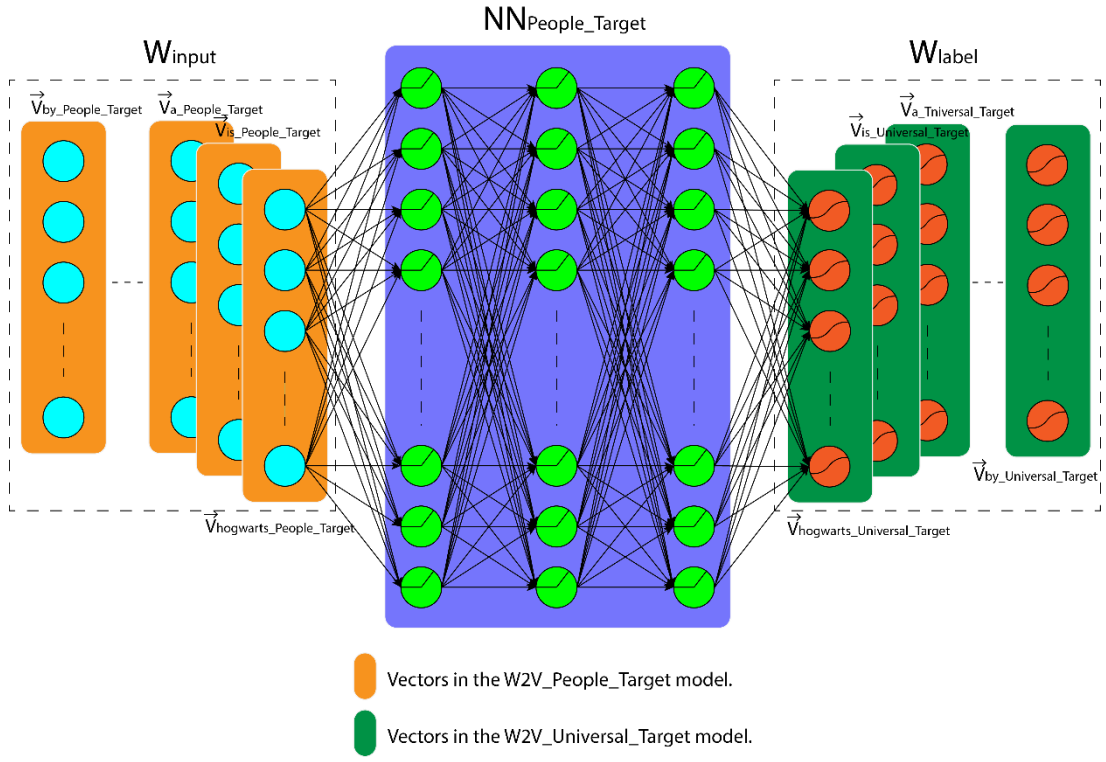


Figure 4-8 Step 2.3. Use the shared/overlapped words to train a neural network to align the W2V_People_Target with W2V_Universal_Target.

Step 2.4. Let $\vec{V}_{People\_Source}$ be the vector of the unique word "xoxovvPeoplevvoxox" in the `W2V_People_Source` model, which is the semantic representation of the `People` concept produced by the word-replacement process discussed in Section 4.2.2 (Figure 4-3, p. 65). Use $\vec{V}_{People\_Source}$ as the input of the NN_{People_Source}, which was produced in Step 2.2, to predict its value in the `W2V_Universal_Source` model (as this unique word does not exist in the `W2V_Universal_Source` model). Denote this predicted vector as $\vec{V}'_{People\_Source}$.

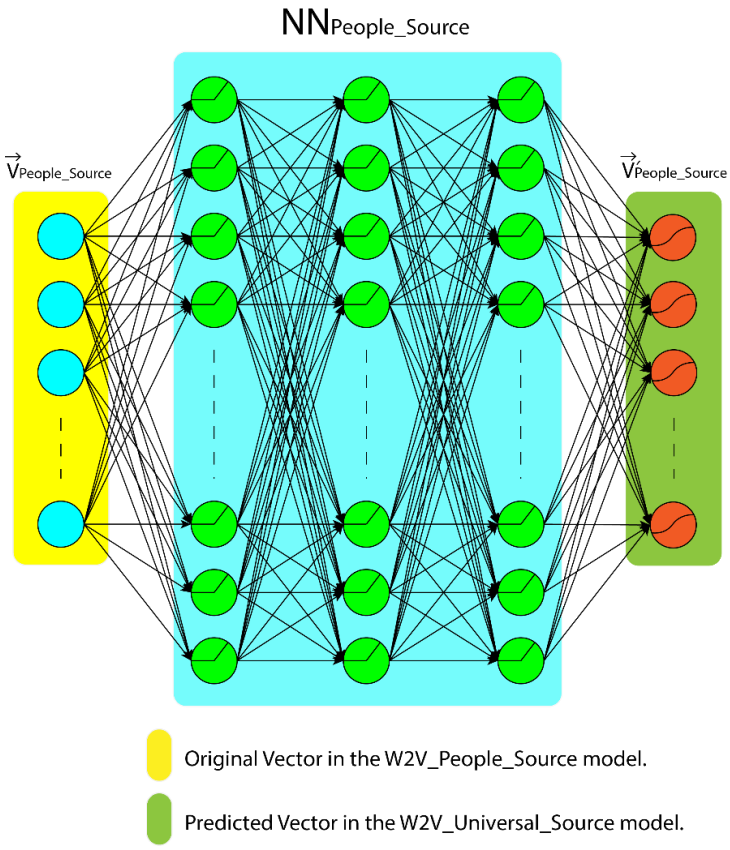


Figure 4-9 Step 2.4. Use the NN_{People_Source} to predict the semantic representation of the People concept (in the source corpus) in the W2V_Universal_Source model.

Step 2.5. Then use the NN_{People_Target} (generated from Step 2.3) to predict the value of $\vec{V}_{People\_Target}$ in the `W2V_Universal_Target` model. Denote this predicted value as $\vec{V}'_{People\_Target}$. Then use $\vec{V}'_{People\_Target}$ as the input of the NN_{ST}, which was produced in Step 2.1, to predict its value in the `W2V_Universal_Source` model and denote it as $\vec{V}''_{People\_Source}$.

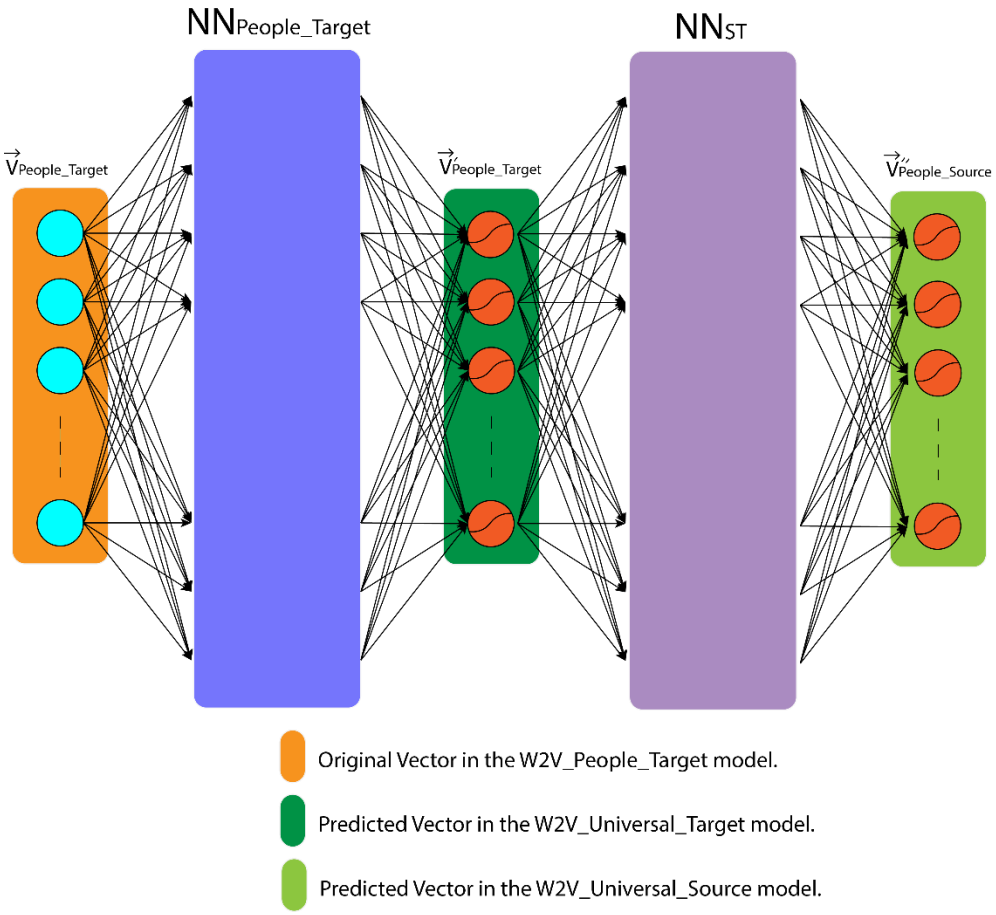


Figure 4-10 Step 2.5. Use NN_People_Target and NN_ST to predict the semantic representation of the People concept (in the target corpus) in the W2V_Universal_Source model.

Step 2.6. Finally, the Cosine Similarity CS_{People} and the Aligned Cosine Similarity CS'_{People} between $\vec{V}_{People\_Source}$ and $\vec{V}_{People\_Target}$ is calculated by the following formulas, and the overall process chart shown in Figure 4-11:

$$CS_{<ConceptName>} = \frac{\vec{V}_{<ConceptName>\_Source} \cdot \vec{V}_{<ConceptName>\_Target}}{\left\|\vec{V}_{<ConceptName>\_Source}\right\| \left\|\vec{V}_{<ConceptName>\_Target}\right\|} \quad 4\text{-}4$$

$$CS'_{<ConceptName>} = \frac{\vec{V}'_{<ConceptName>\_Source} \cdot \vec{V}''_{<ConceptName>\_Source}}{\left\|\vec{V}'_{<ConceptName>\_Source}\right\| \left\|\vec{V}''_{<ConceptName>\_Source}\right\|} \quad 4\text{-}5$$

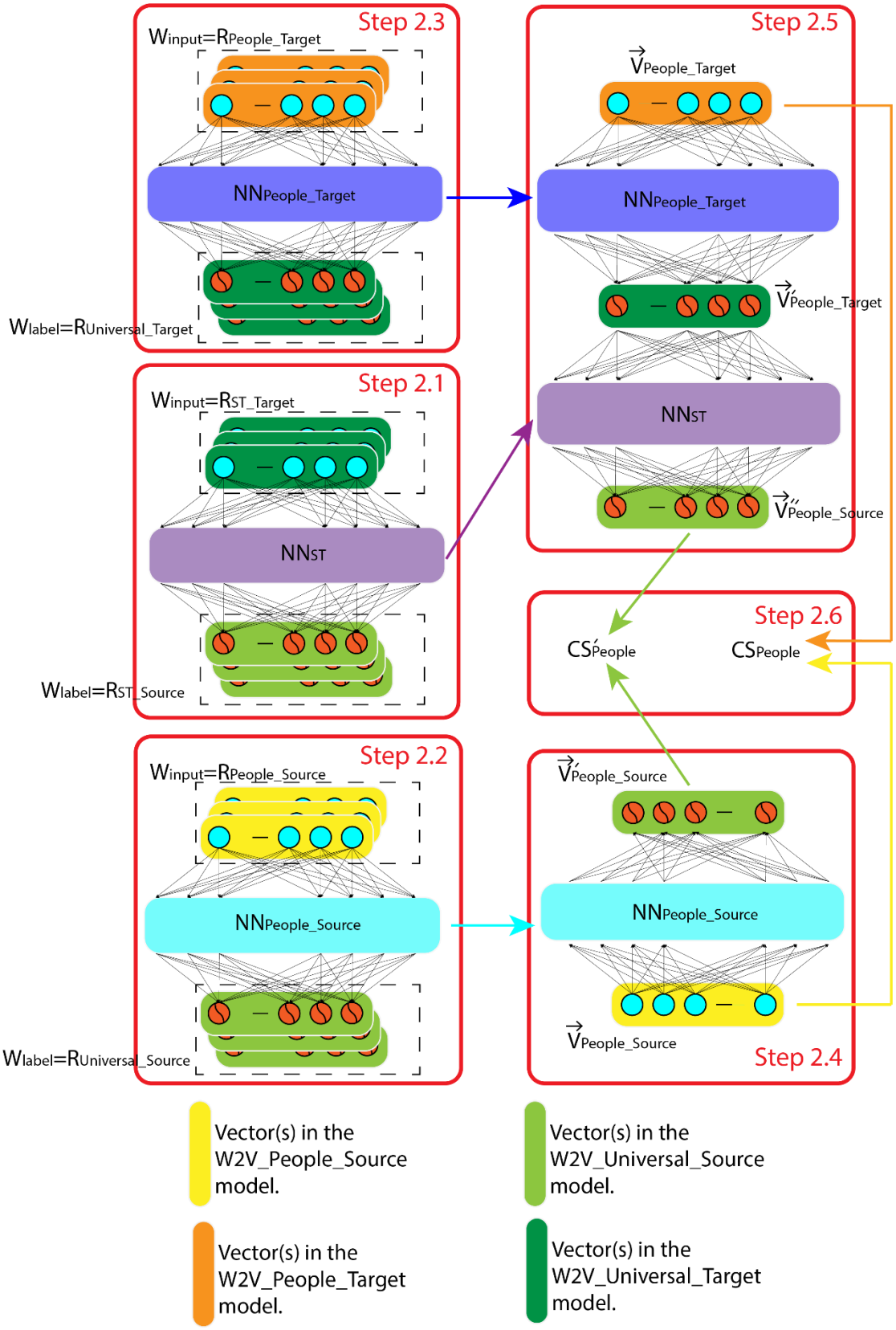


Figure 4-11 Overall Process for an Individual Concept. Please refer to the previous discussion for more details.

In the above figure, R_{ST_Target} is the associated word vectors (words that are shared between `W2V_Universal_Source` and `W2V_Universal_Target`) in the `W2V_Universal_Target` model, and R_{ST_Source} is its equivalent in the `W2V_Universal_Source` model. R_{People_Source} is the associated word vectors (shared between `W2V_People_Source` and `W2V_Universal_Source`) in the `W2V_People_Source` model, and its equivalent in the `W2V_Universal_Source` model is denoted as $R_{Universal\_Source}$. Correspondingly, R_{People_Target} and $R_{Universal\_Target}$ are the related vectors in the `W2V_People_Target` and `W2V_Universal_Target` model.

Step 2.1 only needs to be done once; hence, by repeating Step 2.2 to Step 2.6 (and replacing the `W2V_People_Source/Target` with the related `W2V_<ConceptName>_Source/Target` models), the system will be able to generate a CS' value for each individual concept. Each enumeration will train two separate neural networks as part of the CT process (Step 2.2 and Step 2.3). So in total, the system needs to train $(1 + n \times 2)$ neural networks, where n is the number of concepts. By completing this process, we will end up with a structure that looks like Figure 4-12 shown below.

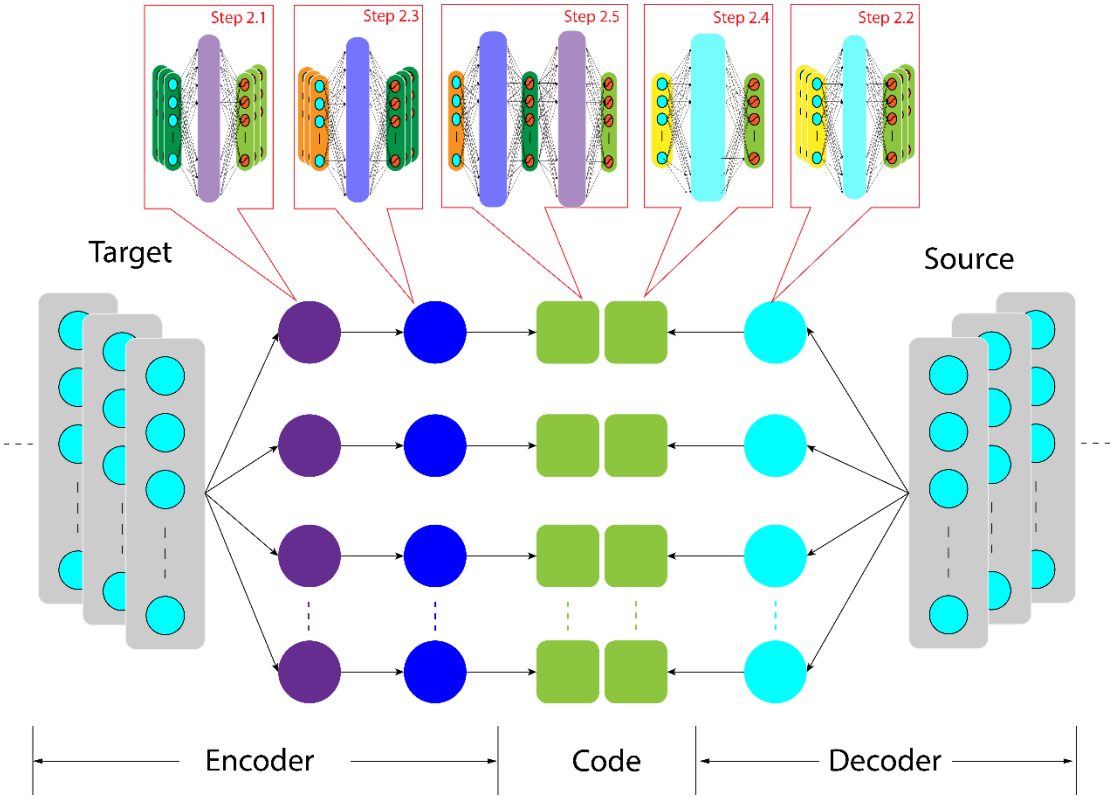


Figure 4-12 Overall Neural Complex Architecture. Similar to an AutoEncoder, NC will try to reconstruct ($CS' = 1$) the semantic representation in the target (input) corpus at the source (output) corpus.

Each horizontal layer in the above figure represents the process of calculating the aligned cosine similarity for an individual concept, and it is essentially what has been presented in Figure 4-11. It looks very similar to an AutoEncoder, a type of artificial neural network for unsupervised learning like Figure 4-13 shown below.

Essentially, it is trying to reconstruct the semantic representations in the target (input) corpus and the source (output) corpus and make them as close to each other as possible by maximising the CS' value (in fact, the Alignment Coefficient, which will be introduced in Section 6.2.3).

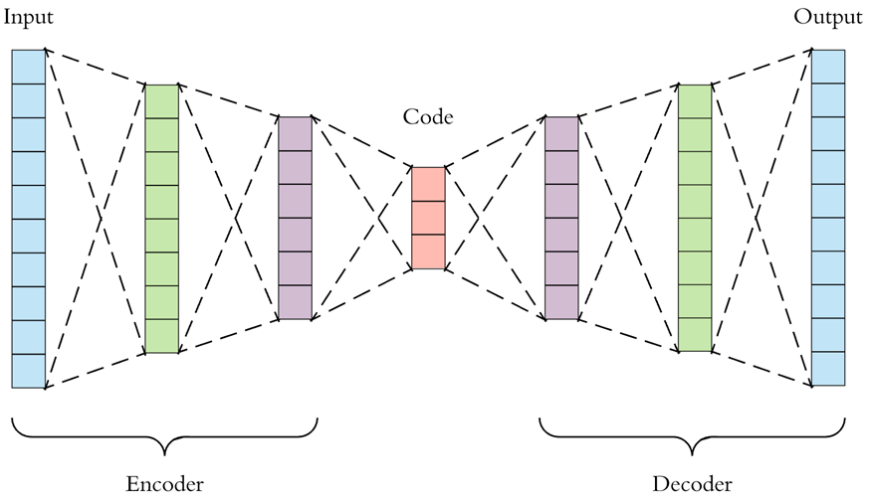


Figure 4-13 A standard autoencoder structure. The left side of the NN is considered as an "Encoder", and the right side is an "Decoder".

However, it is not an autoencoder, and it is not even an ordinary neural network. This is because each individual node in Figure 4-12 is another independent neural network instead of an artificial neuron. So basically, this is a combination of hundreds of neural networks and is denoted as a "Neural Complex" in this thesis. This thesis has not yet described how exactly it works and how to train this complex, but the rest of the thesis will gradually explain these matters.

In summary, by using the Neural Complex, we will eventually align all different Word2Vec models with a single model -- the `W2V_Universal_Source` model. This means the vector representation for various concepts ($\vec{V}$) can be compared directly. The CS' value for a specific concept is, in fact, the Cosine Similarity between its representation in the source corpus and its equivalent representation in the target corpus. Based on the second assumption discussed in Section 4.1, all the CS' values should be close to 1 in theory. However, in practice, the CS' value for a specific concept is determined by the accuracy of

the related neural networks[17]. For example, if all the related neural networks have a 100% accuracy (and with sufficient documents in the corpora), then, for a specific concept, its vector representation in the Source and Target Corpus should completely overlap with each other ($CS' = 1$) after the Neural Complex process. However, if one of the neural networks has low accuracy, then it is likely that the projection of $\vec{V}$ is no longer accurate and the value will be shifted randomly, which will make the final result (CS') much smaller than 1.

Hence, the rationale is to make the related neural networks only work on those informative concepts and reduce the accuracy if they are less informative, so that the CS' value for informative concepts will have a value closer to 1. This is achieved by leveraging the overfitting mechanism within the neural network.

Overfitting is a phenomenon in the Machine Learning and Deep Learning study, where the trained neural network works extremely well on the training dataset, but performs poorly on the real/testing dataset. Many factors can cause overfitting, but in general, when overfitting happens, it means the neural network model is too complex for the problem it is trying to resolve. Therefore, a common approach to overcome overfitting is to reduce the complexity of the model.

In a typical neural network-related application, overfitting is something that needs to be avoided. To identify the overfitting, it is used to split the known data

[17] Strictly speaking, the associated/related W2V models and their randomization also affect the final CS' value. However, as those W2V models (specifically, the vectors included in the W2V models) either used as the input of the related neural networks or used as the label of the training dataset. Hence, impacts from W2V models have not been mentioned here.

into a training dataset and testing dataset, then use only the training dataset to train the neural network and test the result on the testing dataset. If the testing dataset's performance/outcome is similar to the training dataset, then it is an appropriate model; otherwise, it will be considered overfitting.

The Neural Complex, however, uses all the shared words between two Word2Vec models as the input to train the related neural networks without splitting them into two sets. It is because this thesis introduces a new approach to handle overfitting.

4.3.3.2 Implementation Plan/NC Training Approach

This new approach is achieved by implementing/using the Neural Complex in a unique way, where overfitting is no longer something that needs to be actively avoided. In fact, this new approach initially uses a complex neural network structure to deliberately make the Neural Complex (the related neural networks within the NC) overfit on a set of selected concepts. Then slowly reduce the complexity based on the evaluation result to identify the best neural network structure before applying this structure to the rest of the concepts. Essentially, it is how we are going to train the Neural Complex. The detailed steps are shown below:

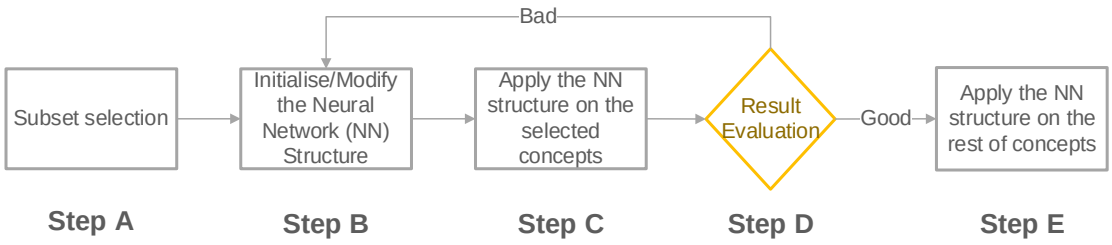


Figure 4-14 Neural Complex Implementation Plan/Training Process. Details are explained below.

Step A. Create a subset from the shared corpus concepts, which only contains those concepts that have a valid mapping in the guiding ontology, as discussed in Section 4.2.1. For convenience, it is denoted as “Mapped Subset” in the rest of the thesis. This mapped subset is, in fact, used as the training dataset (in Step C discussed below) to identify the best neural network structure that will apply to all the individual neural networks in the Neural Complex.

Step B. Initialise (or modify) the neural network (NN) structure. At the initialisation stage, the structure should include at least three hidden layers, and on each layer the number of nodes should be at least 20 times larger than the feature size of the input Word2Vec model. When modifying the NN structure, slowly reduce or increase its complexity based on the evaluation result. There are many ways to modify the complexity, for example, reduce the number of hidden layers or reduce the number of nodes on each hidden layer. By default, the number of nodes on each hidden layer is the same, but it is possible to only reduce or increase the number on certain layers. Various tests have been conducted within the two experiments, and more details will be provided in the related chapters.

Step C. Use the initialised or modified NN structure (from Step B) and the selected Mapped Subset (from Step A) to perform the Neural Complex process discussed above (Figure 4-11 and Figure 4-12) and calculate the related CS' values.

Step D. The two experiments conducted in this thesis have different evaluation processes. Within the first experiment discussed in Chapter

5, a good result means most of the CS' values are close to 1. There is also an intuitive evaluation in the first experiment, which will be discussed later. The second experiment discussed in Chapter 6 is on a much larger scale. Hence, a new parameter called "Alignment Coefficient" has been designed to assess how good the result is, and it can be considered as the loss function adopted in the Neural Complex. More detail will be given later. In general, if it is a good result, then move to the next step. Otherwise, return to Step B to modify the NN structure and try again.

Step E. At this stage, the NN structure used will be considered as the best structure for the domain knowledge. Hence, as with Step C, the system will apply this NN structure to the rest of the concepts (considering it as the testing/real dataset) to generate related CS' values.

Towards the end of Step E, the system will generate an individual CS' value for all the shared corpora concepts, and the more it is close to 1, the more informative the concept is. The reason is as follows.

Intuitively, an informative concept should have a more complex and enriched semantic representation than a non-informative concept. Moreover, an extension of the first assumption discussed in Section 4.1 is that the complexity of the semantic representation should also be embedded in the vector. Hence, for an informative concept, there is more semantic information embedded in the related representation vectors ($\vec{V}_{<ConceptName>\_Source}$ and $\vec{V}_{<ConceptName>\_Target}$) than for the non-informative concepts. In other words, $\vec{V}_{<ConceptName>\_Source}$ and $\vec{V}_{<ConceptName>\_Target}$ of an informative concept are more "complex" than

the related vectors of a non-informative concept, even they have the same feature/dimension size.

As already discussed above, overfitting normally means the model is too complex for the problem it is trying to resolve. Hence, a key fact behind this Neural Complex implementation plan (or training process) is that during the CT process the neural network trained for an informative concept is less likely to be overfitted than the neural network trained for a non-informative concept, because the semantic complexity of the former will overcome, to a certain extent, the potential overfitting. In other words, the neural network is trying to resolve a more complex problem.

Therefore, the overall idea here is to build a subset of corpus concepts (Mapped Subset) that only contains informative concepts (Step A, Figure 4-14, p. 86). A further discussion below (Section 4.3.3.3) explains why concepts within the Mapped Subset can be considered informative. Then use this Mapped Subset as the training data to identify a tailored neural network structure that works on informative concepts without overfitting (Step B to Step D). After that, apply that structure to the rest of the concepts (Step E) to identify if they are informative or not. If the neural network is overfitted, then the concept is a non-informative concept since it does not have a complex enough representation to overcome the overfitting; otherwise, it is an informative concept (since the NN structure is designed/trained to work on informative concepts). In other words, use overfitting to identify the complexity of the problem, which reflects the informativeness of a concept.

4.3.3.3 A Further Discussion of the Mapped Subset and NN Structure

To better understand the reason behind the Neural Complex and its training process, there are a few additional things that need to be further discussed here. The first one is about Step A (Figure 4-14, p. 86) and the guiding ontology.

4.3.3.3.1 Guiding Ontology and the Mapped Subset

In Step A, the corpus concepts with a valid mapping in the guiding ontology have been selected to build this subset (Mapped Subset). As mentioned earlier, the guiding ontology is a well-constructed ontology and closely related to the domain (or describes the domain from a different but related perspective). Then, to a certain extent, concepts (or ontology classes) within this ontology must be informative enough to be able to represent the domain knowledge. Hence, it is reasonable to claim that those corpus concepts with a valid mapping in the guiding ontology can, most likely, be considered informative within the domain knowledge.

For example, in the first experiment, the domain is about News, and the guiding ontology is the BBC Core Concepts Ontology [61]. The goal is to analyse the semantic importance from the “Donald Trump” perspective. In the second experiment, the domain is about “Candida”, and the goal is to analyse the semantic importance from the “Disease” perspective by using a “Candida Gene” ontology as the guiding ontology.

4.3.3.3.2 Consistent NN Structure

The second thing that needs further discussion is why the same NN structure is used to re-train a new neural network for each related CT process in the Neural Complex, instead of allowing them to have a different NN structure.

Without the constraint of using the same neural network structure for all concepts, it is reasonable to believe that we could find different neural network structures for different concepts, and eventually make all the CS' values very close to 1. However, the main purpose of the Neural Complex and related CT processes is to assess the level (or degree) of the overfitting to distinguish the informative concept from the non-informative concept instead of maximising the CS' value for all concepts (except the stage where using the informative Mapped Subset to identify the most suitable NN structure). Hence, having a different NN structure for different concepts is against its original purpose, and this is the first reason.

As already pointed out above, there is a stage (Step B to Step D, Figure 4-14, p. 86) where we do need to maximise the CS' value, or an equivalent parameter (Alignment Coefficient, which will be given in the second experiment), to identify the best or the most suitable NN structure. So why is the use of different NN structures still forbidden at this stage? This is because of the second reason: to ensure the process is objective and consistent.

One of the challenges discussed previously is that it is difficult for the concept selection process to be objective and consistent, partly because of the diversity of the ontology itself, but it is more because the ontology itself is a subjective and abstract idea. Even when describing the ontology from the same

perspective, people may still end up with a different selection of concepts. It is due to the human factor – people always have their own preferences. Using the same NN structure is a mechanism implemented in the SI algorithm to address this challenge.

4.3.3.4 Summary and Discussion of the Neural Complex

In summary, Neural Complex (NC) is a novel method to “polymerise” multiple neural networks and combine them into a higher-order neural network structure, where each node on the hidden layer is another independent[18] neural network.

Within a traditional NN application, the goal is to design a NN structure that can produce an accurate prediction based on the input data. To achieve this, we need to avoid the overfitting problem to ensure the trained NN performs well on both training data and on real data.

The NC, however, takes the opposite approach. Instead of avoiding overfitting to produce an accurate prediction, NC uses overfitting as a measurement to assess the complexity of the problem itself.

The overall NC architecture is shown in Figure 4-12 (p. 83). Each horizontal layer represents the process of calculating the aligned cosine similarity for an individual concept, and Figure 4-11 (p. 81) shows how exactly this process works.

[18] Means it has been trained separately using different training data.

Section 4.3.3.2 discussed the NC training process. The Mapped Subset discussed in Step A, Figure 4-14 (p. 86) is the equivalent of the training dataset in the traditional machine learning context. Step D can be considered the loss function, and more detailed discussions will be given in Session 5.3.1 and Section 6.2.3. Based on the result (indicated by the loss function), the neural complex will adjust its parameter accordingly (Steps B and C). The parameter of the neural complex is the NN structure (such as the number of hidden layers and nodes) that can maximise the output. Once the training process is finished, the system will then apply the result (trained/best NN structure) to the testing/real dataset, which is essentially what Step E does. This mechanism will guarantee all concepts can be processed equally under the same condition without being affected by the human factor to ensure the overall process is objective and consistent.

4.3.4 Final IC Calculation

The IC can be interpreted as a weighted CS' value, as suggested by Equation 4-3. The weight ($\overline{Conf}_{<ConceptName>}$) is the confidence score for that specific concept. It has been implemented differently in the two experiments and will be discussed later in the related chapters.

Hence, the final IC values can be calculated based on the associated CS' produced via the Neural Complex.

4.4 Step 3 (Figure 4-1) - Connectivity Coefficient Calculation

Unlike those count-based measures discussed in Chapter 2, which mainly focus on a single area (e.g. co-occurrence), the idea of “semantic importance”

within the SI algorithm has a two-fold interpretation. Firstly, the concept needs to be informative. Secondly, it should be well connected (strong correlation) with other concepts in the same domain. The informativeness is measured by the IC value discussed above, and this section will focus on how to measure the Connectivity Coefficient (CC) value.

In short, the CC value is calculated based on the correlation strength between the relevant class pairs. The correlation mentioned here refers to the semantic correlation instead of the statistical correlation.

There is actually a difference between correlation at the statistical level and at a deeper semantic level. At the statistical level, correlation refers to a numerical association between a pair of variables. For example, there is a linear correlation between travel speed and estimated arrival time. However, the correlation has a much broader meaning at a deeper semantic level and is not restricted to the numerical relationship between two variables. For example, the concept of “Event” should have a strong semantic correlation with the concept of “Place” since all events must happen at a location. Moreover, there should be a strong semantic correlation between “Money” and “Price”, not only because they are, quite often, represented by numbers, but more importantly because they are semantically close to each other.

There are some research publications within this area. For example, authors of [62] introduced a method to calculate the semantic correlation of the words by using a latent topic model combined with a bootstrapping procedure. They have successfully identified some of the interesting correlations, e.g. between Yankee & Catcher, and between Toyota & Mileage. Instead of building an

additional latent topic model and measuring semantic correlation at the word level, within this thesis, we will adopt an existing statistical method and use the DSMs generated in the previous steps to calculate the semantic correlation at the concept level.

Unlike the IC calculation, which uses an entirely new approach (Neural Complex) developed as part of this thesis, the contribution of this thesis as regards the CC calculation is that it proposes a novel and unobvious application of an existing statistical method.

4.4.1 What is the Maximal Information Coefficient (MIC)?

The adopted statistical method is called the Maximal Information Coefficient (MIC). It was introduced by David Reshef in 2011 [10] to identify all the possible associations between two variables in large datasets.

The rationale of MIC is that if there is an association between two variables, then it is possible to draw a grid on the scatterplot of the two variables that partitions the data to encapsulate that association/relationship. Hence, the MIC is calculated by exploring all grids up to a maximal grid resolution. We compute for every pair of integers (g,h) up to that resolution the largest possible mutual information achievable by any g-by-h grid applied to the data. After that, normalise the value of the identified mutual information to ensure a fair comparison between grids of different dimensions and to obtain modified values between 0 and 1. Let $M = (m_{g,h})$ be the characteristic matrix, where $m_{g,h}$ is the highest normalised mutual information achieved by any g-by-h grid. Hence the maximum value in M is the statistic MIC that we are looking for. An illustrated example is shown in Figure 4-15 [10] and Equation 4-6 is used to calculate the

MIC value between variable x and y. In our application, x and y corresponded to concepts, e.g. People and Organisation. Each x value is a cosine between a vector of the concept and the vector of a specific word, as Table 4-3 shows below; similarly the y values. The below equation calculates the actual MIC value for each individual concept pair (x,y):

$$MIC(x;y) = \max(m_{g,h}) = \max_{g,h<B} \frac{I(g;h)}{log_2(\min(g,h))} \quad 4\text{-}6$$

where I is the mutual information measure of the probability distribution induced on the boxes of scatterplot (g-by-h grid). B is a function of sample size n (the number of points in the scatterplot) and Reshef et al. [10] suggested usually set $B = n^{0.6}$ based on their experiences.

MIC takes values between 0 and 1, where 0 indicates statistical independence and 1 means a completely noiseless association.

Compared with other statistical methods, there are several advantages of MIC due to the generality and equitability of the method.

Generality means that with sufficient data, the MIC is capable of capturing a wide range of linear and non-linear associations, while the other methods are normally limited to one specific relationship (e.g. Pearson's R can only detect linear association). This means that there is no need to make any assumptions about the distribution of the variables when applying the MIC algorithm. However, as a downside, the final MIC value does not indicate the type of association between two variables.

Equitability indicates that this method gives similar scores to equally noisy relationships of different types. Moreover, for a functional relationship, which means a distribution (x,y) in which y is a function of x, it will assign a score that roughly equals the coefficient of determination (R^2) value (with sufficient data) [10].

A more detailed comparison is also provided by Reshef et al., as Table 4-2 shown below.

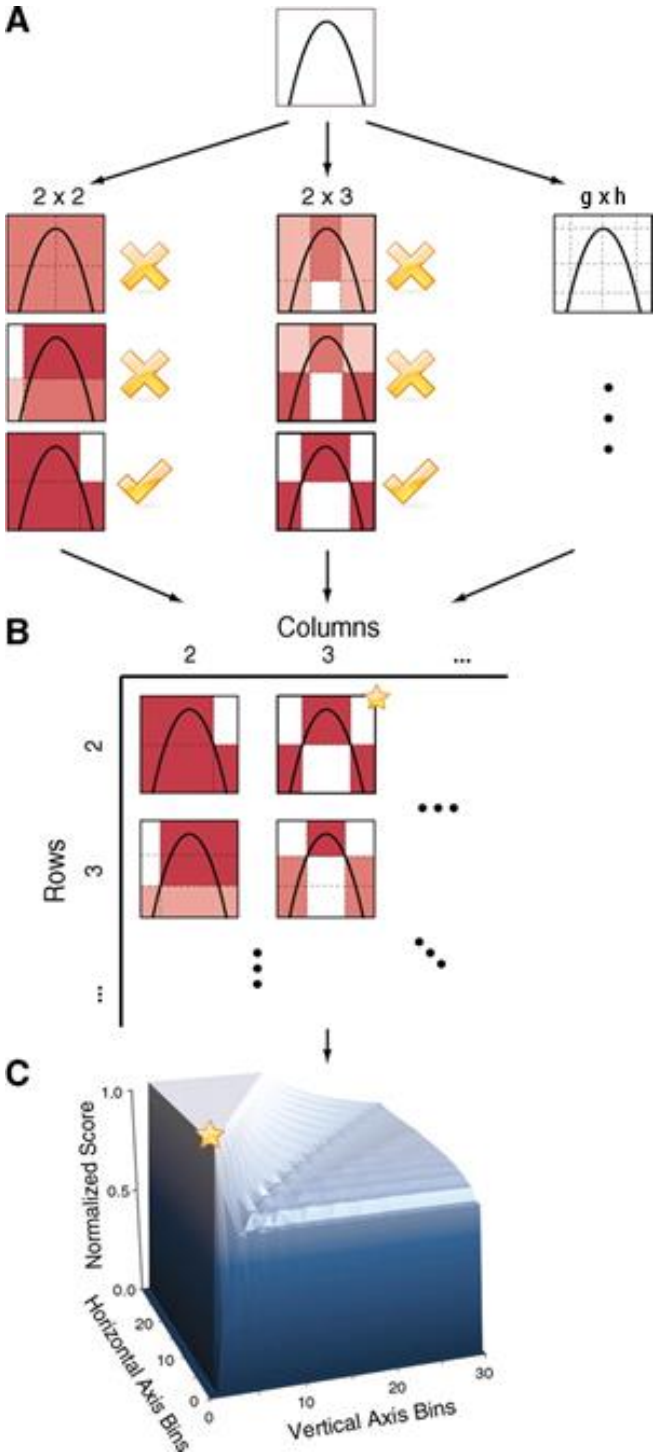


Figure 4-15 MIC Calculation

Computing MIC: "*(A) For each pair*[19] *(x,y), the MIC algorithm finds the g-by-h grid with the highest induced mutual information. (B) The algorithm normalises*

[19] In the original paper published by Reshef et al. grid size was denoted by x and y, which was very confusing because x and y also used to denote variables. In this thesis, we use g and h to denote grid size.

the mutual information scores and compiles a matrix that stores, for each resolution, the best grid at that resolution and its normalised score. (C) The normalised scores form the characteristic matrix, which can be visualised as surface; MIC corresponds the highest point on this surface. In this example, there are many grids that achieve the highest score. The star in (B) marks a sample grid achieving this score, and the star in (C) marks that grid's corresponding location on the surface." [10]

Relationship Type	MIC	Pearson	Spearman	Mutual Information (KDE)	Mutual Information (Kraskov)	CorGC (Principal Curve-Based)	Maximal Correlation
Random	0.18	-0.02	-0.02	0.01	0.03	0.19	0.01
Linear	1.00	1.00	1.00	5.03	3.89	1.00	1.00
Cubic	1.00	0.61	0.69	3.09	3.12	0.98	1.00
Exponential	1.00	0.70	1.00	2.09	3.62	0.94	1.00
Sinusoidal (Fourier frequency)	1.00	-0.09	-0.09	0.01	-0.11	0.36	0.64
Categorical	1.00	0.53	0.49	2.22	1.65	1.00	1.00
Periodic/Linear	1.00	0.33	0.31	0.69	0.45	0.49	0.91
Parabolic	1.00	-0.01	-0.01	3.33	3.15	1.00	1.00
Sinusoidal (non-Fourier frequency)	1.00	0.00	0.00	0.01	0.20	0.40	0.80
Sinusoidal (varying frequency)	1.00	-0.11	-0.11	0.02	0.06	0.38	0.76

Table 4-2 Scores given to various noiseless functional relationships by several different statistics. Maximal scores in each column are accentuated. [10] *page 1519*

4.4.2 The Use of MIC to Calculate Concepts' Correlation Values

Consider a domain ontology as a function that could be used to represent knowledge within a domain. Then the concepts, which will eventually become ontology classes, will be the variables of this function. Moreover, individual words that exist in the corpus are the essential components and "material" that build the domain knowledge. Therefore, each word will have an influence on

the knowledge that the ontology represents. Hence, the individual word will have an indirect impact on the ontology manifested through the concepts that the individuals belong to.

The rationale is that if we could measure the impact a word could exercise on the various concepts, we could then understand the relations between these concepts. In other words, considering each word in the corpus as an independent sample, the concepts are the variables (or properties), and the value of a specific variable/property in a specific sample is the Cosine Similarity between that word and that concept.

In this way, the system can generate a sample table with each row corresponding to a word in the vocabulary list of a universal Word2Vec model, and each column corresponding to a shared concept[20] that has been identified in the Source Corpus[21].

Finally, in using the sample table as the input, the MIC algorithm generates the result that indicates the strength of the correlation between all the class pairs.

In practice, the sample table is based on the source corpus. Hence the universal Word2Vec model is `W2V_Universal_Source`. Two different approaches were considered to calculate the `CS` value for each individual cell in the sample table.

[20] A shared concept means it exists in both the Source and Target Corpus.
[21] Please notice that we only use the Source Corpus to calculate the MIC value. Since the Source and Target Corpus are about the same domain, so the correlation between concept plairs should be consistent. In other words, there is no need to duplicate the calculation.

The first approach is to use $\vec{V}'_{<ConceptName>\_Source}$ which has been generated in Step 2.4 (Figure 4-9, p. 79) in the Neural Complex shown in Section 4.3.3.1. Since it has been aligned with the `W2V_Universal_Source` model already, then the CS between a specific word and a specific concept can be calculated directly by using the associated word vector in the `W2V_Universal_Source` and the related $\vec{V}'_{<ConceptName>\_Source}$.

The second approach is to use the $\vec{V}_{<ConceptName>\_Source}$ within the original `W2V_<ConceptName>_Source` models generated by the DSM construction process without going through the Neural Complex at all. In this case, the rows of the sample table are still the individual words within `W2V_Universal_Source` model's vocabulary list. However, instead of using the associated vectors within the `W2V_Universal_Source` model, the actual vector for an individual word is from the original `W2V_<ConceptName>_Source` models. Since those models have a different vocabulary list compared to the `W2V_Universal_Source` model (because of the word-replacement process), the following rules have been applied to handle the difference:

1. For a specific concept, if the missing word is one of the words that has been replaced during the word-replacement process, then the value of the cosine similarity would be 1.
2. If not, then assign 0 to the cosine similarity.

Then follow the same process to generate the related CS and the final sample table.

This thesis uses the second approach because there is a potential issue within the first one, which could cause a random CS value. It is because $\vec{V}'_{<ConceptName>\_Source}$ is only accurate for those informative concepts since the Neural Complex is designed to overfit those non-informative concepts.

Using Figure 4-3 as an example, the sample table is shown below:

Concept / Word	Organisation (Org)	People
durmstrang	1	$CS(\vec{V}_{durmstrang\_People}, \vec{V}_{People\_Source})$
is	$CS(\vec{V}_{is\_Org}, \vec{V}_{Org\_Source})$	$CS(\vec{V}_{is\_People}, \vec{V}_{People\_Source})$
a	$CS(\vec{V}_{a\_Org}, \vec{V}_{Org\_Source})$	$CS(\vec{V}_{a\_People}, \vec{V}_{People\_Source})$
school	$CS(\vec{V}_{school\_Org}, \vec{V}_{Org\_Source})$	$CS(\vec{V}_{school\_People}, \vec{V}_{People\_Source})$
for	$CS(\vec{V}_{for\_Org}, \vec{V}_{Org\_Source})$	$CS(\vec{V}_{for\_People}, \vec{V}_{People\_Source})$
young	$CS(\vec{V}_{young\_Org}, \vec{V}_{Org\_Source})$	$CS(\vec{V}_{young\_People}, \vec{V}_{People\_Source})$
witches	$CS(\vec{V}_{witches\_Org}, \vec{V}_{Org\_Source})$	$CS(\vec{V}_{witches\_People}, \vec{V}_{People\_Source})$
and	$CS(\vec{V}_{and\_Org}, \vec{V}_{Org\_Source})$	$CS(\vec{V}_{and\_People}, \vec{V}_{People\_Source})$
wizards	$CS(\vec{V}_{wizards\_Org}, \vec{V}_{Org\_Source})$	$CS(\vec{V}_{wizards\_People}, \vec{V}_{People\_Source})$
was	$CS(\vec{V}_{was\_Org}, \vec{V}_{Org\_Source})$	$CS(\vec{V}_{was\_People}, \vec{V}_{People\_Source})$
funded	$CS(\vec{V}_{funded\_Org}, \vec{V}_{Org\_Source})$	$CS(\vec{V}_{funded\_People}, \vec{V}_{People\_Source})$
by	$CS(\vec{V}_{by\_Org}, \vec{V}_{Org\_Source})$	$CS(\vec{V}_{by\_People}, \vec{V}_{People\_Source})$
nerida_vulchanova	$CS(\vec{V}_{nerida\_vulchanova}, \vec{V}_{Org\_Source})$	1

Table 4-3 Sample Table based on Figure 3

The Organisation and People will be the equivalent x and y (or y and x) explained in the last section (Section 4.4.1). Within $CS(\vec{V}_{is\_Org}, \vec{V}_{Org\_Source})$, $\vec{V}_{is\_Org}$ represents the vector for the word "is" within the `W2V_Organisation_Source` model, $\vec{V}_{Org\_Source}$ represents the vector for the "Organisation" concept (the invented unique string replacing all the words covered by the concept) within the `W2V_Organisation_Source` model. The rest of them follows the same rule.

In Table 4-3, there are only two variables, Organisation and People. Hence only one MIC value will generate from it. Moreover, the x-axis of the scatterplot is the CS values the words might have with respect to Org, and similarly for the y-axis and People.

4.4.3 Final CC Calculation

Using the sample table as the input, the MIC algorithm generates the result that indicates the strength of the correlation between all the class pairs. The connectivity coefficient for concept a can then be calculated as:

$$CC_a = \sum_{i=1}^{|R_a|} MIC(a; b_i) \times \overline{Conf}(b_i) \times \overline{Conf}(a) \qquad 4\text{-}7$$

where $CC_a \in [0,1]$, $R_a = \{\langle a, b\rangle | \exists b, \langle a, b\rangle \in R\}$. R is the set of associated concept pairs (associated with a given concept a). $\overline{Conf}(c)$ is the confidence score for concept c. The way to calculate these confidence scores are different in the two experiments and have different meanings. In the first experiment, the confidence score represents the accuracy of a concept that has been identified by a NER tool. In the second experiment, it means how stable an associated NN is. A more detailed discussion will be given in the related chapters.

4.5 Step 4 (Figure 4-1) - The Final SI Calculation

As explained before, IC is weighted (confidence score ranged from [0,1]) CS' value (ranged from [-1,1]). So, IC is in the range of [-1,1], and CC is in the range of [0, n], where n is the number of the class pairs. Hence, in order to make the final SI result within the range of [-1,1] (so that SI values for different concepts, within the same ontology or in different ontologies, can be meaningfully

compared with each other), a normalisation method has been added to spread the range of obtained IC and CC values to fully cover the interval from -1 to 1 with the following equation(s):

$$Normalised\left(IC_{<ConceptName>}\right) = \frac{IC_{<ConceptName>} - Min(IC)}{Max(IC) - Min(IC)} \times 2 - 1 \quad 4\text{-}8$$

$$Normalised\left(CC_{<ConceptName>}\right) = \frac{CC_{<ConceptName>} - Min(CC)}{Max(CC) - Min(CC)} \times 2 - 1 \quad 4\text{-}9$$

where $IC_{<ConceptName>}/CC_{<ConceptName>}$ is the IC/CC value for a specific concept, $Min(IC/CC)$ is the minimum value of IC/CC, $Max(IC/CC)$ is the maximum value of IC/CC.

So, by combining the related equations together, the final equation to calculate the Semantic Impact value for concept a is:

$$SI_a = \lambda_1\left(\frac{\frac{\vec{V}'_{a\_Source} \cdot \vec{V}''_{a\_Source}}{\left\|\vec{V}'_{a\_Source}\right\|\left\|\vec{V}''_{a\_Source}\right\|} \times \overline{Conf}(a) - Min(IC)}{Max(IC) - Min(IC)} \times 2 - 1\right) + \lambda_2\left(\frac{\sum_{i=0}^{|R_a|} MIC(a, b_i) \times \overline{Conf}(b_i) \times \overline{Conf}(a) - Min(CC)}{Max(CC) - Min(CC)} \times 2 - 1\right) \quad 4\text{-}10$$

where

- $\vec{V}'_{a\_source}$ is the concept distribution vector (within the source corpus) generated for a via the word-replacement process (Section 4.2.2).

- $\vec{V}''_{a\_Source}$ is the aligned concept distribution vector generated from the Neural Complex (Section 4.3.3, Step 2.5 in Figure 4-10, p. 80).
- $MIC(a, b_i)$ is the MIC value between a and another concept b (Section 4.4.2).
- $\overline{Conf}(c)$ is the confidence score for concept c.
- λ_1 and λ_2. $\lambda_1 + \lambda_1 = 1$, $\lambda_1 \in [0,1]$, $\lambda_2 \in [0,1]$. The values of λ are normally set empirically and depend on the individual document corpus. For example, if a domain only contains a small number of concepts, then it is highly likely that all these concepts have a strong connectivity with each other, and thus the informative coefficient plays a more critical role in deciding the semantic impact. In this case, the system could be assigned a bigger number to λ_1. (e.g. 0.8) and a smaller number to λ_2 (e.g. 0.2) to reduce the overall contribution of the CC.

4.6 Summary and Moving Forward

In summary, the Semantic Impact (SI) value is a combination of the informativeness, which is measured by the Informative Coefficient (IC), of a concept and its connectivity strength, which is measured by the Connectivity Coefficient (CC), with the other concepts within the domain knowledge.

The IC for a specific concept is calculated by implementing the Neural Complex to represent its informativeness as the consistency of the (concept's) semantic distribution between Source and Target Corpus. The CC is calculated by means of a complex application of the Maximal Information Coefficient (MIC) algorithm.

In order to evaluate the SI algorithm discussed in the chapter, two experiments have been conducted in this thesis. In general, the first experiment has a limited scale and depth within a simplified scenario. The main purpose is to prototype the idea of the SI. Compared with the second experiment, it will use a smaller set of documents with a more closely related guiding ontology. In contrast, the second experiment has a much larger document set with a more distanced (but still related) guiding ontology to properly evaluate this proposed algorithm.

There are also a few modifications in the second experiment to further improve the algorithm itself and the related processes. For example, the mapping process, which was discussed in Section 4.2.1, is purely a manual process in the first experiment, but a dynamic and automatic process has been implemented in the second experiment by liaising with another data source (we will provide detailed discussion in Chapter 6). The second experiment has a more thorough evaluation as well compared to the first. However, it is necessary to include the first experiment in this thesis, not only because it is a simplified version that is easier to explain and understand, but also because these two experiments represent two different application areas: the first one is about expanding the knowledge associated with the guiding ontology since the guiding ontology is really close to and almost part of the target area, while the second one is more about transferring the knowledge from the guiding ontology to the target area. It clearly demonstrates that the SI algorithm can be applied in both cases.

Detailed discussions about these two experiments will be provided in the following two chapters.

Chapter 5 Experiment One – “Donald Trump” within News Domain

5.1 Overview

As mentioned previously, the first experiment is about prototyping the idea of the SI. 200 news articles about “Donald Trump” between February 2017 and September 2017 were manually collected from the BBC News website and split into two corpora: Source Corpus and Target Corpus. The guiding ontology used in this experiment was the BBC Core Concept Ontology (version 1.1.3)[22]. Essentially, this experiment tried to assess the importance and relevance of the concepts within the News domain from the “Donald Trump” perspective. This experiment has been published as a conference paper [11][23].

The previous chapter explained how to calculate the SI in detail. However, a few parts were missing from the discussion (e.g. how to build the DbO), because two experiments were conducted in this thesis, and the actual implementation for those parts was slightly different within these two experiments. So, this chapter will focus on the first experiment and only discuss those missing parts in detail instead of re-explaining how each process/step (Figure 4-1, p.54) works. However, the related outcome from each step will be included in this chapter, followed by a brief discussion about the result.

[22] https://www.bbc.co.uk/ontologies/coreconcepts A generic ontology produced by BBC.
[23] The final result in this thesis is slightly different from the result in the conference paper. This is because we have changed the method slightly in this thesis.

5.2 Step 1 (Figure 4-1) - Exploratory Semantic Analysis (ESA)

The purpose of the ESA is to extract various semantic information from the corpora and build a Distributional Semantic Model (DSM) for each individual (valid) concept identified from source and target corpora and then collect those DSM into two sets accordingly – Source DSM Set and Target DSM Set as shown in Figure 4-1 (p. 54).

5.2.1 Semantic Information Extraction

The semantic information extraction is the first subprocess included in this step. It can further split into three tasks.

5.2.1.1 Entity and Concept Extraction

The first task of the ESA step is to use an existing Named Entity Recognition (NER) tool/method to extract various information about entities (e.g. John, University of Birmingham) together with which concept (the IBM NLU entity type) (e.g. Person, Organisation) they belong to. Within the first experiment, the IBM Watson Natural Language Understanding (NLU) service [63] with the default News annotation model was selected to analyse these documents and extract various items of semantic information (concepts and relations) from them.

The reason for selecting IBM NLU is simple: it is an out-of-the-box tool people can use to extract semantic information from a corpus with a good performance. For example, a comparison study was done in [64], and the IBM NLU system

achieved an average of 84.95 F1 score in the intent classification[24] test and outperformed other considered systems (Dialogflow, Rasa and LUIS).

The IBM NLU provides an easy way to extract meta-data from content such as “concepts/entity types” and “relations” [65]. Clarification for some of these terms, which are related to this research, is shown below:

- Entity Types: Identify the entity type (which are essentially the concept information) in the text. For example, Anatomy, Award and Company.
- Relations: Recognise when two entities are related and identify the type of relation. For example, an "awardedTo" relation might connect the entities "Nobel Prize" and "Albert Einstein".

“Relations” is the key information that has been used in this experiment. However, instead of using the identified relation type (e.g. awardedTo), the system places greater weight on the entity and entity type information for each of the related entities.

For example, by using the IBM-NLU process, 438 relations can be identified from an article [66] in the corpus. Below is an example of how a relation is constructed:

[24] Intent classification is a process that categorise texts or sentences based on user’s intent by analysing the language they use.

Relation


```
    {
      "type": "agentOf",
      "firstEntityType": "Person",
      "secondEntityType": "EventCommunication",
      "secondEntity": "said",
      "firstEntity": "Sean Spicer",
      "sentence": "Before the list was published, press secretary Sean
Spicer said there were \"several instances\" of attacks that had not gained
sufficient media coverage (without specifying which fell into that
category).",
      "score": "0.99692"
    },
```


The `firstEntityType` and `secondEntityType` in the above example are considered as the concepts; subsequently, `firstEntity` and `secondEntity` are considered as the entity names. In other words, two concepts have been identified from the above example: `Person` with an associated entity “Sean Spicer”, and `EventCommunication` with an associated entity “said”.

An annotation mode is required in order to identify these entity types (concepts). There is a default annotation model (about the news) in the IBM-NLU service for testing purposes. The `Person` and `EventCommunication` concepts identified in the above example are part of that annotation model. One of the limitations of IBM-NLU is that it cannot identify additional concepts outside the existing annotation model without having a new customised annotation model. It is time-consuming to build a new annotation mode from scratch. So in this experiment, we are using the default news annotation model. However, people need to use the Watson Knowledge Studio to train a new annotation model, which is time-consuming, if they want to use IBM-NLU to extract concepts that have not been included in the default news annotation model.

The `sentence` field is the original sentence where this specific relation has been identified, and the score is a confidence score about this extraction. The nearer it approaches 1, the more accurate this extraction should be.

5.2.1.2 Mapping Process

The second task is the mapping process. Considering all the concepts identified from the corpus (by using the default annotation model) as the corpus concepts, then the next task is to map them to the guiding ontology as discussed in Section 4.2.1.

The mapping process in this experiment is a manual process, mainly because there are only four valid (i.e. having a valid mapping in the corpus concepts) concepts in the guiding ontology (those ontology concepts are, in fact, ontology classes, we call them concepts for convenience), which are `Person`, `Place`, `Event` and `Organisation`.

By going through the IBM-NLU extraction process with all the documents in the Source and Target Corpus, 35 corpus concepts were identified within each corpus, and 34 of them were shared between the Source and Target Corpus. By using the mapping relation (manually generated) shown in Table 5-1, those 34 concepts were converted into 29 concepts shown in Table 5-2, which will be considered as valid corpus concepts for further processing. Basically, if a corpus concept has a valid mapping in the guiding ontology, then use the associated ontology concept to replace the original corpus concept. For example, “EventMeeting” (in the first column of Table 5-1) is a corpus concept that has a valid mapping in the guiding ontology – mapped to “Event”. Hence

the system will replace “EventMeeting” with “Event” (therefore “EventMeeting” is not in Table 5-2).

Corpus Concepts	Guiding Ontology Concepts
Organisation	Organisation
Person	Person
GeopoliticalEntity	Place
EventCommunication	Event
EventMeeting	Event
EventLegal	Event
Location	Place

Table 5-1 Mapping relation between corpus concepts and guiding ontology (BBC Core Concept Ontology) concepts

Award	Cardinal	Crime	Date
Duration	EntertainmentAward	***Event***	EventBusiness
EventCustody	EventDemonstration	EventEducation	EventElection
EventPerformance	EventPersonnel	EventViolence	Facility
GeographicFeature	HealthCondition	NaturalDisaster	***Organisation***
Person	***Place***	Product	SportingEvent
Substance	Time	TitleWork	Vehicle
Weapon			

Table 5-2 Valid Corpus Concepts

There are a few things that need to be highlighted here. Firstly, it might be objected at this point that the manual mapping process implemented here seems to contradict one of the overall objectives for this research -- to reduce the level of human input. There are two reasons for using the manual process: a) the guiding ontology only contains a small number of ontology concepts, and most of them have a clear mapping relation with the corpus concepts (e.g. organisation → organisation; person → person; location → place). Hence, human involvement here is tiny compared with the overall process. b) We would like to measure what impact the mapping process could generate on the final SI result, which is, in fact, the second thing that needs to discuss here.

It may have been noticed that within the corpus concepts, only `EventCommunication`, `EventMeeting` and `EventLegal` are mapped to the `Event` ontology concept/class, as shown in Table 5-1 above. However, there are another nine corpus concepts in Table 5-2 that have “Event” as the

root word (e.g. `EventBusiness` and `EventCustody`), but have not been mapped to any ontology concepts at all. We deliberately designed it in this way to assess the positive or negative impact the mapping process could bring to the final SI result. More specifically, we are trying to measure the final SI result for those concepts with the "Event" root word to investigate if there is a grouping trend between them. This will be discussed in more detail at the end of this chapter.

5.2.1.3 DbO Construction

The final task within the semantic information extraction process (Figure 4-2, p. 59) is to convert the semantic information extracted from the corpora into a lightweight ontology format – Document-based Ontology (DbO) as discussed in Section 4.2.2. There are three components within a DbO: Ontology Class, Ontology Property and Ontology Individual.

5.2.1.3.1 Ontology Class

Denote a valid corpus concept (in Table 5-2) as α and an ontology concept/class in the guiding ontology as β. If α is a new class (which means there is not a mapped β in the guiding ontology) then the system will create a new ontology class in this specific DbO with three statements, as the following example shows:

```
DbO:Weapon  a               owl:Class ;
       rdfs:isDefinedBy  DbO:0e0e6bf58a95f44aee0f937e33a2532b ;
       rdfs:label        "Event"@en .
```

where 0e0e6bf58a95f44aee0f937e33a2532b is the name of this specific DbO, which is essentially the MD5 value of the document's name.

However, if a β exists in the guiding ontology which maps to α, then the system will run a recursive process to iterate through all the subclasses of β in the guiding ontology and then cross-reference with all the valid corpus concepts that have been identified within this specific document: Let $A = \{\alpha_1,\ \alpha_2,\ \alpha_3, \dots, \alpha_n\}$ be the set of all the valid corpus concepts, and $B = \{\beta_1,\ \beta_2,\ \beta_3, \dots, \beta_n\}$ be the set of all the sub-classes of β. If $\beta_n \in A$ then β_n will also be a sub-class of α. Then list all the subclasses of β_n and run this process again to identify the complete hierarchic structure. In this case, α will be completely replaced by β and the associated subclasses.

This approach works well when there is a clear mapping relation between the corpus concepts and the guiding ontology concepts, as we have in this experiment. However, in the case where the mapping relation is not clear (a specific corpus concept has been mapped to multiple ontology concepts), then it may cause issues. A more detailed discussion will be provided in the next chapter.

5.2.1.3.2 Ontology Property

There are six pre-defined ontology properties (annotationproperty) within the DbO, which are: FirstEntity, FirstEntityType, SecondEntity, SecondEntityType, Score and Sentence.

DbO is based on OWL, and there are standard ways to declare annotation type property. Below is an example of the way to declare the FirstEntity property in a DbO.

```
Property:FirstEntity  a  owl:AnnotationProperty ;
        rdfs:comment  "The first Entity in the relation."@en ;
        rdfs:label    "FirstEntity"@en ;
        rdfs:range    xsd:string .
```

5.2.1.3.3 Ontology Individual

The individual is the last component that a DbO contains. Basically, each individual represents one Relation output from the IBM-NLU service. Below is an example of an individual based on the Relation information shown in the previous section.

```
DbO:e8c321de4f3ace4689b10668017d592d_0.99692
        a                           DbO:Event , DbO:Person ;
        DbO:occupation              "agentOf" ;
        Property:FirstEntity        "Sean Spicer" ;
        Property:FirstEntityType    DbO:Person ;
        Property:Score              "0.99692" ;
        Property:SecondEntity       "said" ;
        Property:SecondEntityType   DbO:Event ;
        Property:Sentence           "Before the list was
published, press secretary Sean Spicer said there were \"several
instances\" of attacks that had not gained sufficient media
coverage (without specifying which fell into that category)." .
```

The first line is the name of this individual, which is a combination of the MD5 value of the sentence and the confidence score (extracted from the Relations discussed in Section 5.2.1.1). The reason to name it in this way is that multiple relations could be identified from the same sentence. Hence we need to have a way to differentiate them.

By applying this method to all the identified relations within a specific document, the system will then be able to generate the associated DbO. Subsequently, repeat this process to generate, in total, 200 DbOs and collect them into Source DbO Set and Target DbO Set. An example of a full DbO document is shown in Appendix I.

5.2.2 DSM Construction

As discussed in Section 4.2.2, for a specific concept, a word-replacement process is used to replace all the associated words from the original text with an invented unique string (which represents that concept), and then re-run the Word2Vec process to generate a new Word2Vec model to represent that specific concept (`W2V_<ConceptName>_Source/Target`).

Within this thesis, the system that has been developed to calculate the SI is based on (or using) the DeepLearning4J (DL4J) framework [67] (version 1.0.0-beta3). DL4J[25] is an open-source, distributed deep-learning library written for Java and Scala.

The Word2Vec (vectorisation) process in both experiments is handled by the related module in the DL4J framework with the following configuration: `MinWordFrequency` = 1, `LayerSize` = 100, `WindowSize` = 5, `Iterations` = 100 and `Seed` = 42.

`MinWordFrequency` is the minimum number of times a word must appear in the text. Normally, a word should appear multiple times in the text before a useful feature/context can be accurately captured by the Word2Vec algorithm. Therefore, in an ordinal Word2Vec application, this value is normally bigger than 5. The reason to set it to 1 in this experiment was to maximise the vocabulary shared between two Word2Vec models, which is essential to the Coordinate Transformation process (discussed in Section 4.3.2).

[25] https://deeplearning4j.org/

`LayerSize` is the feature size of the generated word vectors, in other words, the dimensions of the vector. In general, the larger it is, the higher accuracy it will attain; in the meantime, the longer it will take to train the model. It was set to 100 in this experiment as a compromise between accuracy and training time.

`Iterations` is the number of times allowed to update the coefficients in a model for one batch of the data. Similar to `LayerSize`, a large iteration number means a higher accuracy at the cost of the training time.

`WindowSize` is the context window, and `Seed` is used for random number generation.

By going through each individual concept in both the source corpus and target corpus, the system will be able to generate the related Word2Vec models (`W2V_<ConceptName>_Source/Target`) and then collect them into the corresponding source or target DSM set. The vocabulary size for those models are different because a) the Source and Target Corpus contain different documents, and b) the words replaced by the word-replacement process are different from model to model. However, the difference should not be huge. For example, the vocabulary size of the `W2V_Universal_Source` model is 9963, and the size of the `W2V_Universal_Target` model is 9518.

5.3 Step 2 (Figure 4-1) - Informative Coefficient Calculation

Figure 4-11 (p.81) in the previous chapter already explained the six steps to implement the Neural Complex to calculate the CS' value for each concept. There are five ontology concepts included in the guiding ontology, but only four of them have appeared in the corpora used in this experiment, which are:

`People`, `Place`, `Event` and `Organisation`. Hence, these four ontology concepts were used as the Mapped Subset created in Step A to identify the best neural network (NN) structure for this experiment. This section will start with an introduction to how to find the best NN structure.

5.3.1 Determine the Best NN Structure

Equations 4-4 and 4-5 (p. 80) in the previous chapter explained how to calculate the CS and CS' values for a specific concept. In fact, there are two additional parameters that can be used to assess the result of the Neural Complex:

1. The average cosine similarity for the common words between the `W2V_<ConceptName>_Source` model and the `W2V_<ConceptName>_Target` model, denoted as $\overline{CS}_{CW}$ and is calculated by the following equation:

$$\overline{CS}_{CW} = \frac{\sum_{i=1}^{n} CS(\vec{V}_{i\_Source}, \vec{V}_{i\_Target})}{n} \qquad 5\text{-}1$$

   where n is the number of common words between the `W2V_<ConceptName>_Source` model and the `W2V_<ConceptName>_Target` model. $\vec{V}_{i\_Source}$ is the vector for the i[th] common word and is included in the former model, and $\vec{V}_{i\_Target}$ is the equivalent vector in the latter model.

2. The average aligned cosine similarity for the common words between the source and the target model, which is denoted as $\overline{CS}'_{CW}$.

$$\overline{CS}'_{CW} = \frac{\sum_{i=1}^{n} CS(\vec{V}'_{i\_Source}, \vec{V}''_{i\_Source})}{n} \qquad 5\text{-}2$$

   where $\vec{V}'_{i\_Source}$ is the aligned vector for the i-th common word. As with how to generate $\vec{V}'_{People\_Source}$ in Step 2.4 (Figure 4-9, p. 79), using

$\vec{V}_{i\_Source}$ as the input of the neural network that trained from Step 2.2 (Figure 4-7, p. 77), then the output of that neural network is $\vec{V}'_{i\_Source}$. Respectively, $\vec{V}''_{i\_Source}$ is calculated by replacing $\vec{V}_{People\_Source}$ in Step 2.5 (Figure 4-10, p. 80) with $\vec{V}_{i\_Target}$.

In the best-case scenario, where a perfect NN structure has been identified, the $CS_{<ConceptName>}$ should be a random value between -1 and 1 (because it is before the alignment process, so the two vectors do not belong to the same coordinate system). The $CS'_{<ConceptName>}$ should equal to 1 for those informative concepts, but be randomly distributed between -1 and 1 for those non-informative concepts. The $\overline{CS}_{CW}$ should equal to 0, because similar to $CS_{<ConceptName>}$, an individual CS_{CW} value should be randomly distributed between -1 and 1, hence their average will be equal to 0. The $\overline{CS}'_{CW}$ should always be 1, because the associated neural networks are trained based on those common words (CT process discussed in Section 4.3.2). In other words, $\overline{CS}'_{CW}$ represents the average training result. If a neural network has been trained properly, then for any given words in the training dataset, its $\vec{V}'_{i\_Source}$ should overlap with its $\vec{V}''_{i\_Source}$ completely (CS = 1).

For a given candidate NN structure, suppose that it is applied to the Mapped Subset (`Person`, `Place`, `Event` and `Organisation`) and that $CS_{<ConceptName>}$, $CS'_{<ConceptName>}$, $\overline{CS}_{CW}$ and $\overline{CS}'_{CW}$ are calculated. The results are plotted in Figure 5-1 below:

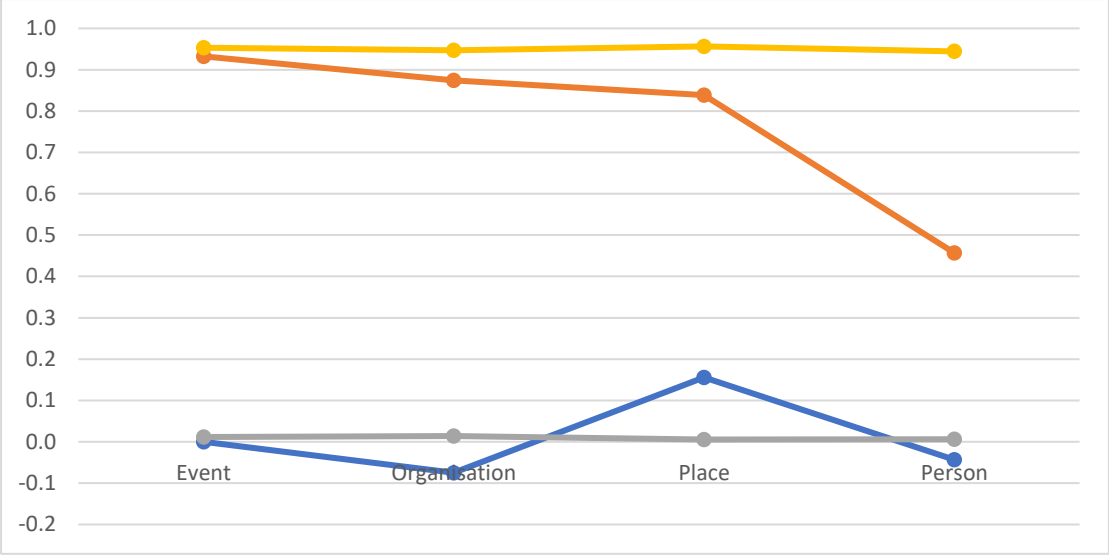


Figure 5-1 NN Structure Result 1 – 3 hidden layers and 2000 nodes on each layer

where the blue line represents $CS_{<ConceptName>}$, the orange line represents $CS'_{<ConceptName>}$, the grey line represents $\overline{CS}_{CW}$ and the yellow line represents $\overline{CS}'_{CW}$. The NN structure in this specific example contains three hidden layers, and each layer contains 2000 nodes.

Figure 5-1 is, in fact, the best NN structure identified in this experiment based on the results from six different structure tests. For example, Figure 5-2 is the result obtained from a structure that only contains 1000 nodes. Compared with the result in Figure 5-1, it clearly shows that both the overall neural network training result, $\overline{CS}'_{CW}$ represented by the yellow line, and the $CS'_{<ConceptName>}$, represented by the orange line, drop slightly. Hence this result is less good than the first one.

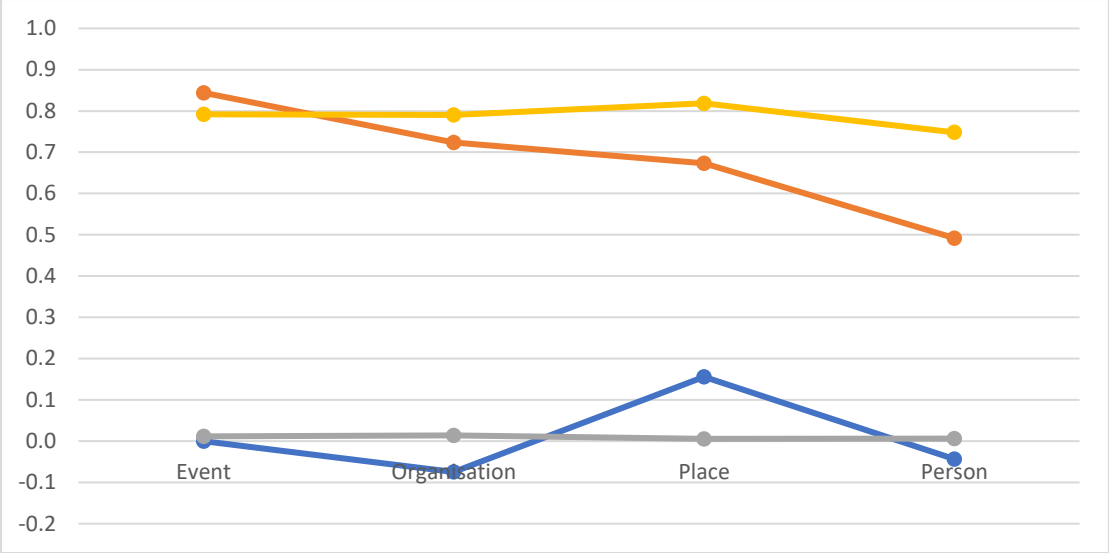


Figure 5-2 NN Structure Result 2 – 3 hidden layers and 1000 nodes on each layer

The other NN structures that have been tested here include the structure resulting from making the following changes (HL= number of hidden layers, Nodes = number of nodes on each layer, unless more than one number specified):

- Reduce the number of nodes to 500 (Figure 5-3).
- Reduce the number of nodes to 100 (Figure 5-4).
- Instead of having the same node number on all hidden layers, reduce the first and last hidden layer nodes to 500 (nodes) but keep the second as 2000 (Figure 5-5).
- Similar to the last test but reduce the second hidden layer nodes to 50 (Figure 5-6).

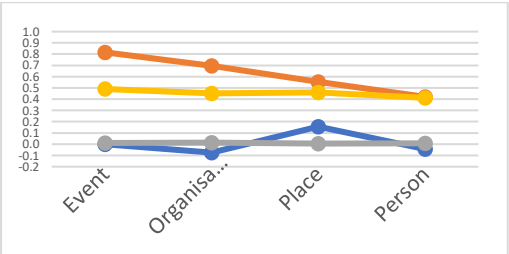


Figure 5-3 NN Structure Result 3 –HL = 3, Nodes = 500

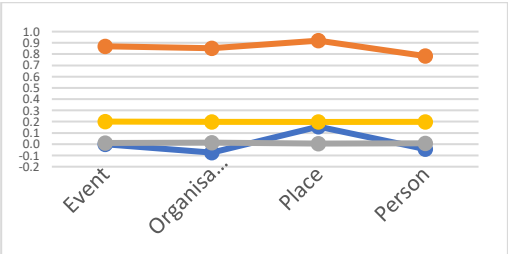


Figure 5-4 NN Structure Result 4 -- HL = 3, Nodes = 100

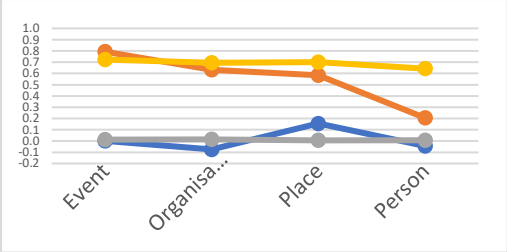


Figure 5-5 NN Structure Result 5 – HL = 3, Nodes = 500,2000,500

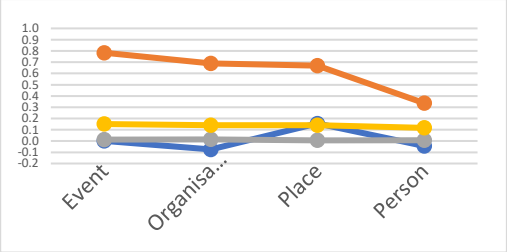


Figure 5-6 NN Structure Result 6 – HL = 3, Nodes = 500,50,500

Within all the tests above, the blue lines and the grey lines are barely changed at all. This is aligned with the expectation. The blue line represents $CS_{<ConceptName>}$, which is the cosine similarity of the concept's semantic distribution between the Source and Target Corpus ($\vec{V}_{<ConceptName>\_Source}$ and $\vec{V}_{<ConceptName>\_Target}$) before the alignment. $\vec{V}_{<ConceptName>\_Source}$ and $\vec{V}_{<ConceptName>\_Source}$ are from `W2V_<ConceptName>_Source` model and `W2V_<ConceptName>_Target` model; moreover, those Word2Vec models were generated from the DSM construction process (Section 4.2.2) and remain unchanged across the whole process. This is why the $CS_{<ConceptName>}$ values are the same in all the tests.

For the grey lines, they all close to 0, which suggests that the CS_{CW} values are randomly distributed, as explained above.

In fact, this experiment only uses the yellow and orange lines as the criteria to decide if the result is good enough (Step D in Figure 4-7, Section 4.3.3.2). The blue and grey lines will be used as the offset in the second experiment to

calculate the Alignment Coefficient (AC), which will be discussed in the next chapter.

It is difficult to determine which is more important between the yellow line and the orange line. However, one of the principles is that we need to ensure there is a good $\overline{CS'}_{CW}$ value (close to 1) first before considering the $CS'_{<ConceptName>}$ value as the former represents the accuracy of the neural network on the training dataset, in other words, the training result. A low $\overline{CS'}_{CW}$ value means the neural network has not been trained properly yet. Hence, the $CS'_{<ConceptName>}$ value might be inaccurate even when it is close to 1. In this case, the result should be considered bad (e.g. Figure 5-4), and the system will return to Step B (Figure 4-14, p. 86) to modify the NN structure and try again.

As mentioned already, the NN structure, which is shown in Figure 5-1 (3 hidden layers and 2000 nodes on each layer) can generate the best result, and therefore is considered as the best NN structure. By applying it to the rest of the concepts, the system will then generate a CS' for all the concepts (Step E in Figure 4-14, p. 86).

A further discussion is needed here to explain the best NN structure identified for this experiment.

In traditional Machine Learning or Deep Learning study, the number of nodes on a hidden layer is normally smaller than the input layer's size to compress the input value, and gradient descent (or something similar) is used to find the global minimum. If the number of nodes is larger than the number of nodes in the input layer (feature size of the Word2Vec model), there is a chance that the

neural network could simply copy and paste the value from the previous layer and then pass it to the next hidden layer without processing it at all. Hence, a recommended approach when designing a neural network structure is to go deep and keep it narrow [68].

However, the best NN structure in this experiment is on the opposite side of the recommendation: fat (20 times bigger than the input layer) and shallow (only three hidden layers). The main reason is because of the unique implementation plan of the Neural Complex discussed in Section 4.3.3.2. After all, the ultimate goal is to measure the overfitting-ness of the concepts instead of avoiding overfitting. Therefore, the NN structure is deliberately designed in this way (fat and shallow) to make it easy to be overfitted.

The selected structure is identified by a Mapped Subset, which contains informative concepts as discussed previously. To ensure it can be used to properly train neural networks associated with the rest of the concepts, we have calculated the $\overline{CS}'_{CW}$ value for all the 29 concepts and the result is shown in Table 5-3 below.

Concepts	$\overline{CS}'_{CW}$
Person	0.99137
Event	0.98853
Place	0.98767
Crime	0.98225
Date	0.98124
Time	0.98072
GeographicFeature	0.98052
EventViolence	0.98037
Vehicle	0.98035
Award	0.98032
Facility	0.98026
Cardinal	0.97967
EventEducation	0.97961
Duration	0.97950
HealthCondition	0.97929
NaturalDisaster	0.97914
EntertainmentAward	0.97891
EventBusiness	0.97870

Concepts	$\overline{CS}'_{CW}$
EventDemonstration	0.97865
EventPerformance	0.97859
EventCustody	0.97858
EventElection	0.97835
SportingEvent	0.97821
EventPersonnel	0.97808
TitleWork	0.97799
Weapon	0.97642
Product	0.97634
Substance	0.97603
Organisation	0.97541

Table 5-3 Concepts' average cosine similarity value for common words between the Source and Target Corpus

It clearly demonstrates that all the associated neural networks have been trained properly, as all the $\overline{CS}'_{CW}$ values are close to 1.

Another way to evaluate the outcome of the alignment is to compare the top 10 closest words for a shared word between a Word2Vec model before and after the alignment. If the alignment works, those top 10 words should be similar (e.g. the same words in the same order).

For a selected word in the `W2V_Universal_Source` model, it is easy to get the top 10 closest words for it, which are denoted as $W_{Universal}$. Moreover, it is easy to get the same information from the `W2V_<ConceptName>_Source` models (created by the word-replacement process discussed in Section 4.2.2), which is denoted as $W_{<ConceptName>}$ and their associated `Aligned W2V_<ConceptName>_Source` models (by using the trained neural network to generate predicted vectors), denoted as $W'_{<ConceptName>}$.

It is easy to understand that the $W_{<ConceptName>}$ and $W_{Universal}$ should be different from each other since the actual text used to produce the Word2Vec models are different (because of the word-replacement process). If the alignment process works, the semantic representation within the

W2V_Universal_Source model and the W2V_<ConceptName>_Source models should be similar, hence $W'_{<ConceptName>}$ should similar to $W_{Universal}$.

Using the word "Trump" and running the test on the Mapped Subset (Person, Event, Organisation and Place), the result is shown below.

Top 10 the closest words for "Trump"	Word2Vec Model
[mr, his, said, but, was, for, that, president, has, he]	*W2V_Universal_Source*
[his, was, he, president, xoxovveventvvoxox, mr, had, has, for, would]	*W2V_Event_Source*
[sir, --, you, ok, figured, checked, fake, skipping, nextph, thats]	*W2V_Org_Source*
[--, ok, finish, jim, skipping, yeah, figured, do, excuse, heres]	*W2V_Person_Source*
[mr, said, his, president, was, but, he, that, on, -]	*W2V_Place_Source*
[mr, his, but, was, has, president, he, for, not, that]	*Aligned W2V_Event_Source*
[mr, his, said, was, for, has, that, on, he, it]	*Aligned W2V_Org_Source*
[said, but, for, was, that, -, not, has, on, it]	*Aligned W2V_Person_Source*
[mr, his, said, was, but, has, he, that, it, for]	*Aligned W2V_Place_Source*

Table 5-4 Result for the Top 10 vocabulary test

The first row (ignoring the heading) marked with green is the top 10 closest words (to the word "Trump") in the W2V_Universal_Source model generated directly from the original source corpus without going through the word-replacement and CT process ($W_{Universal}$). It is used as the benchmark in this test.

From the 2nd to the 5th row (marked with yellow) represent the same information in the related W2V_<ConceptName>_Source models, which have been generated by the word-replacement process without being aligned (i.e. having gone through the CT process) ($W_{<ConceptName>}$). As discussed above, since the texts used to generate these models were different, so the results are quite different from the benchmark row.

The last 4 rows (marked with blue) are the results from the Aligned W2V_Universal_Source models ($W'_{<ConceptName>}$). Compared with those

yellow rows, the results of the blue rows are much closer/similar to the benchmark. This suggests that those aligned models have a similar semantic representation to the universal model, and therefore the related neural networks, which are trained based on the identified NN structure, work as expected.

There are a few stop words in the green line. This is because the corpus only contains 100 documents. With a larger corpus size, the model should be able to have a more accurate semantic representation, which will remove stop words from the above list.

5.3.2 Final IC Calculation

As Equation 4-3 shown in the previous chapter, IC is the product of the $CS'_{<ConceptName>}$ and the associated confidence score -- $\overline{Conf}_{<ConceptName>}$. In this experiment, the way to calculate the confidence score for a specific concept was by enumerating all the `Ontology Individuals` (Section 5.2.1.3) which contain at least one of that specific class[26] in the Source DbO set, then obtain the sum of the score (`Property:Score` in the `Individual` as shown in Section 5.2.1.3) which is the relation confidence score obtained from the IBM-NLU process and ranging from 0 (not confident) to 1 (highly confident). Let n be the total number of the `Ontology Individual` which contains that specific concept, $Score_i$ be the associated score value, then $\overline{Conf}_{<ConceptName>}$ is calculated as:

$$\overline{Conf}_{<ConceptName>} = \frac{\sum_{i=1}^{n} Score_i}{n} \qquad 5\text{-}3$$

[26] In this experiment, an Ontology Individual (Section 5.2.1.3.3) contains two classes: FirstEntityType and SecondEntityType.

The final IC result is shown in Table 5-5 below (sorted by IC):

Concept/Class	CS'	Confidence Score	IC
Event	0.93240	0.82422	0.76851
Date	0.81636	0.80618	0.65813
Organisation	0.87448	0.67535	0.59058
Place	0.83836	0.67033	0.56197
Cardinal	0.77228	0.64542	0.49844
EventPerformance	0.59270	0.71078	0.42128
EventViolence	0.65090	0.62525	0.40697
Person	0.45689	0.71154	0.32509
EventPersonnel	0.46688	0.68484	0.31974
EventCustody	0.29303	0.70333	0.20610
EventBusiness	0.27681	0.74371	0.20586
NaturalDisaster	0.19095	0.51532	0.09840
SportingEvent	0.11371	0.73879	0.08400
Product	0.08085	0.91436	0.07392
Weapon	0.17078	0.39204	0.06695
GeographicFeature	0.12444	0.43846	0.05456
EventElection	0.08788	0.61177	0.05376
EntertainmentAward	0.08944	0.57733	0.05164
Facility	0.04609	0.67142	0.03094
EventDemonstration	0.05316	0.49458	0.02629
HealthCondition	0.01494	0.73245	0.01094
Award	0.00989	0.58031	0.00574
Duration	0.01585	0.35558	0.00564
Vehicle	-0.00800	0.59239	-0.00474
Substance	-0.09238	0.20504	-0.01894
TitleWork	-0.07366	0.65029	-0.04790
Time	-0.07930	0.68005	-0.05393
Crime	-0.07948	0.70386	-0.05594
EventEducation	-0.17788	0.48692	-0.08661

Table 5-5 Final IC results

Based on the above table, `Event` has been recognised as the most informative concept. Intuitively, this is correct since the news article is essentially a description of a series of events.

5.4 Step 3 (Figure 4-1) - Connectivity Coefficient Calculation

The Maximal Information Coefficient (MIC) calculation process in the two experiments is the same and is exactly as described in Chapter 4. So this section will only show the outcome of this process.

There are 29 valid concepts identified from the corpus, as shown in Table 5-2, so the total number of the concept pairs is 406. Moreover, the vocabulary size

of the `W2V_Universal_Source` model is 9963. Hence, the size of the sample table (Section 4.4.2) is 9963 x 406. Using it as the input of the MIC algorithm, we can then generate the MIC result as Table 5-6 shows below. Due to the reason of size, only the top 20 pairs are included in the below table, and the full list is available in Appendix II.

X var	Y var	MIC (strength)
Cardinal	Date	0.37431
Cardinal	Facility	0.35607
Date	Facility	0.32850
EventViolence	Facility	0.30661
EventViolence	Cardinal	0.30093
EventElection	Cardinal	0.28417
Crime	Cardinal	0.28044
EventViolence	Date	0.27788
EventViolence	Crime	0.27470
Organisation	Place	0.26798
Cardinal	EventPersonnel	0.26564
EventElection	EventPersonnel	0.25349
Crime	Date	0.25226
EventElection	Date	0.25098
Crime	Facility	0.25015
EventElection	Facility	0.24460
EventPersonnel	Date	0.24420
EventPersonnel	Facility	0.24282
TitleWork	Facility	0.23623
EventPerformance	EventPersonnel	0.22386

Table 5-6 Top 20 concept pairs in the Source Corpus

It is interesting to see that the highest MIC value is between the (Cardinal, Date) pair. The concept of Cardinal defined in the IBM-NLU refers to numbers (short for cardinal number) instead of a high-rank priest in the religious context. It can include numerical values/entities (e.g. 22, 19 and 13), individual word (e.g. some, many and thousands), as well as short phrases (e.g. hundreds of thousands).

Intuitively, the concept of `Date` is also a numerical based representation, and therefore it is correct. Moreover, `Cardinal` is such a generic concept, and quite a lot of things could somehow have a relation with the number (e.g. the number of the facility, the number of victims in a violent event). This explains why `Cardinal` appears so many times in the top 20 list. Hence this is a promising result since the MIC algorithm itself is a statistical-based method and therefore does not know the compositions of these two concepts.

The final Connectivity Coefficient ($\mathcal{CC}$) value can then be calculated based on Equation 4-7 introduced in the previous chapter. Table 5-7 is the full $\mathcal{CC}$ result.

Concept	Connectivity Coefficient
Date	2.38722
Cardinal	2.03571
Facility	2.01725
Crime	1.82121
EventPersonnel	1.81814
EventViolence	1.70345
EventElection	1.65376
EventPerformance	1.65298
Event	1.59453
TitleWork	1.51326
Organisation	1.40307
EventCustody	1.39910
Person	1.39871
Award	1.38673
HealthCondition	1.36997
Place	1.35907
Vehicle	1.34433
Product	1.33561
EventBusiness	1.26135
Time	1.20069
SportingEvent	1.19619
EntertainmentAward	0.83280
NaturalDisaster	0.75349
Weapon	0.75273
EventDemonstration	0.69887
GeographicFeature	0.67286
EventEducation	0.66633
Duration	0.51619
Substance	0.41285

Table 5-7 Final Connectivity Coefficient (CC) result

5.5 Step 4 (Figure 4-1) - Final SI Result and Discussion

As Equation 4-1 (p. 53) suggested, there are two constants λ_1 and λ_2 to adjust the weight of IC and CC. In this experiment, we consider that informativeness and connectivity are equally important for demonstration purposes and therefore $\lambda_1 = \lambda_2 = 0.5$. Hence, the final SI result is shown in Table 5-8 below.

No.	Concept/Class	Normalised IC	Normalised CC	SI	Term Frequency (TF)
1	Date	0.74184	1.00000	0.87092	0.0226484181
2	Event	1.00000	0.19702	0.59851	0.0424237853
3	Cardinal	0.36835	0.64393	0.50614	0.0130624391
4	Organisation	0.58385	0.00308	0.29346	0.1508043171
5	Place	0.51694	-0.04150	0.23772	0.1407177827
6	EventViolence	0.15442	0.30736	0.23089	0.0050604095
7	EventPerformance	0.18788	0.25623	0.22206	0.0013508139
8	EventPersonnel	-0.04961	0.42353	0.18696	0.0048821295
9	Person	-0.03708	-0.00134	-0.01921	0.7169050590
10	Facility	-0.72506	0.62523	-0.04992	0.0206393395
11	EventCustody	-0.31540	-0.00094	-0.15817	0.0007268339
12	EventElection	-0.67169	0.25703	-0.20733	0.0058900972
13	EventBusiness	-0.31594	-0.14048	-0.22821	0.0001577092
14	Crime	-0.92827	0.42665	-0.25081	0.0015085232
15	Product	-0.62453	-0.06526	-0.34489	0.0000274277
16	TitleWork	-0.90946	0.11470	-0.39738	0.0033804633
17	Award	-0.78400	-0.01347	-0.39874	0.0004662708
18	HealthCondition	-0.77184	-0.03046	-0.40115	0.0006239800
19	SportingEvent	-0.60095	-0.20649	-0.40372	0.0005622677
20	Vehicle	-0.80851	-0.05643	-0.43247	0.0042170079
21	Time	-0.92355	-0.20193	-0.56274	0.0004319862
22	NaturalDisaster	-0.56728	-0.65494	-0.61111	0.0003017046
23	EntertainmentAward	-0.67665	-0.57460	-0.62563	0.0001302815
24	Weapon	-0.64083	-0.65571	-0.64827	0.0017622293
25	GeographicFeature	-0.66982	-0.73661	-0.70321	0.0003977016
26	EventDemonstration	-0.73594	-0.71027	-0.72310	0.0005348400
27	Duration	-0.78424	-0.89531	-0.83978	0.0000822831
28	EventEducation	-1.00000	-0.74322	-0.87161	0.0000274277
29	Substance	-0.84173	-1.00000	-0.92086	0.0001165677

Table 5-8 Full result for experiment one

From the above table, the concept `Date`, which is not part of the guiding ontology, is considered as the most important concept (or a concept that can

generate the most impact on the domain knowledge) in the News domain. Intuitively, it is an interesting and correct result.

There are several observations from the above table. Firstly, it is interesting to see that the concept `Person`, which belongs to the guiding ontology and is used as part of the Mapped Subset to identify the best NN structure, is not as important as the others within the Mapped Subset, which are `Event`, `Organisation` and `Place`, although it has the highest $\overline{CS'_{cw}}$ as shown in Table 5-3. This is because it has the lowest IC value among the other ontology concepts. Intuitively, this is correct because all the news articles in the corpora are about Donald Trump, and therefore, the concept of `Person` may not be as generally applied as the other concepts with a higher *Semantic Impact* value which leads to a small *Informative Coefficient* value as the results show. This phenomenon also occurs in the second experiment, where the domain is about Candida, and the concept of `Candida` has a relatively low IC value.

Secondly, there is a grouping trend in the final SI result. In Section 5.2.1.2, it was mentioned that we deliberately excluded some of the concepts with an "Event" root word from the mapping process. In the final result, it is interesting to see that those concepts have been split into a few groups. For example, most of them are ranked between the 6th and the 13th, which are relevantly high ranked. Another two (`EventDemonstration` and `EventEducation`) are ranked at 26th and 28th, which are towards the bottom of the table.

Individually, it is difficult to explain why `EventPerformance` is more important than `EventEducation` in the News domain, but the grouping

phenomenon in the SI algorithm should behave like this: instead of grouping all the related things together into one group, it should split them into several small groups. This is because the SI is calculated from a specific perspective (discussed in Section 3.3), e.g. from the “Donald Trump” perspective within this experiment. Hence, it is unlikely that different types of event are equally important within the perspective. As a result, some of the event types should be more important than some of the others and therefore split into different groups. The same phenomenon has been observed in the second experiment as well, which will be discussed in the next chapter.

Moreover, it also means that the mapping will have an impact on the final SI value. Hence if we mismatch some of the concepts, for example, map `Substance` to `Event`, then the system will generate a different/wrong result. In fact, the system may not be able to identify the best NN structure at all. For example, the second experiment, which will be discussed in the next chapter, does not have a clear mapping and therefore uses protein as an intermediary to build links between the corpus concepts and the ontology concepts. As a result, these corpus concepts are largely overlapped with each other (a large percentage of the associated words are shared between concepts). By using the same process as we have used in this experiment, the system fails to generate a close to 1 aligned cosine similarity for any corpus concept in various tests. It is why we have to implement a different mapping mechanism in the second experiment, which will be discussed in the next chapter.

Thirdly, the SI result is different from the statistical measurement discussed in Section 2.4. For comparison, we have listed the TF value in Table 5-8.

Concepts like `Date` may only appear once in a news article and therefore have a low TF value, but it is, in fact, the most important concept within the News domain as identified by the SI.

5.6 Summary

The first experiment used 200 News articles about “Donald Trump” to assess what are the important concepts within the News domain from the “Donald Trump” perspective. The result was interesting and promising. For example, we have successfully learned that `Date` should be an important concept even if it had not been included in the guiding ontology, then followed by `Event` and `Cardinal`.

We have also provided a primitive evaluation of the IC result (Section 5.3.1), CC result (Section 5.4), and the overall group trend of the final SI result.

As mentioned previously, the first experiment was based on a simplified scenario – only 200 news articles were selected, used a very closely related guiding ontology, and the mapping process was also done manually. In the next experiment, we will expand the scale (10 times larger than the first experiment), make some modifications to the existing algorithm (Section 6.2) and conduct a more systematic evaluation (Chapter 7).

Suppose the first experiment is a prototype to demonstrate the feasibility of the SI algorithm primitively. In that case, the second experiment is a real application of the SI algorithm, and we will explore the importance of various diseases in the Candida domain.

Chapter 6 Experiment Two – Disease within the Candida Domain

6.1 Overview

The first experiment demonstrated how to implement the Semantic Impact (SI) algorithm. One of the most significant drawbacks of the first experiment was the scale of the experiment. In total, there were only 200 documents in the corpora and four valid concepts within the guiding ontology, although it successfully learnt that the concept of `Date` is more important than the others within the news domain, and it served its purpose as a prototype.

Another concern is that the guiding ontology was "too close/similar" to the target domain/perspective. It is fair to say that the first experiment was conducted under a simplified scenario, and its applicability to larger and more varied scenarios may be open to question. To address these concerns, the second experiment not only increased the scale of the experiment but also selected a distanced but still related guiding ontology to assess the SI algorithm properly.

Essentially, 2000 medical articles about Candida were collected from PubMed, together with a Candida Gene Ontology, which acted as the guiding ontology (discussed in Section 4.2.1), to determine the importance/relevance (higher Semantic Impact value) of various diseases (e.g. fungi infections and C-type lectin receptors) in the Candida domain knowledge.

The second experiment followed the same process shown in Figure 4-1 (p. 54) in Section 4.1. However, there are some significant modifications to the actual

implementation of the SI algorithm. This chapter starts with a detailed introduction to these changes and then discusses the new result. The full evaluation of the result will be provided in the next chapter.

6.2 Changes/Modifications

6.2.1 Change of the Scale

The first and most significant change is the expanded scale, not only at the corpus level but also at the guiding ontology level, as shown in Table 6-1.

Aspect	Experiment One	Experiment Two
Corpus size (Source/Target)	100/100	1000/1000
Total classes/concepts in the Guiding Ontology	5	87
Valid classes/concepts in the Guiding Ontology	4	43
Total concepts in the corpus (Source/Target)	35/35	708/735
Valid concepts in the corpus	29	330
Vocabulary size in the universal W2V model (Source/Target)	9963/9518	17334/17345
Mapping relations	7	750+

Table 6-1 Scale comparison

The guiding ontology used in the first experiment was BBC Core Concepts Ontology, which is a generic ontology to describe core concepts that appear on the BBC website. Hence, it was directly linked with the News domain. In this experiment, the domain is Candida, and the selected guiding ontology is an ontology that describes the Candida concepts from the `Gene` perspective. The guiding ontology has no relation to the `Disease` concepts identified from the corpus, except that both `Gene` and `Disease` concepts are from the same corpus about the same domain - Candida.

In theory, the more documents included in the corpora, the better the result should be. However, it is necessary to make a balance between the size of the corpus and the time required to process the SI calculation.

For example, it takes about 80 minutes to train a neural network with the best NN structure identified in this experiment on a computer cluster with 2 x Intel Xeon E5-2683 v4 CPU + 128GB RAM + 2 x Nvidia K80 GPU (using ParallelWrapper to distribute the training workload on two GPUs and run simultaneously. Without GPUs it takes over 10 hours to train a neural network). As discussed in Section 4.3.3.1, the system needs to train $(1 + n \times 2)$ neural networks, where n is the number of assessed concepts, to calculate the IC value. Hence, the system needs to train 661 neural networks in order to complete this experiment, which takes about 881 hours to run on a single cluster with the 2 GPUs spec listed above. This does not include the time needed to identify the best NN structure in the first place. This part of the calculation was done on 4 clusters to run the Neural Complex process (Section 4.3.3) in parallel so as to accelerate the speed.

The Connectivity Coefficient (CC) calculation is also time-consuming. With 330 valid corpus concepts, the system needs to calculate the Maximal Information Coefficient (MIC) value for 54285 concept pairs with over 17000 samples (more than 940 million cells in the sample table discussed in Section 4.4.2). This process took about 12 days on one cluster with 2 x Intel Xeon E5-2683 v4 CPU.

More valid corpus concepts will be identified from the text by increasing the corpus size. Subsequently, it will considerably increase the total time needed for the SI calculation. This is why this experiment included only 1000 documents in each corpus.

The SI algorithm is indeed computationally expensive. However, it does not mean people cannot use it in practice. As with BERT, in practice, we can

produce a pre-trained model to cover most of the common concepts within a domain. People can then do the fine-tuning process at a later stage to cover additional concepts within the domain based on their downstream application. In this case, people do not need to go through the whole process to identify the best NN structure, since it will be provided as part of the pre-trained model. Only two separate NNs need to be trained to calculate the IC value for each additional concept people want to analyse, and it only takes about 80 mins (80 mins to train one neural network, but people can use two clusters to parallelize the workload) in the second experiment discussed here. For the CC value, people only need to calculate the class pairs between that additional concept and the rest of the concepts instead of all the possible class pairs.

It may also be worth mentioning that people can reduce the training time significantly by using the latest hardware. Take BERT as an example, it could take up to 50 days to train the pre-trained model with just a single 4 GPU machine. However, with the help of TPUs (Tensor Processing Units), according to Google Research, it reduces the training time to just over four days on 4 to 16 Cloud TPUs [69].

6.2.2 Change of the Semantic Information Extraction

In the first experiment, the corpus documents were selected from the BBC News website, and the IBM Natural Language Understanding (NLU) was used to extract various entities and concepts from the corpora. In the second experiment, documents were collected from PubMed by searching with the keyword “Candida”.

PubMed is a data source that comprises more than 29 million citations for biomedical literature from MEDLINE[27], life science journals and online books. It has been widely used in computational linguistics/natural language processing studies. More importantly, there is an existing Named Entity Recognition (NER) tool called PubTator [18] offered by the National Center for Biotechnology Information (NCBI), the same institution responsible for PubMed. PubTator has been widely used in bioinformatics related research [70]–[72] and allows the extraction of pre-annotated entities/classes within the PubMed documents. According to [18], by adapting the "disambiguation model", which is essentially a novel convolutional neural network (CNN), the accuracy of PubTator is around 85.2%, and it is significantly higher than the rule-based approach, which is around 55.7%.

Hence, in this experiment, PubTator was used to replace IBM-NLU. Compared with IBM-NLU, the benefits of using PubTator include (but are not limited to):

1. Not having to build/train a new annotation model to identify the Candida domain concepts.
2. A well-accepted solution in this field. A large number of biomedical/bioinformatics research is based on the information extracted from PubTator [70]–[72].
3. Classified entities/classes. PubTator not only contains the annotation information for different entities, but also categorises them into five different types: `Gene`, `Chemical`, `Disease`, `Species` and `Mutation`.

[27] MEDLINE® contains journal citations and abstracts for biomedical literature from around the world. PubMed® provides free access to MEDLINE and links to full text articles when possible.

4. More sustainable. IBM-NLU is solely owned and developed by IBM, but PubTator is steered more towards a community-based approach with many user groups.
5. Free of charge. Unlike IBM-NLU, PubTator is a free and publicly available tool. One of the reasons to only include 100 documents in the first example was because of the limitation IBM set on the free version of the IBM-NLU (only allowing a limited amount of the text to be processed).

The two types used in this experiment are `Gene` and `Disease`. This experiment is essentially about assessing the importance of various `Disease` concepts in the Candida domain knowledge by using the `Gene` concepts identified from the same corpora, together with a well-constructed Candida Gene Ontology (the guiding ontology).

6.2.2.1 Named Entity Recognition (NER) Change

PubTator provides REST[28] APIs to retrieve the related semantic information. Consider the sentence below as an example:

"*This study aimed to investigate the prevalence of **ESR1** mutation in* [Gene (GENE:2099)]
*Chinese primary and metastatic ER-positive **breast cancer**.*" [Disease (MESH:D001943)]

[28] REST is short for Representational State Transfer. It is a common software architectural style used to create interactive applications that use web services.

ESR1 has been identified as a `Gene` concept with an identifier 2099, and breast cancer as a `Disease` concept with an identifier MESH:D001943. The actual result returned from the API is shown below:


```
{
    "id": "34",
    "infons": {
        "identifier": "2099",
        "type": "Gene",
        "ncbi_homologene": "47906"
    },
    "text": "ESR1",
    "locations": [
        {
            "offset": 344,
            "length": 4
        }
    ]
},
{
    "id": "35",
    "infons": {
        "identifier":
"MESH:D001943",
        "type": "Disease"
    },
    "text": "breast cancer",
    "locations": [
        {
            "offset": 404,
            "length": 13
        }
    ]
},
```


where “id” is a unique number assigned to each individual concept identified within the given document. “identifier” and “type” indicates which type (`Gene` or `Disease`) the identified concept is, and its linked object in another data source (NCBI Gene Database for `Gene`, and MeSH for `Disease`).

NCBI has its own Gene database where each gene has been assigned a unique id (denoted as NCBI GeneID). PubTator uses the NCBI GeneIDs to annotate the `Gene` concepts (the identifier mentioned earlier). In the above example, ESR1 is both identified as a Gene concept and linked with a specific gene in the NCBI Gene database (with a unique ID 2099).

As with the `Gene` concepts, PubTator also uses an identifier to link the `Disease` concepts identified from the document with a MeSH object. MeSH stands for Medical Subject Headings -- a controlled and hierarchically organised

vocabulary produced by the National Library of Medicine. It has been widely used for indexing, cataloguing, and searching biomedical and health-related information.

As a consequence of using PubTator to replace the IBM-NLU, two new challenges have been introduced in this experiment. Firstly, there is no "score" information in the PubTator result. As discussed in the previous chapter, there was a "score" value for every relation identified by the IBM-NLU service to indicate how confident the finding is. Moreover, this "score" value will be used, at a later stage, to calculate the $\overline{Conf}_{<ConceptName>}$ value. Since it is missing from the PubTator extraction, the $\overline{Conf}_{<ConceptName>}$ value in this experiment will be calculated differently. Section 6.2.4 will provide more detailed information.

Secondly, it is more difficult to build a clear mapping between the corpus concepts and the guiding ontology concepts. This will be discussed in the next section.

6.2.2.2 Mapping Change

The guiding ontology used in this experiment is the Candida Gene Ontology[29], which is a subset of the Gene Ontology (GO)[30] and is maintained by the Candida Genome Database (DGD)[31]. GO [73][74] provides a computational

[29] http://current.geneontology.org/ontology/subsets/goslim_candida.owl
[30] GO subsets (also known as GO slims) are cut-down versions of the GO containing a subset of the terms. They are particularly useful for providing an overview of the range of functions and processes found in a given clade or organism's genome.
[31] http://www.candidagenome.org/

representation of our current scientific knowledge about the functions of genes (or, more properly, the protein and non-coding RNA molecules produced by genes) from many different organisms, from humans to bacteria. It is widely used to support scientific research and has been cited in tens of thousands of publications [75]. Within the GO, it has its own annotation scheme (e.g. GO:0005618 represents the cell wall), which is denoted as GO ID. The challenge is that there is no direct way to map from NCBI GeneID to GO ID (or the other way around), although GO provides some mapping files to cross-reference to other external classification systems like the High-quality Automated and Manual Annotation of Proteins (HAMAP) [76].

In the first experiment, the mapping process was done manually and did not rely on domain knowledge since the relationships between the corpus concepts and the ontology concepts were clear (e.g. Location → Place, Person → Person). However, there are 87 ontology concepts within the Candida Gene Ontology, and more importantly, we do not have sufficient domain knowledge to manually build the mapping relation between the corpus concepts and the ontology concepts. Hence, a new mapping mechanism was implemented in the second experiment, as discussed below.

Protein is one of the basic building blocks within living organisms and is determined by genes. There has been a long history of studying proteins and there is a much better understanding of them compared to the genes. The UniProt Knowledgebase (UniProtKB) is a collection of sequences and annotations for over 120 million proteins across all branches of life [77]. More importantly, it also contains annotations that are denoted by GO ID. On the

other hand, almost all the genes in the NCBI Gene database (in the related sequences section) have one or more protein accession(s), which are denoted by the UniProtKB ID. Therefore, as also suggested by other researchers in the field [78], including EMBL-EBI themselves, one way to handle the naming/mapping issue is by using the UniProt Knowledgebase (UniProtKB) as the intermediary to indirectly map from NCBI GeneID to GO ID. For example, Figure 6-1 shows the protein accession[32] information for the gene Cblb[33] in the NCBI Gene database, and Figure 6-2 is its GO annotation in the UniProtKB.

Protein Accession	Links	
	GenPept Link	UniProtKB Link
Q3TTA7.3	GenPept	UniProtKB/Swiss-Prot:Q3TTA7

Figure 6-1 Protein accession for Cblb in the NCBI Gene database, where Q3TTA7.3 (Protein Accession) is the name of the protein accession. GenPept can be ignored here since it has not been used in this research at all. UniProtKB Link is the link this specific protein accession has in the UniProtKB database.

Your list:...12455J	Entry	Gene ontology IDs	Protein names
Q3TTA7	Q3TTA7	GO:0001784; GO:0002669; GO:0005509; GO:0005654; GO:0005829; GO:0005886; GO:0006955; GO:0007165; GO:0007175; GO:0017124; GO:0018193; GO:0019901; GO:0030155; GO:0030971; GO:0031398; GO:0035556; GO:0042110; GO:0043087; GO:0043393; GO:0045121; GO:0045732; GO:0046642; GO:0050852; GO:0050856; GO:0050860; GO:0061630; GO:2000583	**E3 ubiquitin-protein ligase CBL-B**

Figure 6-2 GO annotation for Cblb in the UniProtKB, where Q3TTA7 is the name of this protein accession, Gene ontology IDs are the genes (denoted by the GO IDs) that contain this specific protein accession. In this case, Cblb, which is denoted as Gene_208650 in the PubTator, can map to one of the listed Gene ontology IDs based on the fact that this specific protein only exists in those Gene ontology IDs. This is not an accurate mapping since it only provides a list of options. This is why the mapping relationships in this experiment are one-to-many. However, there is a solution to this issue, which will be discussed in the following section (new DbO construction approach).

[32] "An **accession** is a unique identifier given to a Protein sequence record to allow for tracking of different versions of that sequence record and the associated sequence over time in a single data repository. Because of its relative stability, accession numbers can be utilized as foreign keys for referring to a sequence object, but not necessarily to a unique sequence. All sequence information repositories implement the concept of "accession number" but might do so with subtle variations" https://link.springer.com/article/10.1007/BF00332918.

[33] https://www.ncbi.nlm.nih.gov/gene/208650

There are APIs available for both PubTator and UniProtKB. Thus it is possible to automatically convert all the gene concepts extracted from PubTator, into a list of GO IDs that they can potentially map with, then cross-reference with the GO IDs in the Candida Gene Ontology to create the mapping. It is important to highlight that this is a one-to-many mapping, which will introduce another challenge to this experiment. As a result, we need to modify the Document-based Ontology (DbO) construction method, which is explained in the below section.

6.2.2.3 DbO Construction Change

With reference to Figure 6-3 (an example with fake labels), four different gene concepts are identified from the text: Gene_A, Gene_B, Gene_C and Gene_D. The guiding ontology contains four ontology concepts: GO_1, GO_2, GO_3 and GO_4. Assuming that by going through the new mapping mechanism (UniProtKB) we have successfully identified that GO_1 mapped to both Gene_A and Gene_B, GO_2 mapped to both Gene_B and Gene_C, and GO_4 mapped to Gene_C.

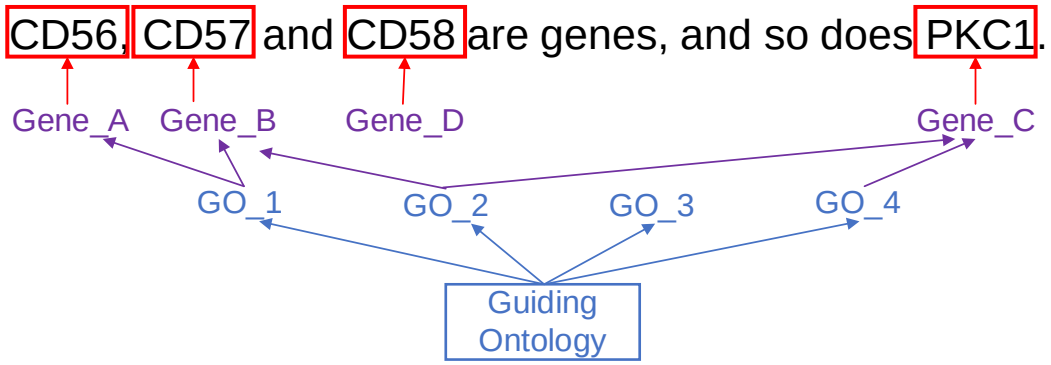


Figure 6-3 Example of the DbO construction change. Please refer to the related discussions for more information.

With the process implemented in the first experiment, the system would use GO_1, GO_2 and GO_4 to replace Gene_A, Gene_B and Gene_C respectively in the related DbO files. Then go through the word-replacement process (discussed in Section 4.2.2, refer to Figure 4-3, p. 65 for an example) to generate DSM models for GO_1 (in the original text, replace both CD56 and CD57 with the same invented unique string xoxovvGo_1vvoxox and use the modified text to generate the Word2Vec model), GO_2 (start over and replace both CD57 and PKC1 from the original text with the invented unique string xoxovvGo_2vvoxox, then use the modified text to generate the Word2Vec model) and GO_4 (start over again and replace PCK1 from the original text with the invented unique string xoxovvGo_4vvoxox and generate the associated Word2Vec model). Then, the system would use GO_1, GO_2 and GO_4 as the Mapped Subset to identify the best NN structure, as discussed previously in Section 4.3.3.2.

As pointed out already, most of (if not all) the mapping relations identified by the UniProtKB method were one-to-many relations, which means a big overlapping between concepts. In the above (Figure 6-3) example, since CD57 could belong to both GO_1 and GO_2, then there was a 50% overlapping between GO_1 and GO_2. Similarly, there would be a 50% overlapping between GO_2 and GO_4. The second experiment uses the same word-replacement process, and essentially, it is a dimensionality reduction process to project all the related word vectors on a single vector. Hence, if there is a large overlapping between two concepts, then the overlapped part is, in fact, noise that will pollute the related DSM models.

We can also explain this from a different aspect. From the first experiment, it is easy to understand that different concepts have different semantic distribution complexities (which is why we can make the related neural networks only overfit on non-informative concepts). In other words, different concepts have different semantic distribution patterns. The Neural Complex process discussed in Section 4.3.3, is essentially building a series of neural networks to identify this pattern. On the other hand, since a concept is a collection (or a combination) of related entities (words), these words should follow a similar pattern as their associated concept does. For any reason, if there is a big overlapping between two concepts, then the boundary or the pattern between them will be unclear, in which case we may not be able to get a close to 1 CS' result from the Neural Complex. In other words, a NN structure that can produce close to 1 CS' values for all the Mapped Subset concepts does not exist at all, because the concepts in the Mapped Subset have been polluted and become less informative or even non-informative.

To address this issue, the second experiment implemented a different approach to produce the DbO files, and subsequently, how to select the Mapped Subset to identify the best NN structure.

Instead of replacing Gene_XXX concepts with GO_XX concepts, the system will consider Gene_A as an equivalent class of GO_1; Gene_B as an equivalent of GO_1 and GO_2; and Gene_C as an equivalent of GO_2 and GO_4 as shown below. For now, please ignore the fact that in a formal OWL definition, the below statements will make GO_1 equals to GO_2 equals to GO_4 and

moreover Gene_A equals to Gene_B equals to Gene_C, since the purpose here is not to do the ontology reasoning.

```
DbO: Gene_A    a              owl:Class;
    rdfs:isDefinedBy           DbO:xxxxxxxxx
    rdfs:lable                       "Gene_A"
    owl:equivalentClass      DbO:GO_1

DbO: Gene_B    a              owl:Class;
    rdfs:isDefinedBy           DbO:xxxxxxxxx
    rdfs:lable                       "Gene_B"
    owl:equivalentClass      DbO:GO_1,
                             DbO:GO_2

DbO: Gene_C    a              owl:Class;
    rdfs:isDefinedBy           DbO:xxxxxxxxx
    rdfs:lable                       "Gene_C"
    owl:equivalentClass      DbO:GO_2,
                             DbO:GO_4

DbO: Gene_D    a              owl:Class;
    rdfs:isDefinedBy           DbO:xxxxxxxxx
    rdfs:lable                       "Gene_D"
```

When selecting the Mapped Subset (used to identify the best NN structure discussed in Section 4.3.3.2), the system will go through all the corpus concepts within the DbO Set and identify those who have an equivalent class in the guiding ontology, in this case, Gene_A, Gene_B and Gene_C, and use them to identify the best NN structure (as described below).

In this experiment, we have implemented and compared both approaches, and the results, which will be discussed in Section 6.3.1, clearly suggest that the new approach is much better than the old approach. The reason will be discussed towards the end of Section 6.3.1.2, but a brief explanation is provided below.

The key difference between those two approaches is that the former (old) builds the Mapped Subset (used to identify the best NN structure, discussed in Section

4.3.3.2) at the guiding ontology level. In other words, GO_1, GO_2 and GO_4. However, the latter builds the Mapped Subset based on those corpus concepts which have a valid mapping in the guiding ontology. In other words, Gene_A, Gene_B and Gene_C. The relationship between entity name and corpus concepts, e.g. CD56 belongs to Gene_A, is clearly or accurately extracted by PubTater. However, the mapping between corpus concept and the guiding ontology concept/class, e.g. Gene_B mapped to both GO_1 and GO_2, is handled by the UniProtKB approach discussed above, which is fuzzy (one-to-many). As a result, quite often there is a big overlapping between two guiding ontology concepts. For example, GO_1 mapped to both Gene_A and Gene_B, and GO_2 mapped to both Gene_B and Gene_C. Hence, there is a 50% overlapping between GO_1 and GO_2 because both of them have the entity name CD57 included, and the overlapped part could be considered as pollution. More discussions will be provided in Section 6.3.1.2.

6.2.3 Change of How to Identify the Best NN Structure

In the first experiment, there were only four ontology concepts. Hence, it was easy to identify the best result based on the chart without having a quantified method. However, the Mapped Subset in this experiment contains over 40 ontology concepts. Therefore, it is difficult to make the decision purely based on the chart. For example, Figure 6-4 and Figure 6-5 are the results from two different NN structures, and it is challenging to tell which one has a better (overall) result.

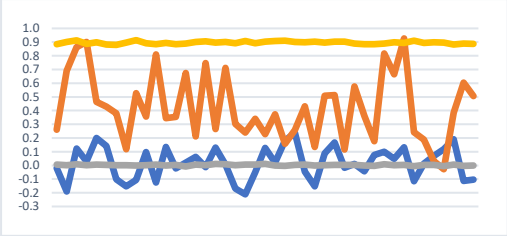


Figure 6-4 NN Structure Result Example 1

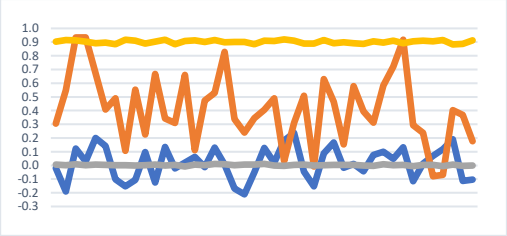


Figure 6-5 NN Structure Result Example 2

Hence, a new parameter called the Alignment Coefficient (AC) is introduced in the second experiment to quantify the results from various NN structures.

Essentially, Cosine Similarity (CS) and Aligned Cosine Similarity (CS') are the basic methods to measure the alignment. Moreover, each concept has its own alignment. The previous chapter has explained the meaning of different lines (blue, orange, yellow and grey) in the above plots. For example, the blue line represents $CS_{<ConceptName>}$ and means the original CS distribution, the orange line represents $CS'_{<ConceptName>}$ and stands for how good the alignment is (the performance on testing data), the grey line represents $\overline{CS}_{CW}$ (CW stands for Words in Common) and should be close to 0, and finally, the yellow line represents $\overline{CS}'_{CW}$ and stands for the performance of a neural network on the training dataset (the closer to 1, the better the neural network has been trained). Moreover, Section 5.3.1 introduced how to calculate $\overline{CS}_{CW}$ and $\overline{CS}'_{CW}$. The AC is, in fact, a combination of those four parameters and calculated as:

$$AC = \left(\frac{\sum_{i=0}^{n} CS'_i}{n} - \frac{\sum_{i=0}^{n} CS_i}{n}\right) \times \left(\overline{CS}'_{CW} - \overline{CS}_{CW}\right) \qquad 6\text{-}1$$

where n is the total number of the assessed concepts. This is similar to the cosine similarity, $AC \in [-1, 1]$. Basically, the blue line and grey line have been used as an offset to erase the potential randomisation a neural network may have. Essentially, the AC is the product of the "real" (after erasing the potential randomisation) performance (of a neural network structure) on the testing data (orange line) and the "real" performance on the training data (yellow line).

The closer to 1, the better the overall result is. Hence, the best NN structure is simply the structure that can generate the highest AC value.

6.2.4 Change of the Confidence Score Calculation

As Equation 4-10 (p. 104) indicated, the confidence score ($\overline{Conf}_{<ConceptName>}$) is needed to calculate both IC and CC value.

In the first experiment, an individual entity's confidence score was retrieved directly from the IBM-NLU service. Since a concept contains multiple entities (words), therefore its confidence score is simply the mean value of the associated entities. In which case, the confidence score indicates the probability of a concept being correctly identified.

However, in this new experiment, PubMed does not contain any information about confidence. Hence, unlike the first experiment, the confidence score is calculated by rerunning the whole Neural Complex to reproduce the work and then comparing the difference between the results with the formula below:

$$Conf_{<ConceptName>} = \begin{cases} 1 - |\Delta CS|, & |\Delta CS| \leq 1 \\ 0, & |\Delta CS| > 1 \end{cases} \quad 6\text{-}2$$

$$\Delta CS = CS'_{<ConceptName>} - CS''_{<ConceptName>}$$

where $CS''_{<ConceptName>}$ is the recalculated $CS'_{<ConceptName>}$ value.

In this case, the confidence score has a different meaning: how stable the neural network structure is at the individual concept level.

The other processes in this experiment are the same as the first experiment.

6.3 Results

This section will focus on the results generated from the second experiment. It will start with the various tests conducted to identify the best NN structure, then followed by the final IC result, final CC result and the final SI result.

6.3.1 Determine the Best NN Structure

This section will first discuss the result produced by the old approach, which was very poor, and then move to the new approach to demonstrate the considerable improvement.

6.3.1.1 Old Approach

The old approach refers to the approach used in the first experiment, where the system uses the guiding ontology concepts to replace the mapped corpus concepts and build the Mapped Subset based on the guiding ontology concepts.

With the old approach, 46 valid ontology concepts (denoted by GO IDs) were selected to build the Mapped Subset and computed 13 different NN structures in order to find the best one.

The first NN structure tested in this experiment contained 3 hidden layers and 2000 nodes on each layer. This structure was the best NN structure identified in the first experiment. However, as Figure 6-6 shows below, the result was very poor: the blue line ($CS_{<ConceptName>}$), grey line ($\overline{CS}_{CW}$) and the yellow line ($\overline{CS'}_{CW}$) align with the expectation, but the orange line ($CS'_{<ConceptName>}$) is nowhere near to 1 and, in fact, the highest value in there is only 0.5346, which brings down the AC value to 0.1137.

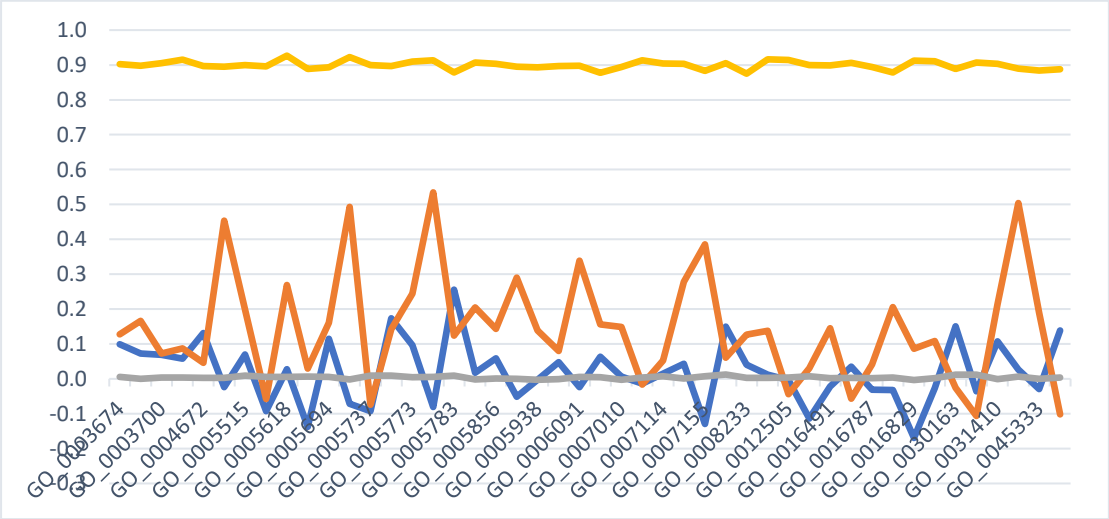


Figure 6-6 NN Structure Result – HL = 3 Nodes = 2000

The $\overline{CS'}_{CW}$ values are around 0.9, which indicates that the neural networks have been trained properly, while low $CS'_{<ConceptName>}$ values suggest that this structure is too complex even for informative classes, so it overfits every concept. To verify this conjecture, we kept the number of hidden layers unchanged but increased the number of nodes to 3000 to make the NN structure even more complex. If the analysis above is correct, then we should

receive an even worse AC result, in which case we should reduce the complexity to get a better AC result. Figure 6-7 confirms our suspicion, as the AC value dropped to 0.0638 as shown in Table 6-2.

We then conducted two tests and reduced the number of nodes to 1500 (test 2 in Table 6-2) and 500 (test 3, Table 6-2). For the 1500 test, it brought the AC value to 0.0969, which is higher than the 3000 test (test 1, Table 6-2), but still lower than the NN structure used in Figure 6-6. Moreover, further reducing the number to 500 only made it worse, as indicated by Figure 6-9 in which the yellow line ($\overline{CS'_{CW}}$) dropped significantly meaning the neural networks are not appropriately trained.

So, it seems that the complexity (of the NN structure) we are looking for is between the third experiment (test 3) in Table 6-2 and the original test shown in Figure 6-6. Other than modifying the number of nodes, changing the number of hidden layers is another strategy to adjust the complexity of a neural network.

Hence, test 4 keeps the number of nodes as 500 but increases the number of hidden layers from 3 to 5. It successfully brings up the AC value to 0.0951 (similar to the test 2 result). By adding another two hidden layers, test 5 brings up the AC value to 0.1084, which is very close to the original test shown in Figure 6-6. From test 3 to test 4, the AC value increased at a notable rate. However, by repeating the process and adding two more hidden layers (from test 4 to test 5), there is only a tiny increment. Hence, instead of keep adding new hidden layers, test 6 starts increasing the number of nodes, but the result generated from it was extremely poor, as shown in Figure 6-12.

Followed by test 6, test 7 restores the number of nodes to 500, but increases the number of hidden layers to 15 and the epoch number to 3000. However, the result was still poor and indicated that the neural network was not appropriately trained based on this structure.

So far, it seems that the original NN structure used in the first experiment is also the best NN structure in this experiment. However, this is not the case, since the results in Figure 6-6 indicate that this structure cannot be used to distinguish the informative and non-informative class because the orange line is nowhere near to 1, in fact, not a single value is close to 1. In theory, there are three possible explanations.

Firstly, it is simply because the structure used in Figure 6-6 is not the best structure, in other words, we have not found the best NN structure yet. Therefore we just need to test more structures until we identify the best one. This is unlikely to be the case based on the existing results – none of the tested structures (test 1 to test 7) manage to produce a single $CS'_{<ConceptName>}$ value that is close to 1.

As discussed earlier, this should be an overfitting situation, and the various tests discussed above confirmed this to a certain extent. However, this does not align with the common understanding of how the neural network works. Since this new experiment has a much larger scale than the first experiment, as shown in Table 6-1, intuitively, a more complex structure needs to be applied to the neural network(s) in order to train it properly. However, in practice, this is not the case, as the AC result from test 1 (which has a more complex structure) is worse than the result from Figure 6-6.

Hence, another (second) explanation is that the feature size for all the Word2Vec models used so far is 100 and worked well in the first experiment. However, since the second experiment is more intricate than the first one, a feature size of 100 may no longer be sufficient to capture the semantic complexity of the concepts during the word-replacement process. Hence the required information has not been included in the input of the neural network at all, which may explain why not a single $CS'_{<ConceptName>}$ value is close to 1 in all the tests conducted so far (test 1 to 7).

In order to eliminate this possibility, we have reproduced both source and target Distributional Semantic Model (DSM) Set and expanded the feature size from 100 to 200 and then conducted five more tests (tests 8 to 12) to verify the potential impact the feature size of the model could bring to the final AC result. Based on the results in Table 6-2, the increased feature size brings additional negative impacts on the final AC result. Hence, the feature size has nothing to do with the low $CS'_{<ConceptName>}$ values in various tests (test 1 to test 7) in this experiment.

The only explanation left (third) is what we have discussed in Section 6.2.2.3 – the overlapped entities between those ontology concepts should be considered as the noise, which will pollute the DSM models. Hence, the $CS'_{<ConceptName>}$ values would always be low because the concepts within the Mapped Subset can no longer be considered informative due to the noise and pollution. This is why a new approach, discussed in Section 6.2.2.3, has been proposed in this experiment.

The full result of the old approach is shown in Table 6-2 below, where HL = number of Hidden Layers, Nodes = number of Nodes, E = number of Epochs (default value is 350), W = feature size of the Word2Vec model (default value is 100).

This part of the work is the most time-consuming task in the whole SI algorithm. As explained before, the system needs to train $(1 + n \times 2)$ neural networks, where n is the number of assessed concepts. Since there are 46 valid ontology concepts identified in this approach, the system needs to train 93 neural networks for each individual test shown below. Including the test shows in Figure 6-6, there are 13 tests in total, so 1209 neural networks have been trained to assess this old approach.

ID	NN Structure	Alignment Coefficient
1	HL = 3, Nodes = 3000 (Figure 6-7)	0.06384
2	HL = 3, Nodes = 1500 (Figure 6-8)	0.09698
3	HL = 3, Nodes = 500 (Figure 6-9)	0.06774
4	HL = 5, Nodes = 500 (Figure 6-10)	0.09514
5	HL = 7, Nodes = 500 (Figure 6-11)	0.10842
6	HL = 7, Nodes = 2000 (Figure 6-12)	-0.01062
7	HL = 15, Nodes = 500, E = 3000 (Figure 6-13)	0.01675
8	HL = 3, Nodes = 500, W = 200 (Figure 6-14)	0.03004
9	HL = 3, Nodes = 1500, W = 200 (Figure 6-15)	0.06107
10	HL = 3, Nodes = 2000, W = 200 (Figure 6-16)	0.04397
11	HL = 3, Nodes = 3000, W = 200 (Figure 6-17)	0.06234
12	HL = 3, Nodes = 1000, W = 200, E = 1000(Figure 6-18)	0.04535

Table 6-2 Full result of the NN structure testing - Old Approach where HL = number of Hidden Layers, Nodes = number of Nodes, E = number of Epochs (default value is 350), W = feature size of the Word2Vec model (default value is 100)

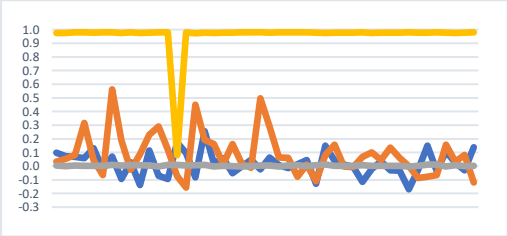


Figure 6-7 NN Structure Result - Test 1 HL = 3 Nodes = 3000

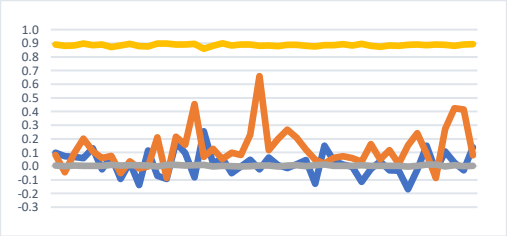


Figure 6-8 NN Structure Result - Test 2 HL = 3 Nodes = 1500

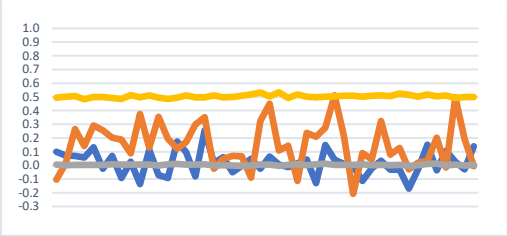


Figure 6-9 NN Structure Result - Test 3 HL = 3 Nodes = 500

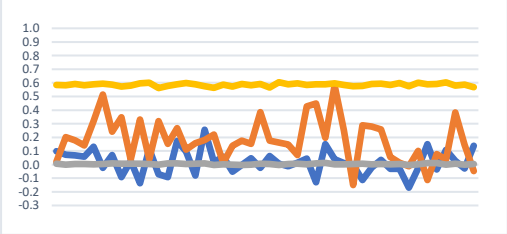


Figure 6-10 NN Structure Result - Test 4 HL = 5 Nodes = 500

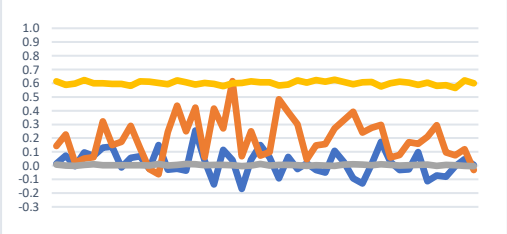


Figure 6-11 NN Structure Result - Test 5 HL = 7 Nodes = 500

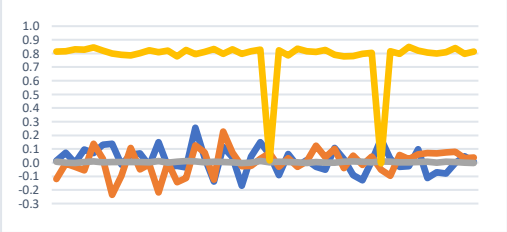


Figure 6-12 NN Structure Result - Test 6 HL = 7 Nodes = 2000

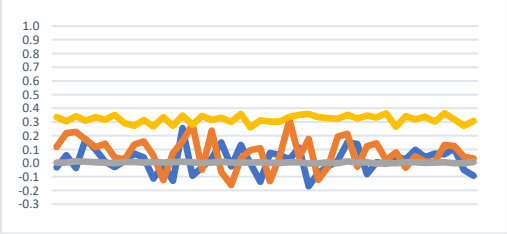


Figure 6-13 NN Structure Result - Test 7 HL = 15 Nodes = 500 Epochs = 3000

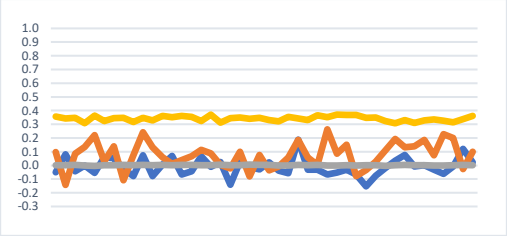


Figure 6-14 NN Structure Result - Test 8 HL = 3 Nodes = 500 Word2Vec Feature Size = 200

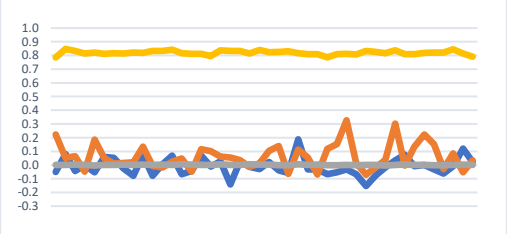


Figure 6-15 NN Structure Result - Test 9 HL = 3 Nodes = 1500 Word2Vec Feature Size =200

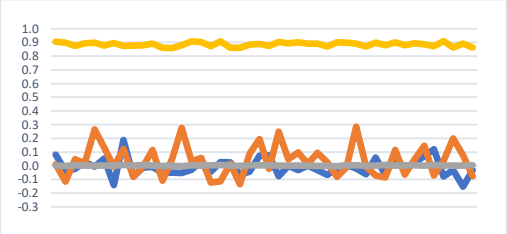


Figure 6-16 NN Structure Result - Test 10 HL = 3 Nodes = 2000 Word2Vec Feature Size = 200

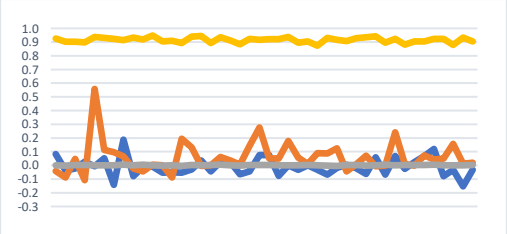


Figure 6-17 NN Structure Result - Test 11 HL = 3 Nodes = 3000 Word2Vec Feature Size =200

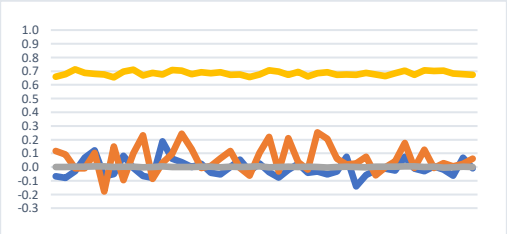


Figure 6-18 NN Structure Result - Test 12 HL = 3 Nodes = 1000 Word2Vec Feature Size = 200 Epochs = 1000

6.3.1.2 New Approach

Essentially, the old approach used the guiding ontology concepts to replace the mapped corpus concepts and used those ontology concepts as the Mapped Subset to identify the best NN structure. This is why the elements on the x-axis of Figure 6-6 are all denoted with the GO IDs (GO_XXXXX).

Instead of replacing the corpus concepts, the new approach adds an additional `owl:equivalentClass` statement in the DbO files to indicate if a corpus concept has a valid mapping in the guiding ontology. A Mapped Subset is then built, which is used to identify the best NN structure, by selecting those corpus concepts that have a valid mapping (in other words, have the `owl:equivalentClass` statement). With this method, 43 corpus concepts have been selected to construct the Mapped Subset.

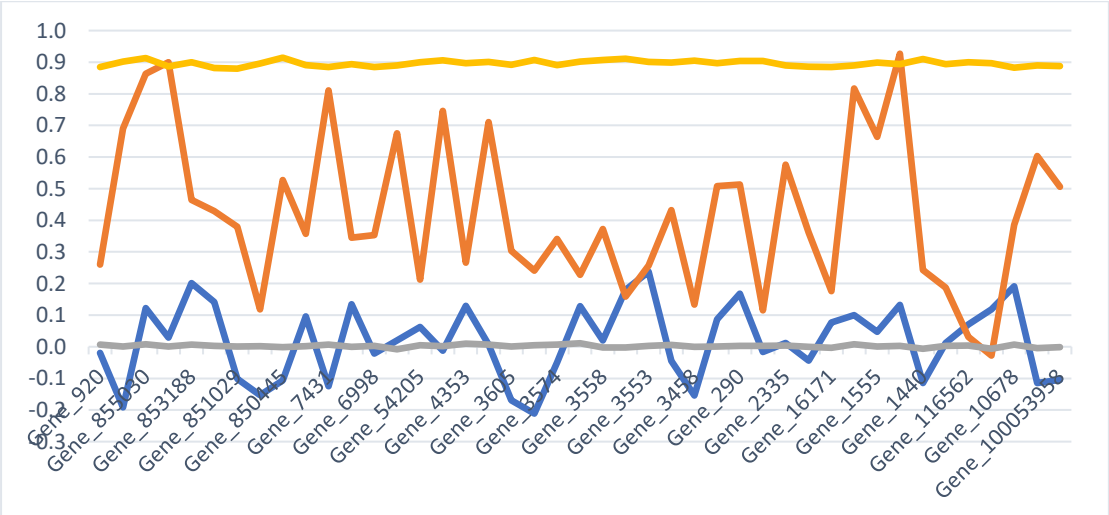


Figure 6-19 New Approach Result – HL = 3 Nodes = 1500 (Test 2)

As with the testing process used in the old approach, 10 different NN structures have been tested. Figure 6-19 is an example with 3 hidden layers and 1500

nodes on each layer. The full result is shown in Table 6-3 below, where HL = number of Hidden Layers, Nodes = number of Nodes.

ID	NN Structure	Alignment Coefficient
1	HL = 3, Nodes = 500 (Figure 6-20)	0.13882
2	HL = 3, Nodes = 1500 (Figure 6-19)	0.36149
3	HL = 3, Nodes = 2000 (Figure 6-21)	0.35701
4	HL = 3, Nodes = 3000 (Figure 6-22)	0.22075
5	HL = 5, Nodes = 500 (Figure 6-23)	0.16323
6	HL = 5, Nodes = 2000 (Figure 6-24)	0.30429
7	HL = 7, Nodes = 500 (Figure 6-25)	0.21052
8	HL = 7, Nodes = 2000 (Figure 6-26)	0.14827
9	HL = 6, Nodes = 2000, 1000, 1500, 500, 700, 200 (Figure 6-27)	0.23086
10	HL = 8, Nodes = 2000, 2000, 2000, 2000, 1500, 800, 200, 70 (Figure 6-28)	0.12673

Table 6-3 Full result of the NN structure testing - New Approach where HL = number of Hidden Layers, Nodes = number of Nodes, E = number of Epochs (default value is 350), W = feature size of the Word2Vec model (default value is 100)

As has been shown in the old approach, the increased feature size of the Word2Vec encoding does not lead to a higher AC value. Hence all the tests retain the Word2Vec models with 100 feature size, which have been generated based on the same configurations (`MinWordFrequency` = 1, `LayerSize` = 100, `WindowSize` = 5, `Iterations` = 100 and `Seed` = 42) as discussed in Section 5.2.2 to generate these Word2Vec models (DSM).

Based on the below result, 3 hidden layers with 1500 nodes on each layer is the best NN structure for this experiment.

Compared with the old approach, where the highest AC value was only 0.1137, this new approach manages to boost the AC value to 0.3615, which is more than 3 times higher than the old approach. As briefly discussed in this chapter already, this is because the new approach reduces the overlapping (of the associated words) between the concepts within the Mapped Subset. As a result, the AC value produced by the same NN structure is much higher in the new

approach than in the old approach. A detailed discussion will be provided in the next chapter.

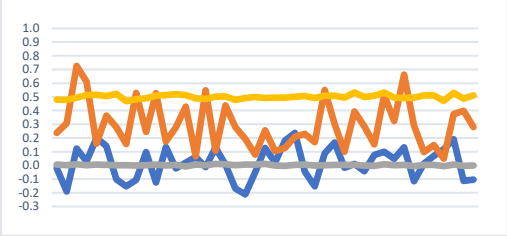


Figure 6-20 New Approach Result - Test 1 HL = 3 Nodes = 500

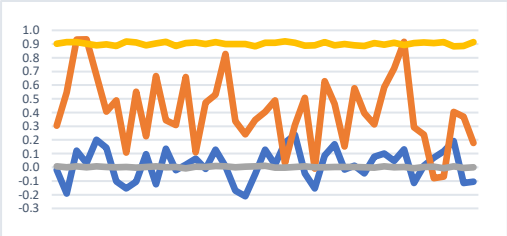


Figure 6-21 New Approach Result - Test 3 HL = 3 Nodes = 2000

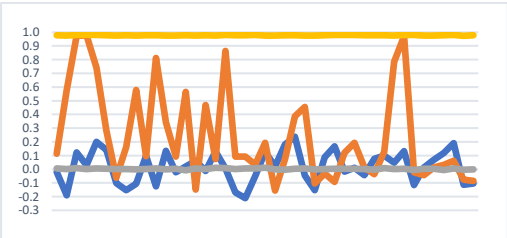


Figure 6-22 New Approach Result - Test 4 HL = 3 Nodes = 3000

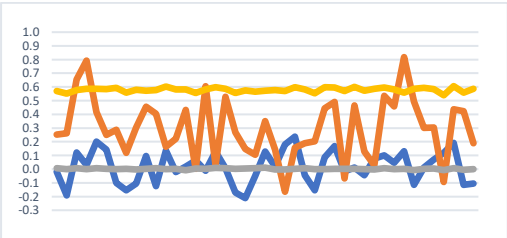


Figure 6-23 New Approach Result - Test 5 HL = 5 Nodes = 500

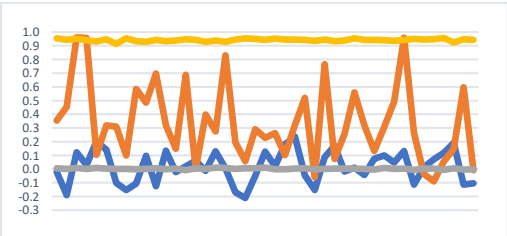


Figure 6-24 New Approach Result - Test 6 HL = 5 Nodes = 2000

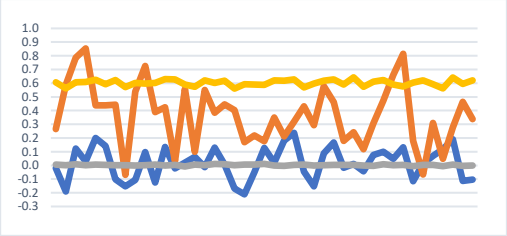


Figure 6-25 New Approach Result - Test 7 HL = 7 Nodes = 500

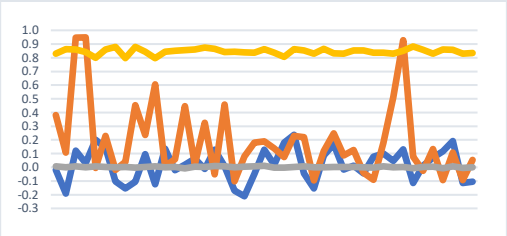


Figure 6-26 New Approach Result - Test 8 HL = 7 Nodes = 2000

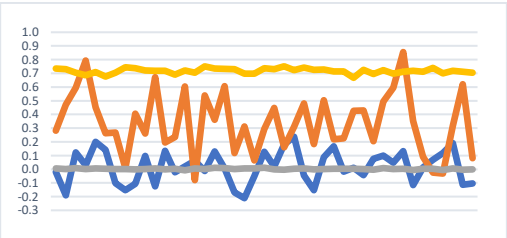


Figure 6-27 New Approach Result - Test 9 HL = 6 Nodes = 2000,1000,1500,500,700,200

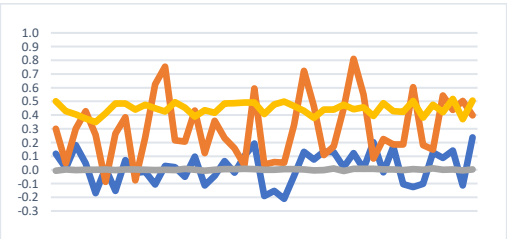


Figure 6-28 New Approach Result - Test 10 HL = 8 Nodes = 2000, 2000, 2000, 2000, 1500, 800, 200, 70

Unlike the result from the first experiment, where the system managed to generate a high $CS'_{<ConceptName>}$ value for all concepts in the Mapped Subset,

quite a few concepts in Figure 6-19 (which is the result of the best NN network) still have a low $CS'_{<ConceptName>}$ value (the orange line).

The root of this issue is the new mapping mechanism implemented in this experiment. As discussed in Section 6.2.2.2, protein has been used as an intermediary to build links between NCBI Gene IDs and GO IDs. Figure 6-1 and Figure 6-2 show an example of this mechanism to identify if the corpus concept Cblb has a valid mapping in the guiding ontology. According to the NCBI Gene database, Cblb has one protein accession – E3 ubiquitin-protein ligase CBL-B (Q3TTA7) (Figure 6-1). By searching this specific protein in the UniProtKB database, it has an association with (or is annotated with) 27 Gene ontology IDs (denoted by the GO IDs), as shown in Figure 6-2. The system then cross-compares those 27 IDs with the guiding ontology (also denoted by the GO IDs) to identify all the shared GO IDs and add them into the related DbO file as the equivalent classes. In other words, as long as there is a shared (means the GO ID exists in both the guiding ontology and the searching result from UniPortKB) GO ID, this specific corpus concept will be considered as having a valid mapping in the guiding ontology, and then be used as one of the concepts in the Mapped Subset to identify the best NN structure.

However, the 27 IDs identified from the UniProtKB means those 27 gene ontologies also have a protein association called Q3TTA7. In other words, they are possible mapping options. Just because there is a common (shared) ID between the 27 IDs and the guiding ontology does not mean that specific common (shared) concept is what Cblb should map to. With sufficient domain knowledge, it is probably possible to look at the context of the Cblb in the

original document to identify what is the best map. However, it is against the objective of the SI – to reduce the reliance on pre-defined knowledge and human intervention.

So, unlike the Mapped Subset in the first experiment, not all the concepts that have been included in the Mapped Subset in the second experiment can be claimed as informative. However, they are more likely to be informative compared with those concepts which have not been included in the Mapped Subset. Hence, the whole purpose of the Mapped Subset is to identify a group of concepts where are more likely to be informative. In other words, the second experiment focuses more on whether or not a specific concept has a valid mapping in the guiding ontology instead of figuring out what exactly it maps to. The whole idea of the Mapped Subset in this experiment is to provide a collection of concepts that are more likely to be informative.

It is why we have implemented the AC parameter in this experiment to assess the overall performance instead of making all individual concepts have a close to 1 $CS'_{<ConceptName>}$ value.

6.3.2 Final IC Result

By implementing the identified NN structure, the system will then be able to generate the CS' values for all the 330 corpus concepts. The full list is in Appendix III, but the top 20 results are shown below:

Concept	$\overline{CS}_{CW}$	$\overline{CS'}_{CW}$	CS	CS'
Disease_D010505	-0.0011829	0.8943114	0.0207074	0.9265831
Gene_853823	0.0006998	0.8891604	0.0293362	0.9255597
Disease_D058565	0.0013040	0.8942040	0.0951685	0.9201904
Gene_1509	0.0030898	0.8984119	0.1319365	0.9124130
Gene_855030	0.0077035	0.9156478	0.1217937	0.8874506
Disease_D003645	0.0037329	0.9009433	-0.1365934	0.8653799
Disease_D014689	0.0012572	0.8962939	-0.0907910	0.8605850
Disease_D007794	0.0059537	0.8974273	0.0340792	0.8475901
Disease_D001471	-0.0008832	0.8935587	0.0078127	0.8209251
Disease_D012480	0.0015221	0.8756122	0.0365848	0.7867964
Disease_C535390	-0.0016075	0.8954012	-0.0130027	0.7828508
Gene_7431	0.0067158	0.8877385	-0.1246973	0.7786208
Disease_D006192	0.0069363	0.9015389	0.1747824	0.7756270
Gene_1595	0.0079732	0.8874393	0.1003887	0.7480999
Gene_4155	0.0072611	0.9003241	0.0069378	0.7246535
Disease_C535342	0.0030218	0.8954569	0.1300061	0.7221987
Disease_D004405	0.0037021	0.9071112	0.0019479	0.7197927
Disease_D007640	0.0028883	0.9006598	-0.0841984	0.7059912
Disease_D012376	-0.0000348	0.9095467	0.1198208	0.6880114
Disease_D010211	0.0121139	0.8919884	0.2602795	0.6745116

Table 6-4 Top 20 CS' results

After that, the system reruns the whole process to reproduce this result with the same NN structure, and then use Equation 6-2 to calculate the confidence score. The full result is in Appendix IV, and the top 20 results are shown below:

Concept	Confidence Score
Gene_920	0.99917
Disease_D008581	0.99906
Disease_D003586	0.99898
Disease_D009877	0.99883
Disease_C567712	0.99865
Disease_D054198	0.99814
Gene_853823	0.99765
Disease_D005402	0.99725
Disease_D007410	0.99675
Gene_4155	0.99659
Disease_D014376	0.99641
Disease_D063646	0.99625
Disease_D015821	0.99581
Gene_3458	0.99540

Concept	Confidence Score
Disease_D010532	0.99522
Disease_D009369	0.99521
Disease_D059413	0.99517
Gene_1509	0.99500
Disease_D001261	0.99486
Disease_D006069	0.99446

Table 6-5 Top 20 confidence scores

The mean of those confidence scores is 0.91412929, which suggests the NN structure is stable enough to reproduce the work.

The IC result can be calculated by multiplying the confidence score with the CS' value as suggested by Equation 4-3. The full result is included in Appendix V, and the top 20 is shown below, where the name information is extracted from the Comparative Toxicogenomics Database (CTD).

ID	IC	Name
Disease_D010505	0.87718	Familial Mediterranean Fever
Disease_D003645	0.84699	Death, Sudden
Disease_D014689	0.84528	Venous Insufficiency
Disease_D058565	0.83814	Cerebral Ventriculitis
Disease_D007794	0.82778	Lameness, Animal
Disease_D001471	0.81087	Barrett Esophagus
Disease_D006192	0.74401	Haemophilus Infections
Disease_D012480	0.71332	Salmonella Infections
Disease_C535342	0.71208	Cataract, zonular
Disease_C535390	0.68640	Aspergillus niger infection
Disease_D007640	0.68139	Keratoconus
Disease_D004405	0.67615	Dysentery, Bacillary
Disease_D012376	0.65821	Rodent Diseases
Disease_D008228	0.64788	Lymphoma, Non-Hodgkin
Disease_D010211	0.64774	Papilledema
Disease_D009410	0.62986	Nerve Degeneration
Disease_D001284	0.62043	Atrophy
Disease_D007008	0.59461	Hypokalemia
Disease_D000230	0.57529	Adenocarcinoma
Disease_D009402	0.57172	Nephrosis, Lipoid

Table 6-6 Top 20 IC results

6.3.3 Final CC Result

330 corpus concepts have been assessed in this experiment, so, in total, there are 54285 concept pairs. Table 6-7 below gives the top 20 MIC results.[34]

var1	var2	mic
Gene_853823	Gene_855030	0.299823
Gene_16196	Gene_3574	0.285502
Disease_D001471	Disease_D009410	0.264536
Disease_D009402	Disease_C567712	0.261319
Disease_D011020	Disease_D016720	0.240265
Disease_D007239	Disease_D002177	0.221277
Disease_D010195	Disease_D010190	0.215015
Gene_6998	Disease_D001791	0.214139
Gene_54205	Gene_100053958	0.207533
Disease_D007710	Disease_D016715	0.205595
Disease_D007640	Disease_C535342	0.205554
Disease_D014245	Disease_D006069	0.202918
Gene_853188	Gene_853823	0.200852
Disease_C535590	Disease_D014860	0.196852
Disease_D015835	Disease_D006551	0.194558
Disease_D002177	Disease_D058365	0.191541
Disease_D014245	Disease_D002690	0.191407
Disease_D010505	Disease_D058565	0.190284
Disease_D007239	Disease_D009181	0.18902
Disease_D007239	Disease_D058365	0.188063

Table 6-7 Top 20 MIC results

The goal of this experiment is to assess the SI for various disease concepts identified from the corpus, hence the system will exclude the gene-related results from the final CC calculation. The full CC results are included in Appendix VI, and below are the top 20 results (sorted by CC value):

[34] Due to the large size, the full list will not be included in this thesis, but it is available to download from the following URL: https://edata.bham.ac.uk/640/

ID	CC	Name
Disease_D016638	22.39373	Critical Illness
Disease_D058365	21.14155	Candidiasis, Invasive
Disease_D058387	20.60300	Candidemia
Disease_D002177	20.59090	Candidiasis
Disease_D010195	20.54879	Pancreatitis
Disease_D015821	20.41633	Eye Infections, Fungal
Disease_D002277	20.26221	Carcinoma
Disease_D009877	20.24393	Endophthalmitis
Disease_D020096	19.75725	Zygomycosis
Disease_D018805	19.74200	Sepsis
Disease_D009181	19.73251	Mycoses
Disease_D018798	19.49380	Anemia, Iron-Deficiency
Disease_D003110	19.29635	Colonic Neoplasms
Disease_D019283	19.22553	Pancreatitis, Acute Necrotizing
Disease_D015473	19.20027	Leukemia, Promyelocytic, Acute
Disease_D003680	19.18037	Deglutition Disorders
Disease_D009196	19.02936	Myeloproliferative Disorders
Disease_D014008	19.00242	Tinea Pedis
Disease_D007239	18.97685	Infection
Disease_D008171	18.96094	Lung Diseases

Table 6-8 Top 20 CC results

One of the interesting findings here is that a few diseases that have a direct relation with Candida have been included in the above table. For example, Candidemia (3rd item in Table 6-8) which is the condition name when Candida is in people's bloodstream, and Candidiasis (4th item in Table 6-8) is just the name of the infection caused by Candida. Since the domain we use for this new experiment is about Candida and the way we build the corpora is by searching the keyword “candida” from PubMed. They are intuitively correct and almost self-approved to have strong connectivity with Candida.

6.3.4 Final SI Result

As with the first experiment, $\lambda_1 = \lambda_2 = 0.5$ is also used in this experiment. The final SI is calculated by Equation 4-10 (p. 104) with the normalised (from -1 to

1) IC (Equation 4-8) and CC (Equation 4-9) as discussed in Section 4.5, and the full results are available in Appendix VII. Below are the top 20 results. The Name and Categories information is again extracted from the CTD.

ID	Concept	Normalised IC	Normalised CC	SI	Name	Categories
1	Disease_D001471	0.87400	0.50255	0.68828	Barrett Esophagus	Cancer | Digestive system disease
2	Disease_D003645	0.94263	0.40586	0.67425	Death, Sudden	Pathology (process)
3	Disease_D014689	0.93940	0.27385	0.60663	Venous Insufficiency	Cardiovascular disease
4	Disease_D007794	0.90614	0.25606	0.58110	Lameness, Animal	Animal disease
5	Disease_D010505	1.00000	0.10657	0.55328	Familial Mediterranean Fever	Genetic disease (inborn)
6	Disease_D006192	0.74696	0.35065	0.54880	Haemophilus Infections	Bacterial infection or mycosis
7	Disease_C535342	0.68629	0.39507	0.54068	Cataract, zonular	Eye disease
8	Disease_D007640	0.62796	0.38128	0.50462	Keratoconus	Eye disease
9	Disease_D009410	0.53004	0.43318	0.48161	Nerve Degeneration	Pathology (process)
10	Disease_D058565	0.92583	0.00249	0.46416	Cerebral Ventriculitis	Nervous system disease
11	Disease_D010195	0.18153	0.74481	0.46317	Pancreatitis	Digestive system disease
12	Disease_D007565	0.40148	0.46632	0.43390	Jaundice	Pathology (process) | Signs and symptoms
13	Disease_D009196	0.30715	0.53464	0.42089	Myeloproliferative Disorders	Blood disease
14	Disease_D004405	0.61800	0.20505	0.41153	Dysentery, Bacillary	Bacterial infection or mycosis | Digestive system disease
15	Disease_D009190	0.30681	0.50778	0.40730	Myelodysplastic Syndromes	Blood disease
16	Disease_D012480	0.68863	0.12174	0.40519	Salmonella Infections	Bacterial infection or mycosis
17	Disease_D008223	0.41926	0.37026	0.39476	Lymphoma	Cancer | Immune system disease | Lymphatic disease
18	Disease_D010211	0.56401	0.21141	0.38771	Papilledema	Eye disease | Nervous

ID	Concept	Normalised IC	Normalised CC	SI	Name	Categories
						system disease
19	Disease_D008228	0.56429	0.20313	0.38371	Lymphoma, Non-Hodgkin	Cancer | Immune system disease | Lymphatic disease
20	Disease_D003680	0.20414	0.55553	0.37983	Deglutition Disorders	Digestive system disease | Ear-nose-throat disease

Table 6-9 Top 20 SI results

The full evaluation will be provided in the next chapter, together with a discussion about the positive findings. For example, initially all different disease types (e.g. Eye disease, Nervous system disease) were randomly distributed in the sample space, but start showing a strong grouping trend after applying the SI algorithm. Moreover, it identified that the most correlated concept to Disease_D003645 (Sudden Death) is Disease_D003643 (Death) without any pre-defined knowledge. Furthermore, a semantic analogy has been identified between Disease_D008223 (Lymphoma) and Disease_D008228 (Non-Hodgkin Lymphoma) due to a close SI between the two concepts.

Chapter 7 Evaluation and Discussion

The last chapter discussed the result of the second experiment together with an intuitive discussion. This chapter will focus on a more systematic way to evaluate the result.

The main challenge for the evaluation is that it is difficult to evaluate the SI result at the individual concept level (e.g. try to explain why Sudden Death can make more impacts than Venous Insufficiency) due to three reasons explained below.

Firstly, unlike the `Gene` concepts, which have an existing well-constructed ontology (the guiding ontology) that describes the Candida domain from the gene perspective, to compare with; an ontology that describes the Candida domain from the disease perspective does not exist. As a consequence, there is no gold standard for the Semantic Impact (SI) result (of those `Disease` concepts) to compare with.

Secondly, the SI result is purely based on the literature. Moreover, PubMed contains documents about both human beings and animals, which have completely different symptoms for specific diseases (this may well explain why lameness has such a high ranking in the final SI result). This makes it difficult to evaluate the result manually without reading all the corpus documents.

Thirdly, which links to the second reason: two thousand documents (1000 in each corpus) were used to conduct this experiment, but it may not be sufficient to cover all the knowledge included in the Candida domain. In fact, one of the

evaluation results discussed in Section 7.4 confirms this suspicion. Thus, the final SI result is, again, literature-based.

Since there is no gold standard to compare with, and it is challenging to evaluate the result manually, then an alternative evaluation strategy proposed in this thesis is to:

1. Firstly, divide SI into smaller components, in this case, Informative Coefficient (IC) and Connectivity Coefficient (CC), and try to demonstrate the output from these components are reasonable and correct.
2. Then look at the combined SI result at the macro-level and show that it has a strong clustering trend.
3. Thirdly, at the micro-level, try to demonstrate a specific clustering result with a high SI value (all involved concepts have a high SI value) is correct by demonstrating that SI works well on those concepts with high impact.
4. Fourthly, compare the SI approach with a different approach and demonstrate why it is a better solution. In particular, it is more reproducible.
5. Finally, identify an obviously non-significant concept (to the domain knowledge) and assess its SI result. If it has a low SI value, it indicates that SI also works well on those concepts with low impact.

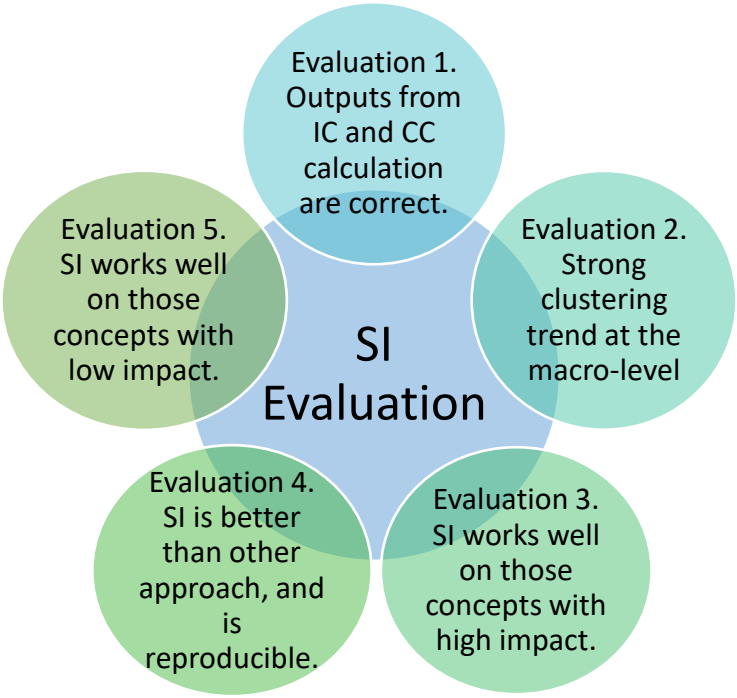


Figure 7-1 Overall Evaluation Plan. Five different aspects that we are going to evaluate.

The second item in the above list needs a further explanation. At the end of the SI algorithm, the system will produce a numerical value for each individual disease concept. It is easy to understand that at a deeper semantic level, the importance (and the impact they could bring to the domain knowledge) of related concepts (e.g. cancer-related disease) should be similar[35]. Hence, in theory, the final SI result will put various concepts into different groups (e.g. a group for cancer-related (disease) concepts and another group for infection-related (disease) concepts). In other words, if the SI algorithm works as expected, a clustering trend should be observed in the cancer-related (as well as infection-related) concepts. This is, in fact, why event-based concepts were grouped together in the first experiment, as discussed in Section 5.5.

[35] Related concepts should have a close/similar Semantic Impact (SI) value, but a close/similar SI value does not necessarily mean two concepts are related.

In practice, there are different types of cancer (e.g. immune system cancer and skin cancer), and it is reasonable to believe some of them are more important to the Candida domain knowledge than the others at a deeper semantic level. Hence, in the final SI result, cancer-related concepts will be divided into a few small groups instead of one group, but the clustering trend remains unchanged.

As a result, by showing all concepts are distributed randomly in the sample space before being processed by the SI algorithm but show a strong clustering trend (one that, moreover, looks intuitively reasonable) in the final SI result, it is then possible to indirectly demonstrate the correctness of SI.

This chapter is organised as follows: Section 7.1.1 and Section 7.1.2 focus on showing the results from the IC and CC calculations are reasonable and correct (item 1). Section 7.2 discusses how to quantitatively measure the grouping trend in the final SI result (item 2). After that, the semantic relationship identified between Lymphoma and Non-Hodgkin Lymphoma as an example is used to explain the correctness of the grouping result at the micro-level (item 3), then compare the SI approach with the Word2Vec method to demonstrate the advantage of the former (item 4). Section 7.4 focuses on the reproducibility of the result to demonstrate the SI approach is stable and reliable (item 4). Finally, Section 7.5 discusses the SI result for some of the stop words[36] (“is”, “that”, “a”) corresponding to item 5 in the above list, then followed by a summary in Section 7.6.

[36] A new concept will be manually created for those stop words without affecting the existing concepts. Section 7.5 will provide more details.

7.1 Evaluation 1 (Figure 7-1) - Informative Coefficient (IC) and Connectivity Coefficient (CC) Evaluation

7.1.1 Informative Coefficient Evaluation

In traditional computational linguistics study, the idea of Distributional Semantic Models (DSM) [9] is that the meaning of words can (at least to a certain extent) be inferred from their usage. By adopting and expanding DSM theory, one of the most important assumptions for the IC approach is that a high-dimensional vector can also be used to infer the semantic representation of a concept, which essentially is a set of words that belong to the same semantic group. Since different concepts have different representations, therefore, an individual concept should have its own unique distribution pattern. Moreover, the informativeness of a concept also has a reflection on its high-dimensional vector and will be captured by its distribution pattern. More specifically, the more informative a concept is, the more complex its distribution should be. The complexity can then be used to overcome the potential overfitting. This is why the overfitting mechanism (Section 4.3.3.3) can be implemented in this thesis to distinguish the informative concepts from the non-informative concepts.

In the previous chapters, we have intuitively discussed the correctness of the IC result (e.g. `Event` is the most informative concept in the news domain). However, what if the above assumption is not valid and the whole IC approach is just a random coincidence? In this section, we will quantitively demonstrate that the IC process is not random at all. In fact, (as a side effect) it has been demonstrated by comparing the results (of deciding the best NN structure)

produced by the old approach (Section 6.3.1.1) and the new approach (Section 6.3.1.2) discussed in the second experiment.

There were two different approaches implemented in the second experiment to construct the related Document-based Ontology (DbO) files and select the Mapped Subset used to identify the best NN structure. In summary, the mapped corpus concepts were replaced by their associated guiding ontology concepts in the old approach; and the Mapped Subset used to identify the best NN structure was built based on the guiding ontology concepts. However, in the new approach, the system only made an additional statement to link the corpus concepts with their potential mapping in the guiding ontology, and the Mapped Subset (for the best NN structure identification) was constructed at the corpus concepts level (Section 6.2.2.3).

It has been demonstrated that the new approach can generate a much better Alignment Coefficient (AC) result compared with the old approach due to the fact that those Mapped Subset concepts overlapped with each other in the old approach.

The t-SNE method has been implemented to plot the related keywords in the `W2V_Universal_Source` model on a 2-d picture to illustrate the overlapping scope. t-SNE stands for t-distributed Stochastic Neighbor Embedding, and Hinton et al. introduced it in [79] to visualise high-dimensional data.

t-SNE has been used in several places in this chapter to plot words in various Word2Vec models on a 2-D figure. There are two main reasons to use t-SNE instead of other dimensionality reduction methods, e.g. UMAP. Firstly, t-SNE

has been included in the DeepLearning4J framework, so it is an out-of-the-box solution. Secondly, those 2-D figures are just used to provide an intuitive feeling about the distribution of the words in the sample space.

In Figure 7-2, each node represents a word that was associated with a concept in the Mapped Subset (to identify the best NN structure) in the old approach, and uses a different colour to distinguish concepts (a larger version is available in Appendix VIII). Essentially, t-SNE will reduce the original Word2Vec dimension, or feature size, from 100 to 2 and correspond to the x-axes and y-axes in the below figures. As with the plot for the old approach, Figure 7-3 is an equivalent plot for the new approach (a larger version is available in Appendix IX). Figure 7-4 is a combined plot that contains both old and new approaches.

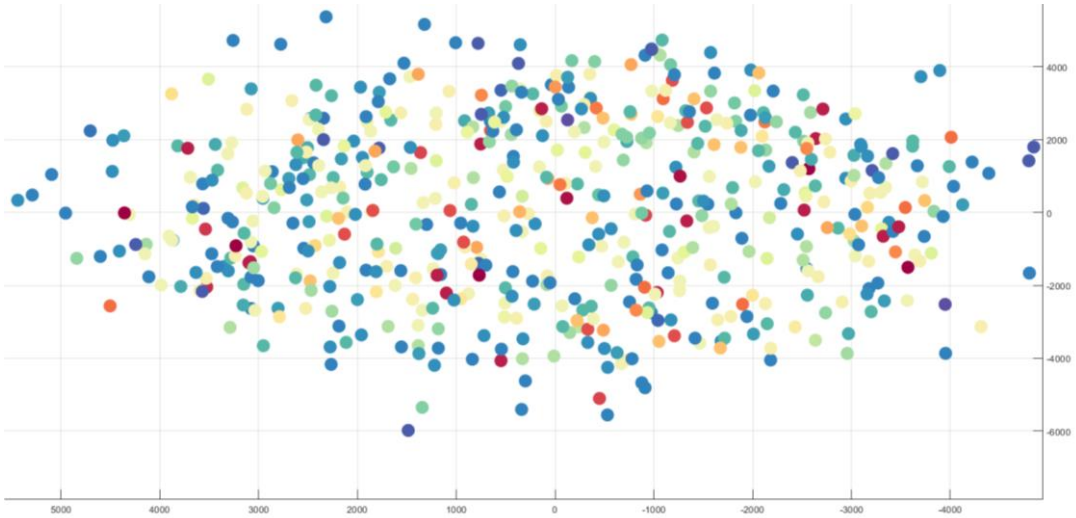


Figure 7-2 t-SNE plot for the old approach, where each node represents a word of a Mapped Subset concept and use a different colour to distinguish concepts

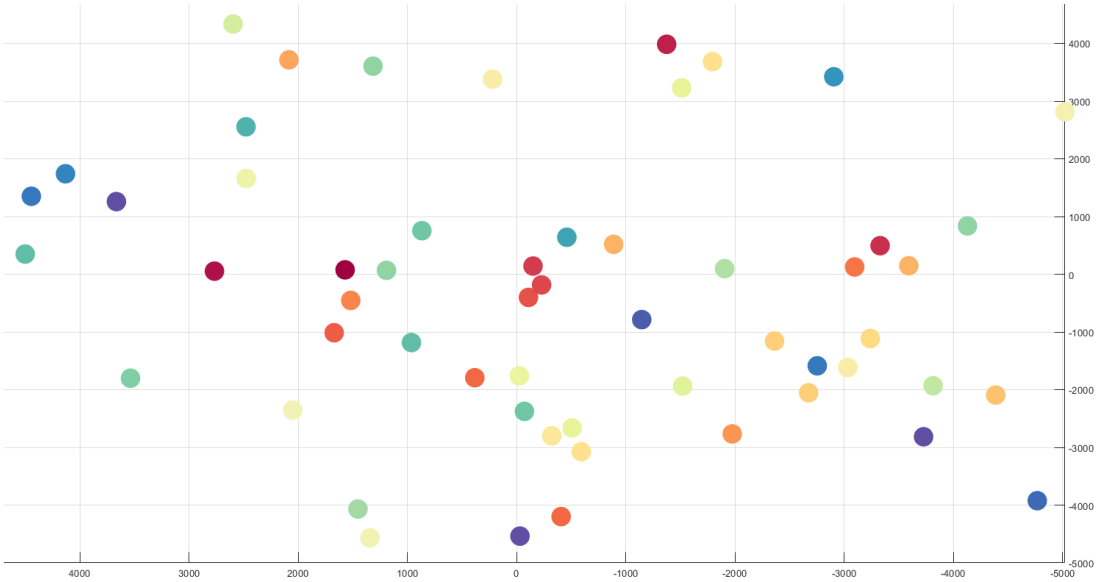


Figure 7-3 t-SNE plot for the new approach. Compared with Figure 7-2, nodes within this new approach are less overlapped.

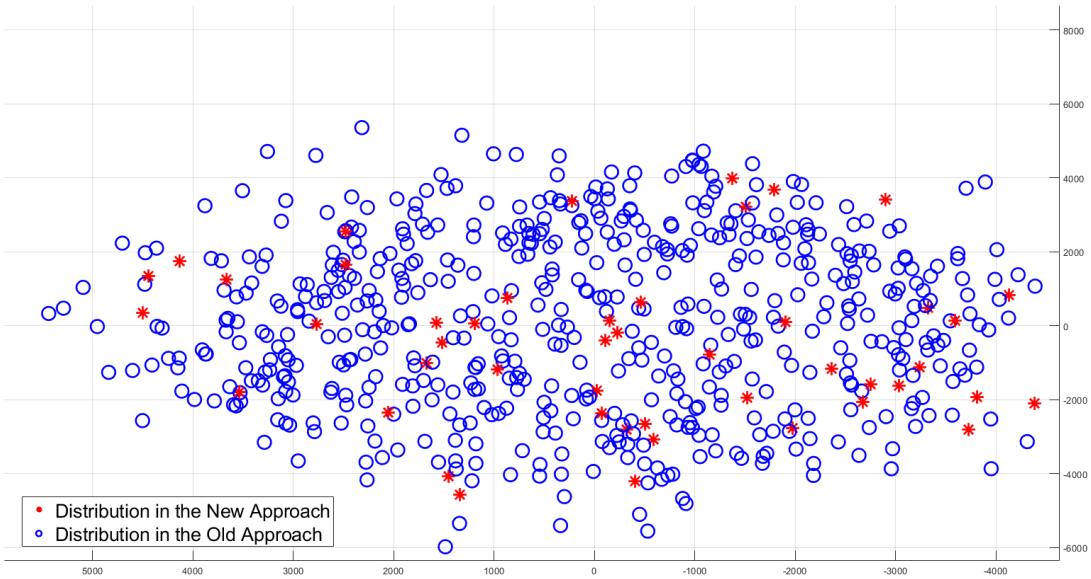


Figure 7-4 Combined t-SNE plot, blue cycles represent the old approach and red stars represent the new approach. It clearly demonstrates that the overall distribution of the old approach and the new approach are the same. In other words, blue cycles and red stars fall into the same area in the sample space.

Obviously, quite a lot of nodes in Figure 7-2 overlap and crowd together but are distributed separately in Figure 7-3. Part of the reason is that there are fewer nodes in the latter, but reduced keywords overlapping plays a more critical role here. Moreover, from Figure 7-4, it is easy to understand that the distribution of the concepts has been kept in the same area in the old and new approaches. Otherwise, the red stars (in Figure 7-4) would be placed in a different area.

If the assumption discussed at the beginning of this section[37] is not correct, this means concepts, including their associated keywords, do not have a unique distribution pattern (in other words, their informativeness has nothing to do with the overfitting behaviour), and the whole IC approach is just a coincidence. Then the overlapping shown in the above figures should not affect the performance of the NN structure, and if we compare the result of a NN structure in the new approach with the result generated by the same NN structure in the old approach, the outcome should be random. In other words, the new approach should not always have a better result.

However, the results shown in Table 7-1 suggest otherwise: in every tested NN structure, the new approach always generates a much bigger/better AC value (denoted as AC') than the old approach (denoted as AC) as shown in Figure 7-5 below.

Neural Network Structure (HL: hidden layer number Nodes: Number of nodes on each layer)	AC	AC'
HL=3, Nodes=500	0.06774	0.13882
HL=3, Nodes=1500	0.09698	0.36149
HL=3, Nodes=2000	0.11366	0.35701
HL=3, Nodes=3000	0.06384	0.22075
HL=5, Nodes=500	0.09514	0.16323
HL=7, Nodes=500	0.10842	0.21052
HL=7, Nodes=2000	-0.01061	0.14827

Table 7-1 Alignment Coefficient results in the old and new approach, where AC is the result from the old approach and AC' is the result from the new approach

[37] An individual concept should have its own unique distribution pattern. Moreover, the informativeness of a concept also has a reflection on its high-dimensional vector and will be captured by its distribution pattern.

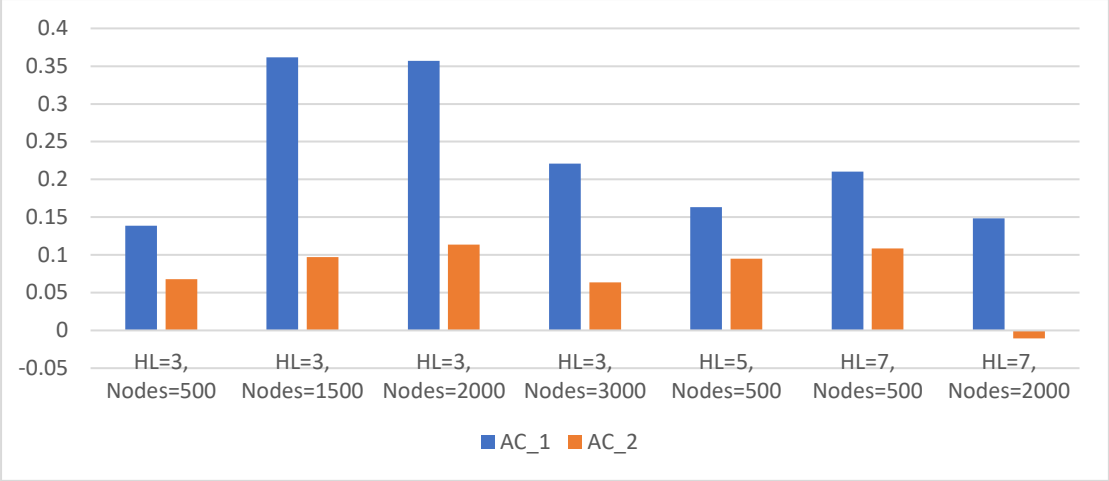


Figure 7-5 AC results in the old and new approach. It clearly demonstrates that the AC values are much higher in the new approach than the old approach.

In summary, the IC result is not a randomly generated coincidence, and the individual concept's semantic distribution pattern and its informativeness have a direct contribution to the final IC result. In other words, IC is a valid way to measure the informativeness of a concept.

7.1.2 Connectivity Coefficient Evaluation

The CC calculation is essentially a MIC based approach. Within this part, the contribution of this research is innovatively using the MIC algorithm in the NLP to identify the potential connectivity between concepts. Hence, proving the MIC algorithm itself is not part of this research. We evaluate the CC result by showing that it conforms to conceptual connections that are evidenced by various authoritative documents that were consulted.

Section 6.3.3 briefly mentioned that some of the diseases with "Candida" as the root word were marked with a high ranking. This is a promising result, since the domain is about the Candida, hence (intuitively) diseases that have a direct relation with Candida should have more robust connectivity with the other concepts in the domain. It is also useful to point out that the SI algorithm is not

a symbolic approach (i.e. it in no way uses the spelling of the names of diseases, genes, etc.), which means during the whole process, the system does not know and does not care that some of the concept names share the same root word.

However, those diseases have not been listed in Table 6-6 (p. 166, results of the IC), which suggests that they are not informative. This is, in fact, consistent with the findings from the first experiment.

In the first experiment, the keyword “Donald Trump” was used to search and collect news articles from the BBC News website. As a result, the “Person” concept, which belonged to the guiding ontology (BBC Core Concepts Ontology), had the lowest IC value compared with the other guiding ontology concepts. The reason behind this was because all the documents within the corpora were focused on one specific person. Consequently, the “Person” concept was not as generally applied as the others and would, of course, be less informative (relatively) than the others.

Using Disease_D058365 (Candidiasis, Invasive), Disease_D058387 (Candidemia) and Disease_D002177 (Candidiasis) as examples, the below table shows the difference between the IC and CC ranking in the second experiment.

Concept ID/Concept Name	IC Ranking	CC Ranking
Disease_D058365/Candidiasis, Invasive	132	2
Disease_D058387/Candidemia	112	3
Disease_D002177/Candidiasis	241	4

Table 7-2 Difference between IC and CC ranking

Indirectly, this also shows that informativeness and connectivity are two different aspects at a deeper semantic level and therefore need to be

addressed separately. In other words, there is a good reason for us to include both IC and CC value in the SI algorithm.

The above example can be considered as a piece of evidence showing that the CC process is an independent process (not affected by the IC process) that can produce a reasonable result. More evidence could be identified after analysing the rest of the results in Table 6-8 (p. 168, top 20 CC results).

For example, pancreatitis was ranked 5th by CC. According to [80], several studies have suggested a role of Candida in infected cases of several acute pancreatitis and urged a review of the place of antifungal therapy and prophylaxis. Moreover, this paper has not been included in our corpora.

Fungal Eye Infections is the 6th in Table 6-8. Since Candida belongs to fungal, then this is, again, self-evidenced.

Carcinoma itself, which was ranked as number 7 in Table 6-8, does not have an obvious connection with Candida, but cancer does. According to the Centers for Disease Control and Prevention (CDC) [81], chemotherapy and radiation could weaken the immune system while killing the cancer cells, which can increase the chances of getting an infection, including fungal infection.

Endophthalmitis is the next on the table. In fact, there is a terminology called “endogenous candida/fungal endophthalmitis”, which clearly suggests the connection between them.

According to CDC, zygomycosis (9th in the table) is a serious fungal infection caused by a group of moulds called mucormycetes. Even if it is not caused by

Candida directly, both Candida and mucormycetes belong to the fungal, so it is reasonable to believe zygomycosis could have a strong connection with the other concepts within the Candida domain.

Quite a few studies have suggested that Candida can cause sepsis which was ranked at number 10 in the table. For example, in [82], it indicates that "*In addition to bacteria, fungi—mainly Candida albicans and other Candida sp.—can cause sepsis and this entity has increased over the last decades.*". Again, this specific paper has not been included in the corpora.

In summary, most of the diseases in Table 6-8 can be shown by means of independent, authoritative evidence from CDC etc., to have strong connectivity with the Candida domain. Therefore, the $\mathcal{CC}$ approach can produce a reasonable result. However, people may argue that since the corpora are all about Candida, so the diseases mentioned in the corpora would be somehow related to Candida anyway. In order to erase this concern, another analysis has been conducted to examine the most correlated concepts with Disease_D003645 (`Sudden Death`). The reason to examine this particular concept is because a) `Sudden Death` has a very high ranking in the final SI result (the 2nd), and b) It is relevantly easier to find out if a disease is deadly or not. The top 10 results are shown in below table (Name and Categories information are extracted from the CTD[38]):

[38] Comparative Toxicogenomics Database. http://ctdbase.org

ID	Name	Categories
Disease_D003643	Death	Pathology (process)
Disease_C565469	Immune Deficiency Disease	Immune system disease
Disease_D014456	Ulcer	Pathology (process)
Disease_D014376	Tuberculosis	Bacterial infection or mycosis
Disease_D007794	Lameness, Animal	Animal disease
Disease_D008223	Lymphoma	Cancer | Immune system disease | Lymphatic disease
Disease_D007246	Infertility	Urogenital disease (female) | Urogenital disease (male)
Disease_C566367	Light Fixation Seizure Syndrome	Congenital abnormality | Eye disease | Genetic disease (inborn) | Mental disorder | Nervous system disease | Signs and symptoms
Disease_D006402	Hematologic Diseases	Blood disease
Disease_D016720	Pneumocystis Infections	Bacterial infection or mycosis

Table 7-3 Top 10 the most correlated concepts (based on the MIC value) to "Sudden Death".

One of the most promising findings here is that the concept of `Death` has been identified as the most correlated concept with `Sudden Death`. As a human being, this is obvious from the names themselves. However, this is quite a big achievement from the system's perspective -- it makes no use of the spellings of these two concept labels, nor any predefined information about them, it can now identify the connection between these two concepts.

The next in Table 7-3 is `Immune Deficiency Disease`, and it is common sense that people will die with a poor or dysfunctional immune system.

The rest of the table includes:

- Ulcer. The term ulcer means a sore that does not heal quickly, and it can occur almost anywhere on the body. Normally, it does not cause sudden death for sure, so it is not entirely clear why this concept exists here. However, it is not entirely wrong either, as a severely bleeding ulcer can cause a rapid loss of blood and possibly death if left untreated.

- Tuberculosis. It used to be one of the most deadly diseases before the vaccine exists. It still has a high death rate in some of the third countries. According to WHO [83], about 10 million new tuberculosis cases been discovered worldwide in 2017.
- Animal Lameness. As mentioned above, quite a few of the documents in the corpora are about animals instead of the human being. There are no official statistics showing how many animals die because of lameness. However, common sense suggests that in the wild world, if animals have difficulty running normally, then they may well be killed by predators.
- Lymphoma, a type of cancer, there is no doubt that it can cause death, although the survival rate after the treatment is still very high.
- Infertility. From an individual human being's perspective, infertility will not cause death. However, the interesting thing to point out is that as a species, infertility means extinction – another version of death.
- Light Fixation Seizure Syndrome. Since the corpus includes animals, it is reasonable to believe it can cause death indirectly (e.g. insects run/fly into fire).
- Hematologic Diseases. It is quite a broad category and can be associated with the human immunodeficiency virus and AIDS. Therefore, it is reasonable to believe it can cause death.
- Pneumocystis Infections could be pneumocystis pneumonia (PCP). It can cause a lung infection in people with a weak immune system. It is especially seen in people with cancer undergoing chemotherapy, HIV/AIDS cases, and the use of medications that suppress the immune

system. In short, PCP itself will not cause death, but it is quite often a complication of a more deadly disease.

In summary, most of the diseases in Table 7-3 could be considered reasonable to correlate with `Sudden Death` strongly (to a certain extent). This suggests that the CC process can be used to identify the connection between different concept pairs, directly or indirectly related to the domain itself.

7.2 Evaluation 2 (Figure 7-1) - Clustering Trend

This part of the evaluation focuses on the macro-level and tries to provide evidence for a strong grouping/clustering trend in the final SI result.

Using the Comparative Toxicogenomics Database (CTD), it is possible to retrieve the category information of the `Disease` concepts (e.g. the Categories column in Table 6-9, p. 170). There were 285 `Disease` concepts processed in the second experiment, which could be split into 33 CTD categories. Using cancer, which is one of the categories identified by CTD, as an example, 28 concepts have been put into this category as listed below.

ID	Name	Categories
Disease_D001471	Barrett Esophagus	Cancer | Digestive system disease
Disease_D008223	Lymphoma	Cancer | Immune system disease | Lymphatic disease
Disease_D008228	Lymphoma, Non-Hodgkin	Cancer | Immune system disease | Lymphatic disease
Disease_D054198	Precursor Cell Lymphoblastic Leukemia-Lymphoma	Cancer | Immune system disease | Lymphatic disease
Disease_D000230	Adenocarcinoma	Cancer
Disease_D019337	Hematologic Neoplasms	Blood disease | Cancer
Disease_D006258	Head and Neck Neoplasms	Cancer
Disease_D015470	Leukemia, Myeloid, Acute	Cancer
Disease_D015179	Colorectal Neoplasms	Cancer | Digestive system disease
Disease_D016715	Proteus Syndrome	Cancer | Congenital abnormality | Musculoskeletal disease

ID	Name	Categories
Disease_D015473	Leukemia, Promyelocytic, Acute	Cancer
Disease_D009062	Mouth Neoplasms	Cancer | Mouth disease
Disease_D010212	Papilloma	Cancer
Disease_D009959	Oropharyngeal Neoplasms	Cancer | Ear-nose-throat disease | Mouth disease
Disease_D003110	Colonic Neoplasms	Cancer | Digestive system disease
Disease_D034721	Mastocytosis, Systemic	Cancer | Immune system disease
Disease_D002277	Carcinoma	Cancer
Disease_D063646	Carcinogenesis	Cancer | Pathology (process)
Disease_D016399	Lymphoma, T-Cell	Cancer | Immune system disease | Lymphatic disease
Disease_D009362	Neoplasm Metastasis	Cancer | Pathology (process)
Disease_D010190	Pancreatic Neoplasms	Cancer | Digestive system disease | Endocrine system disease
Disease_D001943	Breast Neoplasms	Cancer | Skin disease
Disease_D009369	Neoplasms	Cancer
Disease_D002294	Carcinoma, Squamous Cell	Cancer
Disease_D006223	Hamartoma Syndrome, Multiple	Cancer | Genetic disease (inborn)
Disease_D018307	Neoplasms, Squamous Cell	Cancer
Disease_D008175	Lung Neoplasms	Cancer | Respiratory tract disease
Disease_D007938	Leukemia	Cancer

Table 7-4 Diseases that belong to the cancer category

Each `Disease` concept in the above table contains one or more entity names (keywords) as extracted by PubTator through the semantic information extraction process discussed in Section 4.2.1. Hence, it is easy to generate a list of entity names (keywords) that are associated with the Cancer category, denoted as $W_{Cancer} = \{w_1, w_2, \dots, w_n\}$, where w_n is the nth associated word. In total, there are 96 keywords included in the W_{Cancer}, and the full list is included in Appendix X.

As part of the word-replacement process, the system produced a universal Word2Vec model based on the original text in the source corpus without replacing any words (`W2V_Universal_Source`). In other words, it has the most complete vocabulary list. Using t-SNE (as mentioned in Section 7.1.1), it

is easy to reduce the dimension of the `W2V_Universal_Source` model from 100 to 2 to generate a new model denoted as `W2V_Universal_Reduced`.

Then, it is easy to get the vectors of W_{Cancer} from the `W2V_Universal_Reduce` model, which is denoted as $V_{Cancer} = \{\vec{v}_1, \vec{v}_2, \dots, \vec{v}_n\}$, where $\vec{v}_n$ is the vector of w_n in the `W2V_Universal_Reduced` model.

Following the same process, we can generate a V_{Others} which contains vectors (in the `W2V_Universal_Reduced` model) for all the entity names (keywords) that have been associated with the other 32 CTD categories. Then plot V_{Cancer} and V_{Others} to the same chart to generate Figure 7-6 (where red stars are W_{Cancer} and blue dots are W_{Others}) to indicate how W_{Cancer} is distributed in the sample space.

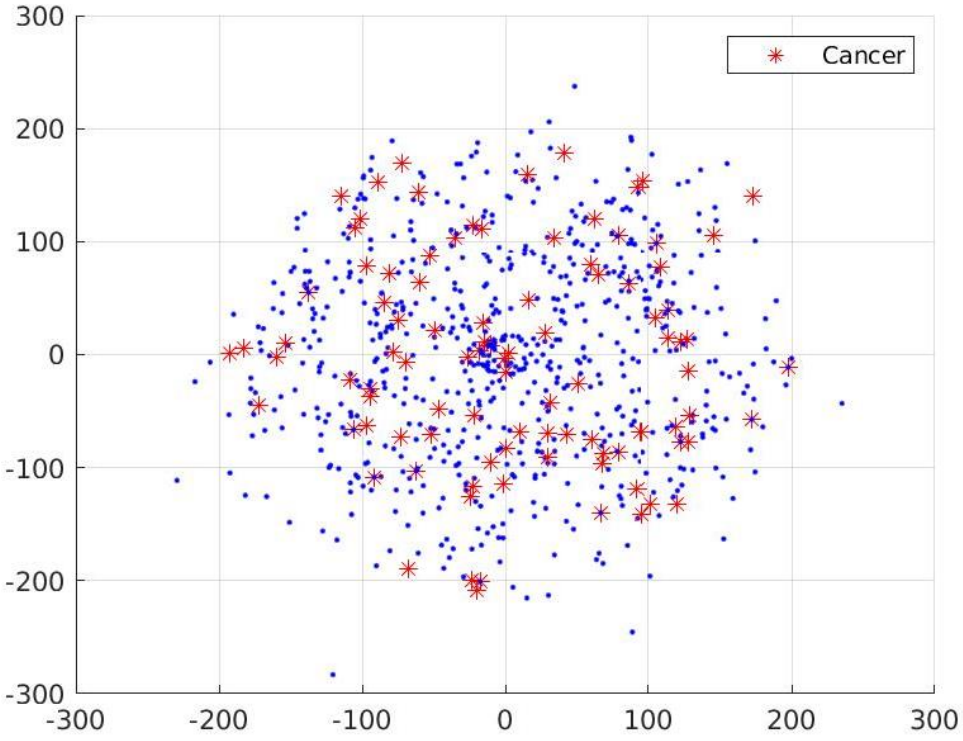


Figure 7-6 Keywords distribution for cancer, where the red stars are cancer-related keywords, and the blue dots are keywords for the other categories. Intuitively, those red stars are randomly distributed in the sample space.

Intuitively, there is no clustering/grouping trend for the red stars shown in the above figure. This suggests that the keywords for cancer were randomly distributed in the sample space before being processed by SI.

In order to quantify this result, the Hopkins Statistic [84], a popular statistic method to measure the clustering tendency of a data set, has been implemented here.

Let D be a real dataset (the red stars in Figure 7-6 example), and its Hopkins value can be calculated as [85]:

1. Sample uniformly n points D.
2. For each point $p_i \in D$, find its nearest neighbour p_j; then compute the Euclidean distance between p_i and p_j and denote it as $X_i = dist(p_i, p_j) = \sqrt{\sum_{a=1}^{n}(p_{i_a} - p_{j_a})^2}$ where p_{i_a}, p_{j_a} is the a[th] element in the vector that represents the coordinate of the two points, n is the dimension of the vector.
3. Generate a simulated data set $(random_D)$ drawn from a random uniform distribution with n points $(q_1, \dots, q_n)$ and the same variation as the original real dataset D.
4. Compute the distance, Y_i from each artificial point $(random_D)$ to the nearest real data point. For each point $q_i \in random_D$, find its nearest neighbour q_j in D, then calculate the Euclidean distance between q_i and q_j, which is denoted as $Y_i = dist(q_i, q_j)$.
5. The Hopkins value (H) is then calculated by the below equation:

$$H = \frac{\sum_{i=1}^{n} Y_i}{\sum_{i=1}^{n} Y_i + \sum_{i=1}^{n} X_i} \qquad 7\text{-}1$$

where n is the number of points.

If the value is between [0, 0.3], then the data is regularly spaced. If the value is around 0.5, then it is randomly distributed. If the value is between [0.7, 1], then the data has a high tendency to cluster.

By applying the Hopkins algorithm, we can then calculate the H value (calculated by using the vectors in the original `W2V_Universal_Source` model) for each category before going through the SI approach, and the H' which is the equivalent value (calculated by using the final SI score) after going through the SI approach. The result shows below (sorted by H'):

Category	Before SI H	After SI H'
Skin disease	0.596437801	0.881599949
Nervous system disease	0.597710657	0.879714703
Blood disease	0.579276024	0.875272126
Endocrine system disease	0.598739246	0.874695793
Urogenital disease (male)	0.597841732	0.873183855
Eye disease	0.598608094	0.872215268
Bacterial infection or mycosis	0.598889669	0.868856470
Cardiovascular disease	0.598110126	0.868585313
Infant-newborn disease	0.596735507	0.865359722
Urogenital disease (female)	0.596771443	0.863862672
Genetic disease (inborn)	0.596645812	0.862379303
Lymphatic disease	0.598490714	0.860171082
Immune system disease	0.607350585	0.856510690
Connective tissue disease	0.597706811	0.856379037
Wounds and injuries	0.596781703	0.856110720
Fetal disease	0.597362568	0.854759383
Cancer	0.597397149	0.852696218
Ear-nose-throat disease	0.597078938	0.852263828
Mouth disease	0.605219056	0.847849818

Category	Before SI H	After SI H'
Pathology (process)	0.596301881	0.842915861
Viral disease	0.595515463	0.840475952
Metabolic disease	0.594727810	0.839120430
Nutrition disorder	0.595822336	0.838282195
Animal disease	0.595533169	0.838200262
Parasitic disease	0.595710173	0.837915677
Pathology (anatomical condition)	0.595999450	0.837000528
Signs and symptoms	0.594394809	0.836566795
Congenital abnormality	0.594278874	0.834016864
Mental disorder	0.594665790	0.833866830
Respiratory tract disease	0.593822376	0.830800807
Digestive system disease	0.601336763	0.827631886
Pregnancy complication	0.603897542	0.820621276
Musculoskeletal disease	0.605564208	0.817757795

Table 7-5 Hopkins result before and after SI processing

Based on the above Hopkins values, it is easy to see that keywords for all the categories are randomly distributed before the SI approach. However, after processing by the SI algorithm, they all showed a strong clustering trend ($H' > 0.8$). In other words, the SI algorithm can produce a result that aligns with the expectation.

7.3 Evaluation 3 and 4 (Figure 7-1) - Lymphoma and Non-Hodgkin Lymphoma

This part of the evaluation focuses on two areas:

a) At the micro-level, demonstrate a specific clustering result with a high SI value (all involved concepts have a high SI value) is correct.
b) Try to produce a similar result with a different method/algorithm, in this case, Word2Vec and explain why SI is a better approach.

The reason to select Word2Vec as a comparison is because it is one of the most popular and well-established word embedding methods which also has a "clustering/grouping" effect. In addition, it is also a predictive-based measurement that has been used for concept selection, as discussed in Section 2.4. Using the below figure as an example [86], originally, it has been used to explain how Word2Vec understands semantic relationships, like Moscow and Russia are related the same way Beijing and China are (capital and country), and not in the same way Lisbon and Japan are. Consequently, those keywords/entity names that represent the concept of "Countries" have been grouped together and separate from those keywords/entity names that represent the "Capital" concept. In this case, there should be a high Cosine Similarity (CS) value between the keywords/entity names of the same concept. For example, in the pre-trained GoogleNews Word2Vec model [39], which contains 300-dimensional vectors for 3 million words and phrases, the CS between Moscow and Beijing is 0.5461714.

[39] "Pre-trained word and phrase vectors", available at the following URL: https://code.google.com/archive/p/word2vec/

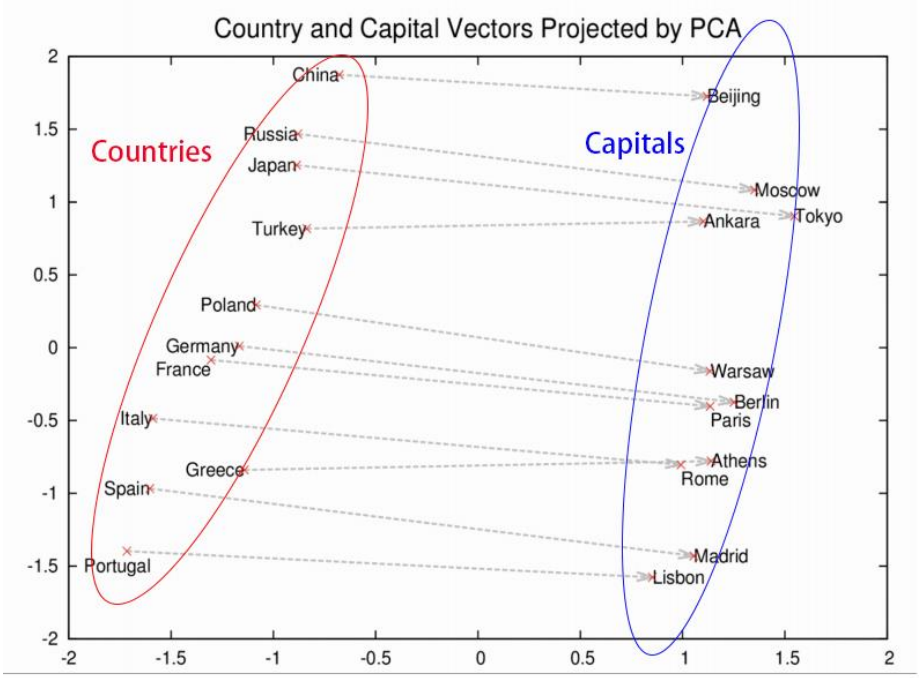


Figure 7-7 Word2Vec grouping effect [86]

Disease_D008223 (Lymphoma) and Disease_D008228 (Non-Hodgkin Lymphoma) are selected here for two reasons: 1) In the final result, both of them have a high SI value, and more importantly, they are close to each other (ranked 17th and 19th). 2) They only contain a limited number of entity names (keywords), so that it is easy to compare. For example, Lymphoma contains three keywords: *malignant lymphomas*, *malignant lymphoma* and *lymphoid leucosis*, and Non-Hodgkin Lymphoma only have one keyword, and it is an abbreviation: *NHL*.

With such limited information, the SI algorithm still manages to group them together (ranked 17th and 19th). In order to quantify how close they are, we use the below formula to calculate their closeness:

$$C = (1 - |CS'_L - CS'_N|) \times \frac{\overline{CS}'_{CW-L} + \overline{CS}'_{CW-N}}{2} \qquad 7\text{-}2$$

where $C \in [-1,1]$, CS'_L is the after alignment cosine similarity for the concept Lymphoma (the orange line) and $\overline{CS}'_{CW-L}$ is the average value of the after alignment cosine similarity for the overlapped words (the yellow line) for the same concept. On the other hand, CS'_N and $\overline{CS}'_{CW-N}$ are the equivalent values for the Non-Hodgkin Lymphoma concept. As C has the same range as the Cosine Similarity, we can then use it to compare with the result from the Word2Vec model.

Based on the above method, the C value for Lymphoma and Non-Hodgkin Lymphoma is 0.84579.

Within the `W2V_Universal_Source` model, we can easily calculate the cosine similarity between the related keywords. For example, calculate the CS value between *malignant_lymphomas*, which is one of the keywords for Lymphoma, and *NHL*, which is the keyword for Non-Hodgkin Lymphoma. The idea here is to calculate and compare all the keyword pairs between two concepts and see how close they are. In order to make this comparison more comprehensive, a few more keyword pairs are manually added (also compare keywords within the same concept), and the complete list of the class pairs together with the results is shown in Table 7-6 below:

Class Pair	Cosine Similarity
lymphoma => nhl	-0.12602
malignant_lymphoma => nhl	0.32301
malignant_lymphomas => nhl	0.23673
lymphoid_leucosis => nhl	0.30389
lymphoid => nhl	-0.12746
malignant_lymphoma => malignant_lymphomas	0.57829

Table 7-6 CS value of the related keyword pairs

Within the SI approach, Lymphoma and Non-Hodgkin Lymphoma are close to each other. Suppose a traditional word embedding method like Word2Vec

could successfully identify the same semantic relationship. In that case, all the related keywords between these two concepts should have a high cosine similarity (like the Moscow and Beijing example used previously). However, the above table suggests otherwise. In fact, the only keyword pair that has a high CS value is between *malignant_lymphoma* and *malignant_lymphomas*, but both of them belong to the Non-Hodgkin Lymphoma concept (it only suggests malignant_lymphoma and malignant_lymphomas might belong to the same concept, without identifying the semantic relationship between Lymphoma concept and Non-Hodgkin Lymphoma concept).

It is suggested that the SI approach could generate a much better result within the given corpus compared with Word2Vec. However, it does not mean the Word2Vec cannot be used to identify the semantic relationship between the two concepts because, in theory, if the corpus is large enough, the Word2Vec approach should be able to recognise the similarity between the related keywords between the two concepts. So it might just be the case that the corpus is not big enough. Hence, more experiments have been conducted here to evaluate the performance of Word2Vec with different corpus sizes, and the results are shown below:

Class Pair	Corpus Size (Candida Focused)			
	1000 (original result)	3000	10000	36703
lymphoma=>nhl	-0.12602892	-0.05788552	0.26840233	0.16148878
malignant_lymphoma => nhl	0.32301828	-0.09093403	0.28296840	0.36110824
malignant_lymphomas => nhl	0.23673100	0.12023943	0.20338778	0.11125774
lymphoid_leucosis => nhl	0.30389359	0.29170742	0.17907258	0.07976471
lymphoid => nhl	-0.12746886	-0.03416051	-0.02137199	0.10107623
malignant_lymphoma => malignant_lymphomas	0.57829564	0.42944559	0.41625821	0.15482646

Table 7-7 Word2Vec results with expanded Candida corpora

It is helpful to point out here that the corpus used in the above tests was constructed by adding more documents into the existing 1000 corpus. Those documents were selected by searching the keyword “Candida” in PubMed – the same way to construct the original Source and Target Corpus.

However, expanding the corpus size does not lead to a better result as suggested above. In fact, as Figure 7-8 shows below, cosine similarity for some of the keyword pairs even shows a decreasing trend. It is because Lymphoma and Lymphoma_Non-Hodgkin are cancer concepts. In other words, they are more closely related to the Cancer domain instead of Candida domain. Therefore, most of the corpus documents (about Candida) have nothing to do with them and certainly do not contain the related keywords. Consequently, adding a lot of unrelated documents into the corpus could be considered as pollution, which will, of course, lead to an even worse result.

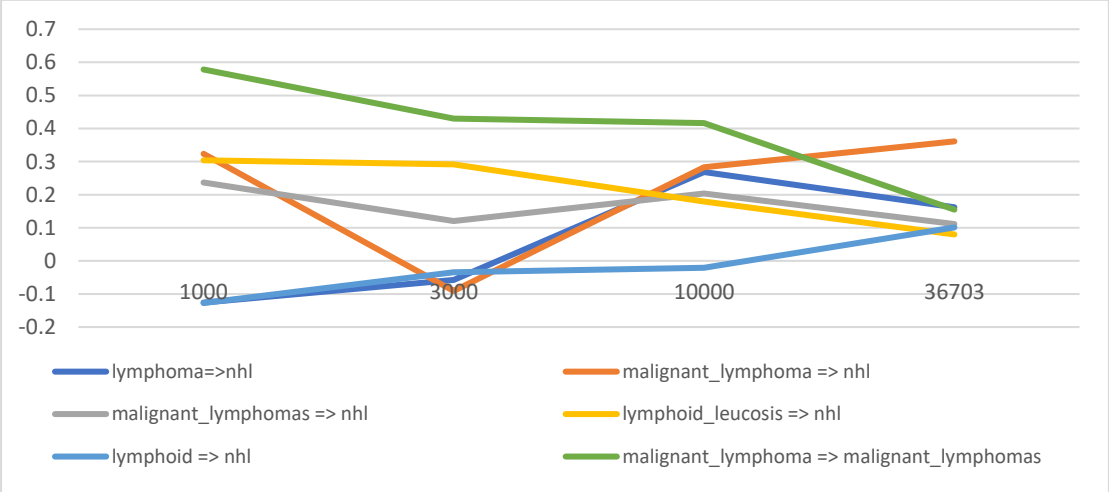


Figure 7-8 Cosine Similarity trend of the class pairs in Table 7-7, x-Axes is the corpus size, y-Axes is the cosine similarity value

If this is the case, the Word2Vec approach should be able to generate a better result by replacing the existing corpus, which focuses on Candida, with a Lymphoma focused corpus (by searching the keywords “Lymphoma” in

PubMed). As with the previous tests, a series of new tests have been conducted with new Lymphoma corpora, and the result is shown below:

Class Pair	Corpus Size (Lymphoma Focused)			
	1000	3000	10000	36703
lymphoma=>nhl	-0.126028925	0.405804008	0.645361722	0.764874756
malignant_lymphoma => nhl	0.323018283	0.214865893	0.392064750	0.623977780
malignant_lymphomas => nhl	0.236731008	0.134873793	0.260588676	0.518352568
lymphoid_leucosis => nhl	0.303893596	-0.022426017	-0.219005600	-0.021735659
lymphoid => nhl	-0.127468869	0.154514313	0.138115734	0.438723415
malignant_lymphoma => malignant_lymphomas	0.578295648	0.170317248	0.351530075	0.515104294

Table 7-8 Word2Vec results with Lymphoma corpora

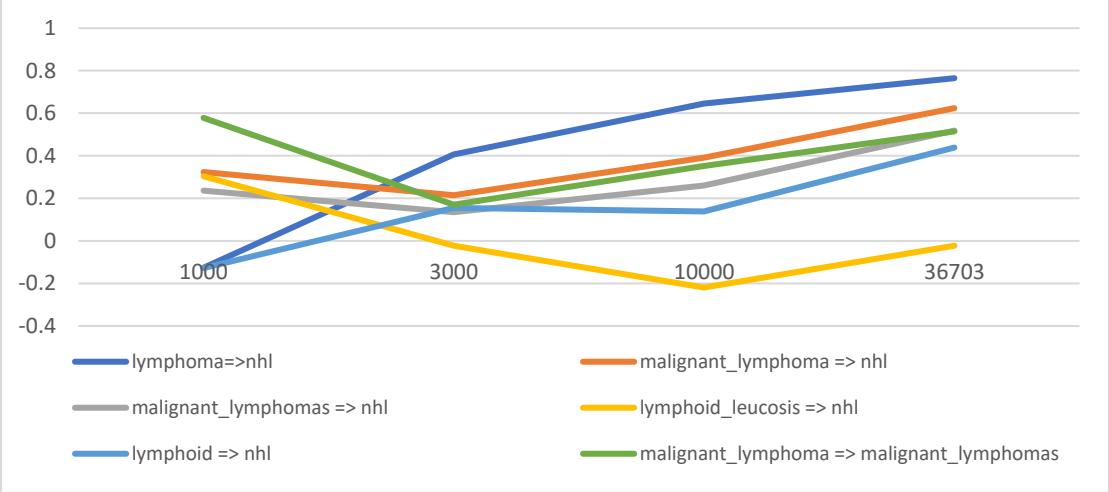


Figure 7-9 Cosine Similarity trend of the class pairs in Table 7-8, x-Axes is the corpus size, y-Axes is the cosine similarity value

By doing so, the Word2Vec results start showing the clustering trend like the SI result does. However, the problem is that the system supposes to focus on the Candida domain, not the Lymphoma domain. Hence, three conclusions can be made based on the above two tests, as explained below.

Firstly, the SI result suggests that Disease_D008223 (Lymphoma) and Disease_D008228 (Lymphoma_Non-Hodgkin) are close to each other. Therefore they should have a close relationship within the Candida domain at a deeper semantic level. It is a valid result because:

a) Intuitively this should be right because they share the same root word, although SI algorithm is not a symbolic-based approach (does not know and does not care that some of the concepts share the same root word).
b) Both of them have been classified as “Cancer | Immune system disease | Lymphatic disease” by the CTD.
c) The same/similar correlation can be observed by using the Word2Vec method on a large (35k+) Lymphatic corpus.

Secondly, as with domain knowledge, semantic representation is also domain-dependent. The traditional word embedding methods (e.g. Word2Vec) cannot effectively identify the relationship between words/concepts without having a carefully selected and domain-specific corpus (unless the concept itself is a generic concept like man and woman).

It brings up a tricky problem: assuming that we want to analyse a corpus about Domain A, which not only contains concepts that exclusively belong to it but also includes concepts that mainly exist in other domains. From the knowledge representation perspective, these non-Domain A concepts are also part of the Domain A knowledge and therefore need to be captured (e.g. the Cancer concept in the Candida domain).

In order to get an accurate semantic representation, it is critical to ensure there are sufficient documents in the corpus. In theory, the more Domain A related documents the corpus contains, the more accurate the result should be. However, in practice (as Table 7-7 shows above), the more documents it contains, the worse the results are for the non-Domain A concepts (for Domain A concepts, the result should be improved). The only way to enhance the result

for those non-Domain A concepts is to build and analyse a corpus that is related to their own domain (results in Table 7-8), which has nothing to do with Domain A. Simply expanding Corpus A to cover the other domains is not acceptable because it will reduce the accuracy for those original Domain A concepts and could even be considered as pollution from the knowledge representation perspective. In other words, it is difficult to accurately measure the semantic representation for both domain and non-domain related concepts with a traditional approach.

Thirdly, even if we can build a separate corpus for other domains (other than Domain A) and analyse them separately, there is no way to bring them back at a later stage to analyse their relationship with Domain A as a whole, which is what we want to do in the first place.

Based on the discussions above, it is reasonable to say the SI algorithm is a different and better approach. It relies less on a separate corpus to identify the semantic relationship between concepts in other domains. Hence, the system does not need to build separate corpora for different domains. Instead of doing so, it trains a separate neural network for all the individual concepts to handle the tricky issue discussed above.

7.4 Evaluation 4 (Figure 7-1) - Reproducibility

In the last section, we made some tests on an expanded Candida corpus and demonstrated that Word2Vec failed to recognise the closeness between Lymphoma and Non-Hodgkin Lymphoma. However, in order to claim that SI is a better approach (than Word2Vec), it is essential to show that a similar result

(that Lymphoma and Non-Hodgkin Lymphoma are close to each other) can be reproduced by using the SI approach on the expanded Candida corpus. This is what this section is aiming to do.

7.4.1 Expand Corpus to 3000

Using the same expanded (3000) corpus mentioned in Table 7-7, the system can eventually reproduce a similar C (closeness) value: 0.78248, but with a different NN structure.

The original NN structure used in the second experiment contained 3 hidden layers and 1500 nodes on each layer. However, applying this NN structure on the extended corpus (following the same process as discussed in Section 4.1) leads to a poor closeness result – 0.59893 calculated by Equation 7-2. However, it does not mean SI is unreliable and cannot reproduce the result. It is because the semantic distributions of the corpus concepts have changed (since the new corpora are three times larger than the original), and therefore the original NN structure, which was identified as the best NN structure in the second experiment, is no longer the best NN structure in this expanded scenario. In other words, we need to conduct more tests to identify the best NN structure in this expanded scenario.

By expanding the corpora, additional concepts will be identified, which will increase the processing time considerably. For example, 708 corpus concepts were identified from the original source corpus (as indicated in Table 6-1), but this number increased to 1420 in the expanded source corpus. In order to effectively run these tests, only the original corpus concepts are selected for

processing and the same Mapped Subset (as used in the second experiment) is used to identify the best NN structure. Table 7-9 below gives the results for all the tests conducted for this part of the evaluation (sorted by C value), and the best NN structure, in this case, contains 3 hidden layers and 2000 nodes on each layer.

NN Structure	C Value
HL=3, Nodes=2000	0.78249
HL=5, Nodes=2000	0.72837
HL=5, Nodes=1500	0.72505
HL=3, Nodes=1000	0.65329
HL=3, Nodes=2500	0.64498
HL=3, Nodes=1000, E=1000	0.62457
HL=3, Nodes=1500, E=700	0.62340
HL=3, Nodes=1500	0.59893
HL=9, Nodes=1500	0.46653

Table 7-9 C value generated by different NN structure, where HL = number of Hidden Layers, Nodes = number of Nodes, E = number of Epochs (default value is 350)

The above results suggest that unlike the Word2Vec approach, which failed to recognise the semantic closeness between Lymphoma and Non-Hodgkin Lymphoma within the expanded corpus, the SI approach can successfully identify the closeness and reproduce a similar result at the cost of identifying a new NN structure.

However, the need to change the best NN structure in this scenario indicates that 1000 documents in a corpus (or 2000 in both corpora) are not sufficient to cover all the knowledge and aspects within the Candida domain. In an ideal situation where there are sufficient documents in the corpus, the semantic distributions of the concepts should be more stable, and the Alignment Coefficient (AC) or the Informative Coefficient (IC) value should stay similar if additional documents are added to the corpus. In other words, the best NN structure should remain the same in the case of corpus expansion. We need to go through various tests (Table 7-9) to re-identify the best NN structure because

the original corpus (which only contains 1000 documents) does not have sufficient documents to stabilise the semantic distribution for various concepts within the corpus. It is, in fact, a defect of the second experiment. Unfortunately, there is no simple fix that we can adopt to resolve it without rerunning the whole experiment, which takes months to do as discussed in Section 6.2.1. Hence, it will be considered as future work that will be discussed in the next chapter.

Although we cannot resolve this issue, there is, in fact, a way to show the stability of the semantic distribution for all the concepts with sufficient documents in the corpus. In other words, the best NN structure should be the same after the corpus expansion if sufficient documents have been included in the corpus already. This is discussed in the below section.

7.4.2 Expand the Corpus by Duplication

Previously, the 3000 corpus was constructed by adding 2000 (4000 in total for both Source and Target Corpus) additional Candida related documents into the original 1000 (2000 in total) corpus used in the second experiment. These additional documents contain new semantic information (e.g. new entity name/keyword associated with a concept) and context (e.g. co-occurrence). It is the main reason why the semantic distribution of the corpus concepts has been affected, as discussed above.

However, there is a simple way to expand the corpus without introducing new semantic information and context -- duplicating the existing documents in the corpus.

So the rationale is this: duplicate the existing documents/texts in the corpora which were used in the second experiment and use the same NN structure to rerun the whole IC process. Then, the semantic information and context for the identified corpus concepts will remain the same. In this way, we can simulate the scenario of expanding a corpus that already contains sufficient documents (the newly added documents do not change the semantic distribution of the concepts), and rerun the SI algorithm with the same NN structure to generate a new result. If the new result has a similar Alignment Coefficient (AC) value, then it suggests that the best NN structure can be the same if sufficient documents have been included in the corpora.

The result is shown in Figure 7-10 below. The new AC value is 0.32210, which is very close to the old AC value (0.36149), and the average difference of the CS' value for each individual concept between the original result and the new result is -0.015505319. This result confirms two things:

1. The best NN structure could be the same with sufficient documents included in the corpora.
2. Compared with the Word2Vec method, SI can produce a more stable grouping trend, and the result is reproducible.

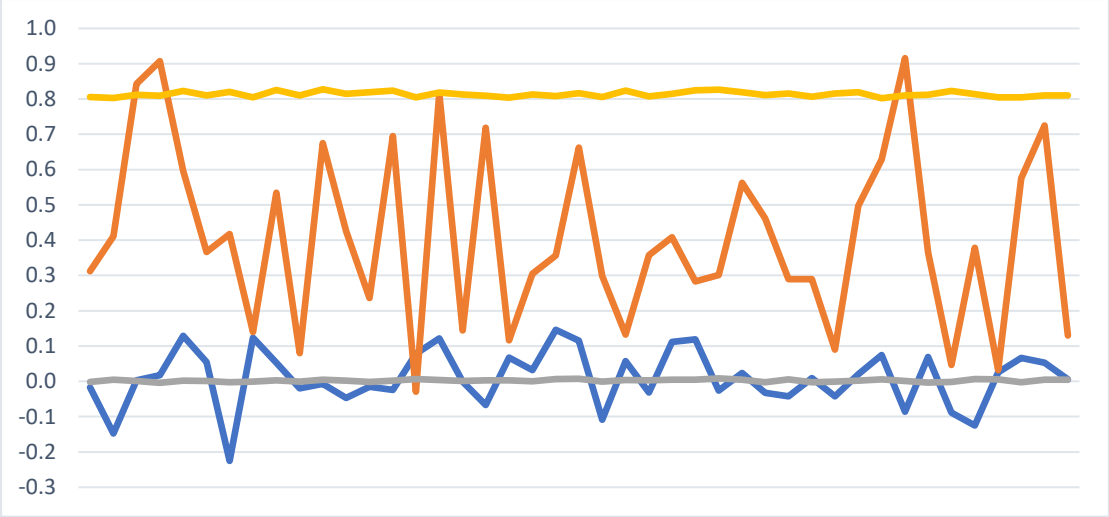


Figure 7-10 Reproduced results with duplicated documents

7.5 Evaluation 5 (Figure 7-1) - Stop Words Concept

Both Lymphoma and Non-Hodgkin Lymphoma selected in the last section have a high SI value, suggesting that they are semantically important to domain knowledge. Based on the evaluation result, it is reasonable to say the SI algorithm can produce a stable result on those important concepts. This section will focus on evaluating the performance of the SI algorithm on the non-important concepts. If the algorithm works as it should, then for a non-important concept, its SI value should be small.

However, instead of using an existing corpus concept, we have manually created a new concept for stop words since it is almost guaranteed that stop words should not be considered important at a deeper semantic level. It is why quite often, they are manually removed from NLP related tasks.

For evaluation purposes, “is”, “that” and “a” have been used to form this new concept – As with how to create the `W2V_<ConceptName>_Source` model, the system will go through the word-replacement process (discussed in Section 4.2.2) to replace “is”, “that” and “a” from the original text with an invented unique

string, and use the modified text to generate new models called `W2V_IsThatA_Source` and `W2V_IsThatA_Target`. Then, use the same NN structure (3 hidden layers with 1500 nodes on each layer) and go through the Neural Complex process discussed in Section 4.3.3 to generate the $CS'_{IsThatA}$ value. After that, calculate its confidence score (Equation 6-2, p. 152) by rerunning the Neural Complex process. Then calculate the IC and CC score as discussed previously. The final results are shown in Table 7-10 below.

Parameter	Value
CS'	0.02947
Confidence Score	0.94368
IC	0.02781
CC	15.47415
Normalised IC	-0.61397
Normalised CC	0.04288
SI	-0.28554

Table 7-10 Results for the stop words concept

As the above result suggests, it has a low SI (mean of Normalised IC and Normalised CC) value which put its overall ranking to 241 of 285. This aligns well with the expectation and indicates that SI can also produce reliable results on those non-important concepts.

7.6 Summary

For various reasons, it is difficult to evaluate the final SI result (Appendix VII) at the individual level, and difficult to find domain experts[40] to manually compare the results with a gold standard. Hence, an alternative strategy has been

[40] We tried to identify some domain experts to do the manual evaluation at the beginning of this research. However, it turns out really difficult to do so. Hence, as a future line of work, we can identify a different domain which has a gold standard exist, then ask domain experts to manually evaluate the result. Refer to Section 8.2.2 for more details.

proposed in this chapter to evaluate the overall performance of the SI algorithm from five different angles (5 items listed at the beginning of this chapter).

Section 7.1.1 and Section 7.1.2 corresponded to the first item – instead of evaluating the SI result as a whole, we have demonstrated that the outcomes of the individual components (of SI) are reasonable and correct. Section 7.1.1 used a quantitative method to show the IC is a consistent process instead of a randomly generated coincidence. Section 7.1.2 demonstrated that the results from the CC process conform to conceptual connections (e.g. the concept of `Death` has been identified as the most correlated concept with `Sudden Death`) that are evidenced by various authoritative documents that were consulted.

The grouping/clustering trend of the SI result (item 2) was evaluated in Section 7.2. Using the Hopkins Statistic, we successfully demonstrated that the concepts were randomly distributed in the sample space but showed a strong grouping/clustering trend in the final SI result.

Section 7.3 and Section 7.4 focused on item 3 and item 4 and demonstrated that, compared with traditional methods like Word2Vec, SI relies less on a domain-specific corpus to identify the semantic relationship between (important) corpus concepts. We also explained why it is difficult or even impossible to achieve a similar result by using an existing method (e.g. Word2Vec), and therefore demonstrated the advantage of the SI approach. We then discussed the reproducibility of the SI approach and demonstrated that it is reliable and could produce a stable result.

The last part (item 5) of the evaluation – Section 7.5 provided evidence that the SI not only works on those semantically important concepts but also is able to produce a good result for the non-important concepts in the domain.

In short, it has been demonstrated that each component within the SI could produce a good and consistent result. At the macro-level, the overall SI result shows a strong clustering trend, although SI itself is not an algorithm aiming at clustering or classification. At the micro-level, the SI results for both semantically important and non-important concepts are reasonable and reproducible. Compared with a traditional method like Word2Vec, SI can identify various semantic relationships across domains without building a separate domain-specific corpus.

Chapter 8 Conclusion and Future Work

Semantic Impact (SI) is a novel method proposed in this thesis to derive a numerical measure that summarises how strongly a concept impinges on the domain of discourse. Compared with the other measures discussed in Section 2.4, SI is a predictive-based approach and provides objective and consistent measurement at a deeper semantic level. Moreover, it neither relies on pre-defined domain knowledge (noting that the guiding ontology is for a different domain or describes the domain from a different perspective) nor additional knowledge obtained from the human intervention for decision-making purposes (except the process to identify the best NN structure, which will be further discussed in this chapter).

The next section evaluates the answers to the two research questions raised in Chapter 1.

8.1 Research Questions Revisited and Main Contributions

Let us start with RQ2[41]: By going through the Exploratory Semantic Analysis (ESA) step (Section 4.2), two Distributional Semantic Model (DSM) sets have been produced corresponding to the Source and Target Corpus. Essentially, the system will produce an individual Word2Vec model for all valid corpus concepts, `W2V_<ConceptName>_Source/Target` (Section 4.2.2), then uses the associated vectors as the input of the Informative Coefficient (IC) and Connectivity Coefficient (CC) calculations. These associated concept vectors

[41] How to make the measure objective and consistent at a deeper semantic level?

are, in fact, the semantic representation of the concepts in the related high-dimensional space. In this way, we can ensure the SI algorithm operates at a deeper semantic level (this partly answers the second research question).

SI is a predictive-based approach, which aligns with the developing trend in the DSM and Embedding study, as discussed in Section 2.4. Other predictive-based approaches also operate at the semantic level but, quite often, they are one-sided. For example, [53][54] discussed an approach to use Word2Vec for concept selection. In each case, what they actually measured was the semantic closeness between a specific word (concept name) and another word in a pre-defined seed list. In other words, the importance was assessed only from the closeness aspect. This method might be appropriate in some cases, but there is a difference between semantic closeness and semantic importance. Using Figure 7-7 (p. 193) as an example, those words related to countries' capital (concept names) could be considered as semantically close to each other. However, considering a specific country, e.g. France, as the domain, then the other capitals are clearly not as semantically important as Paris to the domain, although they may be still semantically close to each other.

The SI approach, however, has a more thorough definition and includes both informativeness (measured by the IC process discussed in Section 4.2) and connectivity (measured by the CC process discussed in Section 4.3) into the measurement. As demonstrated in Section 7.1.2 already, the IC and CC are two independent processes. A high IC value means a concept is semantically enriched. However, an enriched concept may not necessarily have a strong connection with the other concepts in the domain. In order to be considered as

important in the SI approach, a concept should be both semantically enriched and have solid connectivity to be able to influence the other domain concepts to make an impact on the domain knowledge. Hence, SI is more objective than the other predictive-based approaches (this partly answers the second research question).

Besides the objective aspect, which was discussed above, consistency is another crucial characteristic of the SI approach. Here, the "consistency" is two-fold as follows.

As part of the IC process, a Mapped Subset (which contains a list of concepts that are more likely to be informative concepts) was used to identify the best neural network (NN) structure, as discussed in Section 4.3.3 (and Section 6.3.1). Then the same NN structure was used to re-train a new neural network for each valid corpus concept instead of allowing them to have a different NN structure. In this way, we can ensure all the different corpus concepts have been assessed equally with a consistent approach (the same NN structure). So the meaning of the first fold of the consistency is a consistent way to assess the informativeness level for all the valid corpus concepts.

Secondly, as discussed in Section 4.3.3, the Informative Coefficient is, in fact, using the overfitting mechanism to identify the semantic complexity of the corpus concept. For an informative concept, it should have a complex semantic distribution to overcome the potential overfitting to a certain extent. As a result, it will have a more consistent semantic representation (a CS' value closer to 1) between the source corpus and the target corpus. Hence, it is reasonable to

say that the IC is, in fact, a way to measure the consistency of the semantic representation, which is the meaning of the second fold.

In short, we are using a consistent way to measure the consistency of the semantic representation (which reflects the informativeness) of the valid corpus concepts.

Above are the answers to the second research question. We now turn to the first question (RQ1[42]).

As already mentioned at the beginning of this chapter, the SI approach itself does not require pre-defined domain knowledge. However, it relies on third-party Named Entity Recognition (NER) tools/systems to do the semantic information extraction (Section 4.2.1). For example, in the first system, the IBM-NLU system was used to identify the concepts within the news domain and their associated words. In addition to the NER system, a guiding ontology is also needed. We fully accept that knowledge about a specific domain must come from somewhere. However, instead of using pre-defined knowledge of that specific domain (or a specific perspective of the domain), this research proposes a way to "transfer" the required knowledge from a related domain (or a related perspective of the same domain). In short, the system will still need some pre-defined knowledge, but the knowledge is not directly related to the given domain (or given perspective).

[42] How to reduce the level of human intervention required in the concept selection and make the overall OL process less reliant on pre-defined domain knowledge?

As regards human intervention: in order to implement the SI algorithm, it is necessary to decide which NER tool/system is going to be used to extract semantic information extraction. However, as already pointed out previously, SI is focusing on measuring the semantic importance of the identified concepts instead of producing a new method for concept extraction.

Hence, within the SI algorithm itself, there are only two places (stages) that require human intervention: a) Manually selecting a guiding ontology and b) Manually adjusting the neural network (NN) structure to identify the most suitable structure as discussed in Section 4.3.3.2, Section 5.3.1 and Section 6.3.1. Moreover, there is a line of future work discussed later in this chapter (Section 8.2.3) to automate this process.

In short, due to the novelty of the SI algorithm, it has been designed to provide objective and consistent measurement at a deeper semantic level, and the whole process is less reliant on pre-defined domain knowledge and human intervention than the other measurements.

The main contributions of this research (other than answering the two research questions) can be summarised as:

- We have invented a new semantic importance measurement called the Semantic Impact (SI) to objectively and consistently assess the importance of domain concepts at a relatively deep semantic level.
- As an extension of the Distributional Semantic Models (DSM) theory, we have added to knowledge about using a high-dimensional vector to infer the semantic representation of a concept – a collection of words that

belong to the same semantic group. Moreover, an individual concept should have its own unique semantic distribution pattern, which is also embedded in the high-dimensional vector. This unique semantic distribution pattern can be used to support various downstream NLP tasks[43].

- Furthermore, it has been demonstrated in this research that the informativeness of a concept also has a reflection in its high-dimensional vector and will be captured by its distribution pattern. It leads to the discovery of the phenomenon described below.
- An interesting phenomenon has been discovered – Concepts' semantic complexity can affect the overfitting-ness of the NN used in the Coordinate Transformation process (Section 4.3.2). More informative concepts are less likely to overfit the relevant NN(s). As a result, it provides a new method to assess how informative a concept is, as described in the next bullet.
- Based on that phenomenon, a novel approach called Neural Complex (NC, Section 4.3.3) has been proposed. NC can be used to resolve an ambiguous problem where there is no objective and consistent way to measure the correctness of the result.
- A novel approach has been identified in this research to adapt the Maximal Information Coefficient (MIC) to calculate the strength of the connection (Connectivity Coefficient, CC) between concepts within a domain. Measuring the connectivity between concepts will make the

[43] For example, in this research we have used this unique distribution pattern to identify informative concepts.

Semantic Impact algorithm (described in the first bullet) more comprehensive than traditional measurements discussed in Section 2.4.

8.2 Improvement and Future Work

Through the evaluation discussed in Chapter 7, we have demonstrated that the result produced by the SI algorithm is reasonable and reliable. This section will briefly discuss some of the improvements that can be made in the future.

8.2.1 Increasing the Corpus Size

As mentioned at the end of Section 7.4.1, the original corpora used in the second experiment may not contain sufficient documents to produce a stable semantic distribution (for corpus concepts), which may have an impact on the final SI result.

Hence, one possible line of future work is to expand the original corpus used in the second experiment and compare the new result with the original result discussed in this thesis.

We can, of course, build massive corpora that include a considerable number of documents to stabilise the semantic distribution. However, doing so will significantly increase the processing time.

More importantly, as Section 7.4.2 suggested, once the corpora contain sufficient documents, the best NN structure would remain the same with a similar Alignment Coefficient (AC) value produced. Therefore, it is reasonable to believe that expanding corpora that already contained sufficient documents to stabilise the semantic distribution may not improve the final SI result further.

As a result, by building the corpora with the minimum number of documents that can produce a stable semantic distribution, the overall process time can be reduced without affecting the final result.

It is easy to check if the corpora contain sufficient documents. Section 7.4.2, in fact, already discussed the basic idea – by comparing the AC value produced by the expanded corpora with the AC value produced by the original corpora. Hence, the first line of future work is to identify the best corpus size for the second experiment, then compare the result generated with the new corpora with the original result to see how it affects the SI.

8.2.2 Additional Evaluation Approach

As discussed in Chapter 7, it is difficult to evaluate the SI result at the individual concept level, and therefore a new evaluation strategy was proposed and used to assess the final SI result from five different angles. In the future, we would like to expand the evaluation strategy further to include the expert review (if possible).

For example, we could select a new domain that has a gold standard ontology that we can compare with. We could then set up two user groups to manually review the findings against the gold standard. The first user group, called the Ontology Expert Group, would be formed by a group of people with ontology experience. On the other hand, the second group would be the Domain Expert Group and would be formed by a group of people with a deep understanding of the domain itself. We could then ask these two groups to compare the SI results

with the concepts included in the gold standard ontology to evaluate the correctness of the SI results.

8.2.3 Develop an Optimiser for the Neural Complex

One of the objectives of the SI approach is to reduce the level of human intervention required in the concept selection process (once the guiding ontology has been chosen). Since the second experiment improved and automated the mapping process (discussed in Section 6.2.2.2), the only place that requires human intervention is the process to identify the best NN structure, where we need to manually analyse and adjust the parameters (e.g. the number of hidden layers) to maximise the Alignment Coefficient (AC) value.

Previously, we have discussed that the way to identify the best neural network structure is by using a Mapped Subset (which contains a list of concepts that are more likely to be informative concepts) to maximise the Alignment Coefficient value (Section 6.3.1). It is how we train this neural complex: the Mapped Subset, in this case, is the training dataset, and we use the loss function (AC) to determine whether it (the NN structure) is a good result or not, then adjust the neural network structure accordingly. Subsequently, applying the identified neural network structure to the rest of the concepts is the equivalent of applying this neural complex on the testing/real dataset.

One of the most significant pieces of work is to develop a mathematical approach (a systematic way like gradient descent) to determine which parameter (e.g. the number of hidden layers, or the number of nodes) the system needs to change, and exactly how to change it, to produce a better

result (assessed by using the loss function). In other words, develop an optimiser for the Neural Complex (NC).

So, the third line of future work is to refine this idea further and develop the related framework to automate the training process.

8.2.4 A Contest for Higher AC Score

The purpose of the Coordinate Transformation (CT) process (discussed in Section 4.3.2) is to train a neural network to align two Word2Vec models. Multiple CT processes combined to form the Neural Complex as discussed in Section 4.3.3.

In both experiments conducted in this thesis, we tested quite a few different structures to identify the best NN structure. However, we have not, in fact, changed the type of neural network. More specifically, we only changed some of the basic configurations/settings of the neural network (e.g. the number of nodes on each hidden layer), but it is still a fully connected feedforward neural network, as illustrated in Figure 4-5 (p. 71).

The highest Alignment Coefficient (AC) value we managed to achieve by adjusting the basic configuration was 0.36149 in the second experiment (3 hidden layers and 1500 nodes on each layer, Table 6-3, p. 160). One interesting possibility for future work is implementing a different type of neural network, e.g. LSTM NN, and see how if a higher AC score can be achieved. In theory, the higher the AC score is, the more accurate the IC result will be.

Changing to a different NN type (e.g. LSTM) will not affect the overall Neural Complex architecture or its training process (Figure 4-14, p. 86). Hence, we are providing an opportunity, or a contest, for people to experiment with different types of NN to reach a higher AC score.

As mentioned previously, the highest AC score we have achieved in the second experiment was 0.36149 (which can be considered as a baseline), and the related resources are provided in the footnote[44]. In theory, the maximum value of AC is 1, so there is still space for improvement.

It is recognised by the author (of this thesis) that a close to 1 score (e.g. 0.9+, in fact, any value that bigger than 0.7) could be challenging to achieve due to how the mapping process has been implemented in the second experiment. However, the author is more than happy to see how the community could disprove him.

8.2.5 "Draco dormiens nunquam titillandus"[45]

At the beginning of the thesis, an example of Harry Potter was given to explain the motivation of this research, and indicated that a good (objective and consistent) measure should be able to identify that `MagicalCreature` is less important in the main story but plays a significant role in the prequel.

---

[44] Word2Vec (DSM) Sets (for both source and target) generated for the second experiment: https://edata.bham.ac.uk/643/ (20GB)
Neural Network (3 hidden layers and 1500 nodes on each layer) Set trained for the second experiment: https://edata.bham.ac.uk/644/ (47GB)

[45] The motto of Hogwarts.

So, as a future line of work and the last section of this thesis, I would like to take people back to Hogwarts and see how could we (as Muggles) use the SI algorithm to address the `MagicalCreature` challenge.

A slight change to how we have implemented SI in the two experiments discussed in this thesis could be adopted: instead of building a source corpus and target corpus with documents that describe the same domain from the same perspective, we could construct the source corpus based on the main story and the target corpus based on the prequel while implementing a guiding ontology about the main story or simply providing a list of informative/important concepts within the main story (e.g. `Wizard` and `Student` as discussed previously, but excluding `MagicalCreature`). Then the same process as we have discussed in this thesis can be followed to produce the IC score.

When calculating the CC score, the system needs to generate a sample table (e.g. Table 4-3, p. 102) and use it as the input to produce the MIC value for each concept pair. In the two experiments discussed, the sample table was produced based on the source corpus (source DSM set). In this new proposed MagicalCreature experiment, we need to use the target corpus (target DSM set) to build the sample table since it is the connectivity in the prequel that we need to measure. The rest of the process remains the same.

It would be interesting to see what SI score the `MagicalCreature` would get in this new experiment. Ideally, it should have a high SI score since what we measure here is the prequel. After that, switch the source corpus and target corpus and use a guiding ontology about the prequel (or simply states the important/informative concepts such as `Wizard` and `MagicalCreature`, and

exclude `Student`) and re-run the process to produce a new SI value for `MagicalCreature` (and `Student`). In this case, the SI for `Student` should be high, and the value for `MagicalCreate` should be low.

This new experiment (as a future line of work) will help us better understand the behaviour of SI in the scenario of having two different (but closely related) corpora. In case of success, we can further expand this experiment to build a source corpus based on Harry Potter and a target corpus based on the Lord of the Rings, then assess what the important concepts are in the latter based on the knowledge of Harry Potter. It would be great fun to see what will happen when relocating Hogwarts to Middle-earth and when Dumbledore meets Gandalf.

At last, there are many interesting things about the Semantic Impact that we can further explore in the future. This is the end of the thesis but is also the beginning of the Semantic Impact exploration, and the only way to make great achievements in this coming exploration is through hard work, as to how it states in the motto of the University of Birmingham – “Per Ardua ad Alta”.

References

[1] B. Smith and C. Welty, “Ontology: Towards a new synthesis,” in *Formal Ontology in Information Systems: Collected Papers from the Second International Conference*, 2001, pp. 3–9.

[2] B. Yildiz, “Ontology Evolution and Versioning,” Vienna University of Technology, Karlsplatz, 2006.

[3] D. M. W. Powers, “Bibliographies and Literature Reviews Goals , Issues and Directions in Machine Learning of Natural Language and Ontology,” *SIGART Bull.*, vol. 1, no. 1, pp. 101–114, 1986.

[4] D. M. W. Powers and C. C. R. Turk, *Machine Learning of Natural Language*. Springer-Verlag, 1989.

[5] D. M. W. Powers, “Natural language the natural way,” *Computer Compacts*, vol. 2, no. 3–4, pp. 100–109, 1984, doi: 10.1016/0167-7136(84)90088-X.

[6] G. L. Zúñiga, “Ontology: Its Transformation from Philosophy to Information Systems,” in *Formal ontology in information systems: Proceedings of the second international conference (FOIS’01)*, 2001, pp. 187–197, doi: 10.1145/505168.505187.

[7] A. Maedche and S. Staab, “Ontology Learning for the Semantic Web,” *IEEE Intelligent Systems*, vol. 16, no. 2, pp. 72–79, 2001, doi: 10.1109/5254.920602.

[8] I. Horrocks, “Ontologies and the semantic web,” *Communications of the ACM*, vol. 51, no. 12, pp. 58–67, 2008, doi: 10.1145/1409360.1409377.

[9] S. Pulman, “Distributional Semantic Models,” 2013. doi: 10.1093/acprof:oso/9780199646296.003.0012.

[10] D. N. Reshef *et al.*, “Detecting novel associations in large data sets,” *Science*, vol. 334, no. 6062, pp. 1518–1524, Dec. 2011, doi: 10.1126/science.1205438.

[11] J. Wan, J. Barnden, B. Hu, and P. Hancox, “What Is Semantically Important to ‘Donald Trump’?,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2019, vol. 11354 LNCS, pp. 314–329, doi: 10.1007/978-3-030-15127-0_31.

[12] Y. Ding and S. Foo, “Ontology research and development. Part I - A review of ontology generation,” *Journal of Information Science*, vol. 28, no. 2, pp. 123–136, Apr. 2002, doi: 10.1177/016555150202800204.

[13] Y. Ding and S. Foo, “Ontology research and development. Part 2 - A review of ontology mapping and evolving,” *Journal of Information Science*, vol. 28, no. 5, pp. 375–388, Oct. 2002, doi: 10.1177/016555150202800503.

[14] A. Gómez-pérez and D. Manzano-macho, “A survey of ontology learning methods and techniques OntoWeb Consortium,” OntoWeb Consortium, 2003.

[15] W. Wong, W. Liu, and M. Bennamoun, “Ontology learning from text: A look back and into the future,” *ACM Computing Surveys*, vol. 44, no. 4, pp. 1–36, 2012, doi: 10.1145/2333112.2333115.

[16] L. Zhou, “Ontology learning: State of the art and open issues,” *Information Technology and Management*, vol. 8, no. 3, pp. 241–252, 2007, doi: 10.1007/s10799-007-0019-5.

[17] M. N. Asim, M. Wasim, M. U. G. Khan, W. Mahmood, and H. M. Abbasi, “A survey of ontology learning techniques and applications,” *Database*, vol. 2018, no. 2018. Oxford University Press, Jan. 01, 2018, doi: 10.1093/database/bay101.

[18] C. H. Wei, A. Allot, R. Leaman, and Z. Lu, “PubTator central: automated concept annotation for biomedical full text articles,” *Nucleic Acids Research*, vol. 47, no. W1, pp. W587–W593, Jul. 2019, doi: 10.1093/nar/gkz389.

[19] F. Petroni *et al.*, “Language models as knowledge bases?,” in *EMNLP-IJCNLP 2019 - 2019 Conference on Empirical Methods in Natural Language Processing and 9th International Joint Conference on Natural Language Processing, Proceedings of the Conference*, Sep. 2020, pp. 2463–2473, doi: 10.18653/v1/d19-1250.

[20] R. Munday, “Glossary of Terms in Being and Time,” 2013. http://www.visual-memory.co.uk/b_resources/b_and_t_glossary.html.

[21] N. F. Noy and D. L. McGuinness, “Ontology Development 101: A Guide to Creating Your First Ontology,” *Stanford Knowledge Systems Laboratory*, 2001. http://www.ksl.stanford.edu/people/dlm/papers/ontology-tutorial-noy-mcguinness-abstract.html (accessed Feb. 04, 2021).

[22] K. T. Frantzi, S. Ananiadou, and J. Tsujii, “The C-value/NC-value method of automatic recognition for multi-word terms,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1998, vol. 1513, pp. 585–604, doi: 10.1007/3-540-49653-x_35.

[23] G. A. Miller and W. G. Charles, “Contextual Correlates of Semantic Similarity,” *Language and Cognitive Processes*, vol. 6, no. 1, pp. 1–28, Jan. 1991, doi: 10.1080/01690969108406936.

[24] M. Arguello Casteleiro *et al.*, “Ontology learning with deep learning: A case study on patient safety using PubMed,” in *CEUR Workshop Proceedings*, 2016, vol. 1795.

[25] Ed Grefenstette, “Lecture 2a- Word Level Semantics,” 2017. https://github.com/oxford-cs-deepnlp-2017/lectures.

[26] Z. S. Harris, “Distributional Structure,” in *Papers on Syntax*, Springer Netherlands, 1981, pp. 3–22.

[27] S. T. Dumais, “Latent Semantic Analysis,” *Annual Review of Information Science and Technology*, vol. 38, no. 1, pp. 188–230, Jan. 2004, doi: 10.1002/aris.1440380105.

[28] M. Baroni, G. Dinu, and G. Kruszewski, “Don’t count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors,” in *52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014 - Proceedings of the Conference*, 2014, vol. 1, pp. 238–247, doi: 10.3115/v1/p14-1023.

[29] “XLNet — SOTA pre-training method that outperforms BERT.” https://medium.com/logits/xlnet-sota-pre-training-method-that-outperforms-bert-26d4e9978983 (accessed Apr. 29, 2021).

[30] T. Mikolov, W. T. Yih, and G. Zweig, “Linguistic regularities in continuous spaceword representations,” in *NAACL HLT 2013 - 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Main Conference*, 2013, pp. 746–751.

[31] T. Mikolov, Q. V. Le, and I. Sutskever, “Exploiting Similarities among Languages for Machine Translation,” Sep. 2013.

[32] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space,” Jan. 2013.

[33] M. E. Peters *et al.*, “Deep contextualized word representations,” in *NAACL HLT 2018 - 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference*, Feb. 2018, vol. 1, pp. 2227–2237, doi: 10.18653/v1/n18-1202.

[34] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language*

Technologies - Proceedings of the Conference, Oct. 2019, vol. 1, pp. 4171–4186, Accessed: Oct. 06, 2020. [Online]. Available: http://arxiv.org/abs/1810.04805.

[35] P. Blunsom, “Lecture 3 - Language Modelling and RNNs Part 1.” Oxford University, 2017.

[36] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, “A Neural Probabilistic Model,” *Journal of Machine Learning Research*, vol. 3, pp. 1137–1155, 2003, doi: 10.5555/944919.944966.

[37] E. H. Huang, R. Socher, C. D. Manning, and A. Y. Ng, “Improving word representations via global context and multipleword prototypes,” Association for Computational Linguistics, 2012.

[38] D. R. Tobergte and S. Curtis, *Quick Training of Probabilistic Neural Nets by Importance Sampling*, vol. 53, no. 9. 2013, pp. 1689–1699.

[39] X. Rong, “word2vec Parameter Learning Explained,” Nov. 2014, doi: https://doi.org/10.48550/arXiv.1411.2738.

[40] D. A. Huffman, “A Method for the Construction of Minimum-Redundancy Codes,” *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098–1101, 1952, doi: 10.1109/JRPROC.1952.273898.

[41] T. Demeester, T. Rocktäschel, and S. Riedel, “Lifted rule injection for relation embeddings,” *EMNLP 2016 - Conference on Empirical Methods in Natural Language Processing, Proceedings*. pp. 1389–1399, 2016, doi: 10.18653/v1/d16-1146.

[42] Q. Le and T. Mikolov, “Distributed representations of sentences and documents,” *31st International Conference on Machine Learning, ICML 2014*, vol. 4, pp. 2931–2939, May 2014.

[43] Y. Zhu *et al.*, “Aligning books and movies: Towards story-like visual explanations by watching movies and reading books,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, vol. 2015 Inter, pp. 19–27, doi: 10.1109/ICCV.2015.11.

[44] Y. Wu *et al.*, “Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation,” Sep. 2016, doi: http://arxiv.org/abs/1609.08144.

[45] P. Vincent, H. Larochelle, Y. Bengio, and P. A. Manzagol, “Extracting and composing robust features with denoising autoencoders,” in *Proceedings of the 25th International Conference on Machine Learning*, 2008, pp. 1096–1103, doi: 10.1145/1390156.1390294.

[46] M. E. Peters, M. Neumann, L. Zettlemoyer, and W. T. Yih, “Dissecting contextual word embeddings: Architecture and representation,” in

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, EMNLP 2018, 2020, pp. 1499–1509, doi: 10.18653/v1/d18-1179.

[47] Y. Goldberg, “Assessing BERT’s syntactic abilities,” *arXiv*, Jan. 2019, doi: http://arxiv.org/abs/1901.05287.

[48] I. Tenney *et al.*, “What do you learn from context? Probing for sentence structure in contextualized word representations,” *arXiv*, May 2019, doi: http://arxiv.org/abs/1905.06316.

[49] P. Cimiano and J. Völker, “Text2Onto A framework for ontology learning and data-driven change discovery,” *Lecture Notes in Computer Science*, vol. 3513, pp. 227–238, 2005, doi: 10.1007/11428817_21.

[50] S. M. Yang, X. Bin Wu, Z. H. Deng, M. Zhang, and D. Q. Yang, “Relative term-frequency based feature selection for text categorization,” in *Proceedings of 2002 International Conference on Machine Learning and Cybernetics*, 2002, vol. 3, pp. 1432–1436, doi: 10.1109/icmlc.2002.1167443.

[51] R. Navigli, P. Velardi, and A. Gangemi, “Ontology Learning and Its Application to Automated Terminology Translation,” *IEEE Intelligent Systems*, vol. 18, no. 1, pp. 22–31, 2003, doi: 10.1109/MIS.2003.1179190.

[52] G. Wohlgenannt and F. Minic, “Using word2vec to build a simple ontology learning system,” in *CEUR Workshop Proceedings*, 2016, vol. 1690.

[53] N. Gupta, S. Podder, S. Sengupta, and K. M. Annervaz, “Domain Ontology Induction Using Word Embeddings,” in *15th IEEE International Conference on Machine Learning and Applications (ICMLA)*, Feb. 2017, pp. 115–119, doi: 10.1109/icmla.2016.0027.

[54] N. Mahmoud, H. Elbeh, and H. M. Abdlkader, “Ontology learning based on word embeddings for text big data extraction,” in *ICENCO 2018 - 14th International Computer Engineering Conference: Secure Smart Societies*, Feb. 2019, pp. 183–188, doi: 10.1109/ICENCO.2018.8636154.

[55] Z. Zhang *et al.*, “Semantics-aware BERT for language understanding,” *arXiv*, vol. 34, no. 05, pp. 9628–9635, Sep. 2019, doi: 10.1609/aaai.v34i05.6510.

[56] J. Wan and J. Barnden, “A new semantic model for domain-ontology learning,” in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2015, vol. 8944, pp. 140–155, doi: 10.1007/978-3-319-15554-8_12.

[57] RARE Technologies, “Recipes & FAQ · RaRe-Technologies/gensim Wiki.” https://github.com/RaRe-Technologies/gensim/wiki/Recipes-&-FAQ#q11-ive-trained-my-word2vecdoc2vecetc-model-repeatedly-using-the-exact-same-text-corpus-but-the-vectors-are-different-each-time-is-there-a-bug-or-have-i-made-a-mistake-2vec-training-non-determ (accessed Mar. 08, 2021).

[58] J. Brownlee, “A Gentle Introduction to the Rectified Linear Unit (ReLU),” *Machinelearningmastery.Com*, 2019. https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/ (accessed Apr. 28, 2021).

[59] X. Glorot and Y. Bengio, “Understanding the difficulty of training deep feedforward neural networks,” in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, 2010, pp. 249–256, Accessed: Mar. 02, 2022. [Online]. Available: http://proceedings.mlr.press/v9/glorot10a.

[60] D. P. Kingma and J. L. Ba, “Adam: A Method for Stochastic Optimization,” *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, Dec. 2014, Accessed: Mar. 02, 2022. [Online]. Available: https://arxiv.org/abs/1412.6980v9.

[61] S. Data, O. Diagram, and O. Terms, “Core Concepts Ontology,” 2018. https://www.bbc.co.uk/ontologies/coreconcepts.

[62] Y. Zhang, J. Schneider, and A. Dubrawski, “Learning the semantic correlation: An alternative way to gain from unlabeled text,” in *Advances in Neural Information Processing Systems 21 - Proceedings of the 2008 Conference*, 2009, vol. 21, pp. 1945–1952, Accessed: Mar. 14, 2021. [Online].

[63] IBM, “Watson Natural Language Understanding - Overview | IBM,” 2020. https://www.ibm.com/cloud/watson-natural-language-understanding (accessed Mar. 19, 2021).

[64] A. Abdellatif, K. Badran, D. E. Costa, and E. Shihab, “A Comparison of Natural Language Understanding Platforms for Chatbots in Software Engineering,” *IEEE Transactions on Software Engineering*, Dec. 2020, doi: 10.1109/TSE.2021.3078384.

[65] IBM, “Watson Natural Language Understanding - Overview | IBM,” 2020. https://www.ibm.com/cloud/watson-natural-language-understanding (accessed Apr. 04, 2021).

[66] BBC, “Trump says terror attacks ‘under-reported’: Is that true?,” *US & Canada*, 2017. http://www.bbc.co.uk/news/world-us-canada-38890090 (accessed Feb. 02, 2018).

[67] A. Ds, “Eclipse Deeplearning4j,” 2020. https://projects.eclipse.org/proposals/eclipse-deeplearning4j (accessed Mar. 20, 2021).

[68] J. T. Heaton, *Introduction to Neural Networks with Java*, 1 st. Heaton Research, 2005.

[69] Google Research, “google-research/bert: TensorFlow code and pre-trained models for BERT.” https://github.com/google-research/bert (accessed Mar. 05, 2022).

[70] K. Lee *et al.*, “HiPub: Translating PubMed and PMC texts to networks for knowledge discovery,” *Bioinformatics*, vol. 32, no. 18, pp. 2886–2888, 2016, doi: 10.1093/bioinformatics/btw511.

[71] J. Li *et al.*, “Annotating chemicals, diseases, and their interactions in biomedical literature,” in *Proceedings of the fifth BioCreative challenge evaluation workshop*, 2015, pp. 173–182.

[72] C. H. Wei *et al.*, “Accelerating literature curation with text-mining tools: a case study of using PubTator to curate genes in PubMed abstracts.,” *Database : the journal of biological databases and curation*, vol. 2012, 2012, doi: 10.1093/database/bas041.

[73] M. Ashburner *et al.*, “Gene ontology: Tool for the unification of biology,” *Nature Genetics*, vol. 25, no. 1, pp. 25–29, 2000, doi: 10.1038/75556.

[74] C. The Gene Ontology, I. That, M. Acencio, A. Lægreid, M. Kuiper, and O. Among, “The Gene Ontology Resource: 20 years and still GOing strong.,” *Nucleic Acids Research*, vol. 8, no. 47, pp. D330–D338, 2019, doi: 10.17863/CAM.36439.

[75] University of Oxford, “About the GO Lab,” 2018. http://geneontology.org/docs/introduction-to-go-resource/%0Ahttps://golab.bsg.ox.ac.uk/about/ (accessed Mar. 30, 2021).

[76] I. Pedruzzi *et al.*, “HAMAP in 2015: Updates to the protein family classification and annotation system,” *Nucleic Acids Research*, vol. 43, no. D1, pp. D1064–D1070, 2015, doi: 10.1093/nar/gku1002.

[77] A. Bateman, “UniProt: A worldwide hub of protein knowledge,” *Nucleic Acids Research*, vol. 47, no. D1, pp. D506–D515, 2019, doi: 10.1093/nar/gky1049.

[78] F. Z. Smaili, X. Gao, and R. Hoehndorf, “Onto2Vec: Joint vector-based representation of biological entities and their ontology-based annotations,” *Bioinformatics*, vol. 34, no. 13, pp. i52–i60, 2018, doi: 10.1093/bioinformatics/bty259.

[79] L. Van Der Maaten and G. Hinton, “Visualizing data using t-SNE,” *Journal of Machine Learning Research*, vol. 9, no. 86, pp. 2579–2625, 2008.

[80] P. Montravers, S. Boudinet, and H. Houissa, “Candida and severe acute pancreatitis: We won’t be fooled again,” *Critical Care*, vol. 17, no. 3, p. 137, 2013, doi: 10.1186/cc12613.

[81] CDC, “Cancer Patients and Fungal Infections | Fungal Infections | Fungal | CDC,” *Centers for Disease Control and Prevention*, 2017. https://www.cdc.gov/fungal/infections/cancer-patients.html.

[82] S. Duggan, I. Leonhardt, K. Hünniger, and O. Kurzai, “Host response to Candida albicans bloodstream infection and sepsis,” *Virulence*, vol. 6, no. 4, pp. 316–326, Jan. 2015, doi: 10.4161/21505594.2014.988096.

[83] World Health Organization, “How many TB cases and deaths are there?,” *World Health Organization*, 2015. https://www.who.int/gho/tb/epidemic/cases_deaths/en/.

[84] B. Hopkins and J. G. Skellam, “A new method for determining the type of distribution of plant individuals,” *Annals of Botany*, vol. 18, no. 2, pp. 213–227, 1954, doi: 10.1093/oxfordjournals.aob.a083391.

[85] A. Kassambara, “Assessing Clustering Tendency,” *Datanovia*, 2018. https://www.datanovia.com/en/lessons/assessing-clustering-tendency/ (accessed Apr. 12, 2021).

[86] T. Mikolov, I. Sutskever, and L. Le Quoc, “Learning the meaning behind words,” *Google Open Source Blog*, 2016. https://opensource.googleblog.com/2013/08/learning-meaning-behind-words.html (accessed Apr. 12, 2021).

Appendix I – DbO Example

```
@prefix DbO:   <http://OntoAI.XYZ.DbO/#> .
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix owl:   <http://www.w3.org/2002/07/owl#> .
@prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#> .
@prefix Property: <http://OntoAI.XYZ.Property/#> .

DbO:84e16c2ba57f28c2d2db092fe5f78e24_0.904583
        a                           DbO:EventPerformance , DbO:Person ;
        DbO:occupation              "agentOf" ;
        Property:FirstEntity        "his" ;
        Property:FirstEntityType    DbO:Person ;
        Property:Score              "0.904583" ;
        Property:SecondEntity       "inauguration" ;
        Property:SecondEntityType   DbO:EventPerformance ;
        Property:Sentence           "A number of tech leaders met with Donald
Trump before his inauguration." .

DbO:b56a6dec64292ec55dc0ef52b257628c_0.908328
        a                           DbO:Event , DbO:Organisation ;
        DbO:occupation              "agentOf" ;
        Property:FirstEntity        "firms" ;
        Property:FirstEntityType    DbO:Organisation ;
        Property:Score              "0.908328" ;
        Property:SecondEntity       "say" ;
        Property:SecondEntityType   DbO:Event ;
        Property:Sentence        "The firms say President Trump's immigration
ban \"inflicts significant harm\" on their businesses." .

DbO:c0816cda414020c964ab90e1ae444a1f
        a               owl:Ontology ;
        Property:File
"http://OntoAI.XYZ.Corpus/#c0816cda414020c964ab90e1ae444a1f" .

DbO:512284f5e10da6aa73bdc319a1e5b041_0.864576
        a                           DbO:Place , DbO:Organisation ;
        DbO:notablyAssociatedWith   "partOf" ;
        Property:FirstEntity        "firms" ;
        Property:FirstEntityType    DbO:Organisation ;
        Property:Score              "0.864576" ;
        Property:SecondEntity       "US" ;
        Property:SecondEntityType   DbO:Place ;
        Property:Sentence             "Thirty more US technology firms have
signed a brief opposing President Trump's immigration ban, bringing the total
number involved to 127." .

Property:FirstEntityType
        a              owl:AnnotationProperty ;
        rdfs:comment   "The first Entity Type in the relation."@en ;
        rdfs:label     "FirstEntityType"@en ;
        rdfs:range     xsd:string .

DbO:eventPlace  a            owl:ObjectProperty ;
        rdfs:domain          DbO:Event ;
        rdfs:isDefinedBy     DbO:c0816cda414020c964ab90e1ae444a1f ;
```

```
        rdfs:label          "eventPlace"@en ;
        rdfs:range          DbO:Place ;
        rdfs:subPropertyOf  DbO:notablyAssociatedWith .

DbO:8c3aba4c0949879e8284599f1789c1e3_0.693657
        a                           DbO:Place , DbO:Facility ;
        DbO:eventPlace              "locatedAt" ;
        Property:FirstEntity        "Home" ;
        Property:FirstEntityType    DbO:Facility ;
        Property:Score              "0.693657" ;
        Property:SecondEntity       "World" ;
        Property:SecondEntityType   DbO:Place ;
        Property:Sentence           "World Home" .

DbO:455488c7d0a7a1a704b6975328eddb08_0.897337
        a                           DbO:Event , DbO:Place ;
        DbO:occupation              "agentOf" ;
        Property:FirstEntity        "States" ;
        Property:FirstEntityType    DbO:Place ;
        Property:Score              "0.897337" ;
        Property:SecondEntity       "urge" ;
        Property:SecondEntityType   DbO:Event ;
        Property:Sentence           "Trump travel ban: States urge retention
of temporary block        6 February 2017" .

DbO:6601b8c5570ca2f7600d983b20319edc_0.832511
        a                           DbO:Place , DbO:Person ;
        DbO:notablyAssociatedWith   "partOfMany" ;
        Property:FirstEntity        "refugees" ;
        Property:FirstEntityType    DbO:Person ;
        Property:Score              "0.832511" ;
        Property:SecondEntity       "Syrian" ;
        Property:SecondEntityType   DbO:Place ;
        Property:Sentence             "Mr Trump's executive order halted the
entire US refugee programme for 120 days, indefinitely banned Syrian refugees
and suspended permission to enter the US for all nationals from seven Muslim-
majority countries." .

Property:FirstEntity  a  owl:AnnotationProperty ;
        rdfs:comment  "The first Entity in the relation."@en ;
        rdfs:label    "FirstEntity"@en ;
        rdfs:range    xsd:string .

DbO:48eb488fe1fa13922eec2a1f3b6862a7_0.623791
        a                           DbO:Event , DbO:Person ;
        DbO:eventTheme              "affectedBy" ;
        Property:FirstEntity        "leaders" ;
        Property:FirstEntityType    DbO:Person ;
        Property:Score              "0.623791" ;
        Property:SecondEntity       "invites" ;
        Property:SecondEntityType   DbO:Event ;
        Property:Sentence             "Trump invites top tech leaders to NYC
14 December 2016" .

DbO:c51364e390ed7cb449df6913579f4b41_0.816618
        a                           DbO:Place , DbO:Person ;
        DbO:eventPlace              "locatedAt" ;
        Property:FirstEntity        "Adele" ;
        Property:FirstEntityType    DbO:Person ;
```

```
        Property:Score             "0.816618" ;
        Property:SecondEntity      "Grammys" ;
        Property:SecondEntityType  DbO:Place ;
        Property:Sentence           "Who did Adele perform a tribute to at
the Grammys?" .

Property:SecondEntityType
        a             owl:AnnotationProperty ;
        rdfs:comment  "The second Entity Type in the relation."@en ;
        rdfs:label    "SecondEntityType"@en ;
        rdfs:range    xsd:string .

DbO:60211482cf44840796d9240f58e0c8b9_0.741121
        a                          DbO:Place , DbO:Person ;
        DbO:eventPlace             "locatedAt" ;
        Property:FirstEntity       "holders" ;
        Property:FirstEntityType   DbO:Person ;
        Property:Score             "0.741121" ;
        Property:SecondEntity      "Iraq" ;
        Property:SecondEntityType  DbO:Place ;
        Property:Sentence         "This means visa holders from Iraq, Syria,
Iran, Libya, Somalia, Sudan and Yemen will be allowed to enter the US until
the full case has been heard." .

DbO:6601b8c5570ca2f7600d983b20319edc_0.449823
        a                          DbO:Place , DbO:Person ;
        DbO:eventOrganisation      "employedBy" ;
        Property:FirstEntity       "nationals" ;
        Property:FirstEntityType   DbO:Person ;
        Property:Score             "0.449823" ;
        Property:SecondEntity      "countries" ;
        Property:SecondEntityType  DbO:Place ;
        Property:Sentence           "Mr Trump's executive order halted the
entire US refugee programme for 120 days, indefinitely banned Syrian refugees
and suspended permission to enter the US for all nationals from seven Muslim-
majority countries." .

DbO:610516571f1b90e4c817af52cd396c1b_0.35456
        a                          DbO:Person ;
        DbO:parentOf               "parentOf" ;
        Property:FirstEntity       "dad" ;
        Property:FirstEntityType   DbO:Person ;
        Property:Score             "0.35456" ;
        Property:SecondEntity      "my" ;
        Property:SecondEntityType  DbO:Person ;
        Property:Sentence           "Fighting to keep my dad's name off my
marriage certificate" .

DbO:EventPerformance  a   owl:Class ;
        rdfs:isDefinedBy  DbO:c0816cda414020c964ab90e1ae444a1f ;
        rdfs:label        "EventPerformance"@en .

DbO:0a78572c78ffbc67d3bf6db20b94bab4_0.869332
        a                          DbO:Organisation ;
        DbO:notablyAssociatedWith  "partOf" ;
        Property:FirstEntity       "Earth  Travel  Capital  iPlayer  Culture
Autos  Future  TV  Radio  CBBC  CBeebies  Food  iWonder  Bitesize  Travel
Music  Earth  Arts  Make It Digital  Taster  Nature  Local" ;
        Property:FirstEntityType   DbO:Organisation ;
```

```
        Property:Score             "0.869332" ;
        Property:SecondEntity      "BBC  News  News  Sport  Weather  Shop" ;
        Property:SecondEntityType  DbO:Organisation ;
        Property:Sentence          "Explore the BBC     News    News    Sport
Weather    Shop    Earth    Travel    Capital    iPlayer    Culture    Autos
Future     TV     Radio     CBBC     CBeebies     Food     iWonder     Bitesize
Travel     Music     Earth     Arts     Make It Digital     Taster     Nature
Local" .

DbO:notablyAssociatedWith
        a                  owl:ObjectProperty ;
        rdfs:isDefinedBy   DbO:c0816cda414020c964ab90e1ae444a1f ;
        rdfs:label         "notablyAssociatedWith"@en .

DbO:84e16c2ba57f28c2d2db092fe5f78e24_0.790662
        a                          DbO:Event , DbO:Person ;
        DbO:eventPlace             "participantIn" ;
        Property:FirstEntity       "Donald Trump" ;
        Property:FirstEntityType   DbO:Person ;
        Property:Score             "0.790662" ;
        Property:SecondEntity      "met" ;
        Property:SecondEntityType  DbO:Event ;
        Property:Sentence          "A number of tech leaders met with Donald
Trump before his inauguration." .

DbO:0195f1cf75b09484fc146c6d2b00a5b7_0.239713
        a                          DbO:Organisation , DbO:Person ;
        DbO:eventOrganisation      "employedBy" ;
        Property:FirstEntity       "their" ;
        Property:FirstEntityType   DbO:Person ;
        Property:Score             "0.239713" ;
        Property:SecondEntity      "businesses" ;
        Property:SecondEntityType  DbO:Organisation ;
        Property:Sentence            "They join 97 others who have filed a
legal  document  stating  the  ban  \"inflicts  significant  harm\"  on  their
businesses and is unconstitutional." .

DbO:84e16c2ba57f28c2d2db092fe5f78e24_0.935118
        a                          DbO:Event , DbO:Person ;
        DbO:eventPlace             "participantIn" ;
        Property:FirstEntity       "leaders" ;
        Property:FirstEntityType   DbO:Person ;
        Property:Score             "0.935118" ;
        Property:SecondEntity      "met" ;
        Property:SecondEntityType  DbO:Event ;
        Property:Sentence          "A number of tech leaders met with Donald
Trump before his inauguration." .

DbO:Place  a               owl:Class ;
        rdfs:isDefinedBy   DbO:c0816cda414020c964ab90e1ae444a1f ;
        rdfs:label         "Place"@en .

DbO:Organisation  a        owl:Class ;
        rdfs:isDefinedBy   DbO:c0816cda414020c964ab90e1ae444a1f ;
        rdfs:label         "Organisation"@en .

DbO:6ed34fab75c5291f602caa5832f3aba1_0.913699
        a                          DbO:Event , DbO:Person ;
        DbO:occupation             "agentOf" ;
```

```
        Property:FirstEntity       "president" ;
        Property:FirstEntityType   DbO:Person ;
        Property:Score             "0.913699" ;
        Property:SecondEntity      "said" ;
        Property:SecondEntityType  DbO:Event ;
        Property:Sentence          "However, the president has said he will
fight the order as it puts national security at risk." .

Property:SecondEntity
        a             owl:AnnotationProperty ;
        rdfs:comment  "The second Entity in the relation."@en ;
        rdfs:label    "SecondEntity"@en ;
        rdfs:range    xsd:string .

Property:File  a      owl:AnnotationProperty ;
        rdfs:comment  "The file in the Corpus Vault."@en ;
        rdfs:label    "File"@en ;
        rdfs:range    xsd:string .

DbO:270b6dc80e8ec1064fc336ca745e1661_0.587636
        a                          DbO:Organisation ;
        DbO:notablyAssociatedWith  "partOf" ;
        Property:FirstEntity       "BBC" ;
        Property:FirstEntityType   DbO:Organisation ;
        Property:Score             "0.587636" ;
        Property:SecondEntity      "Copyright 漏" ;
        Property:SecondEntityType  DbO:Organisation ;
        Property:Sentence          "Copyright 漏 2017 BBC." .

DbO:b56a6dec64292ec55dc0ef52b257628c_0.440315
        a                          DbO:Event , DbO:Person ;
        DbO:eventTheme             "affectedBy" ;
        Property:FirstEntity       "President" ;
        Property:FirstEntityType   DbO:Person ;
        Property:Score             "0.440315" ;
        Property:SecondEntity      "say" ;
        Property:SecondEntityType  DbO:Event ;
        Property:Sentence       "The firms say President Trump's immigration
ban \"inflicts significant harm\" on their businesses." .

Property:Score  a     owl:AnnotationProperty ;
        rdfs:comment  "Confidence score between 0.0 and 1.0. The higher the
score, the greater the confidence."@en ;
        rdfs:label    "Score"@en ;
        rdfs:range    xsd:double .

DbO:eventOrganisation
        a                   owl:ObjectProperty ;
        rdfs:domain         DbO:Event ;
        rdfs:isDefinedBy    DbO:c0816cda414020c964ab90e1ae444a1f ;
        rdfs:label          "eventOrganisation"@en ;
        rdfs:range          DbO:Organisation ;
        rdfs:subPropertyOf  DbO:notablyAssociatedWith .

DbO:53fdda80d4b5480d53567b153efe6b8d_0.619426
        a                          DbO:Place , DbO:Person ;
        DbO:eventPlace             "residesIn" ;
        Property:FirstEntity       "federal judge" ;
        Property:FirstEntityType   DbO:Person ;
```

```
        Property:Score             "0.619426" ;
        Property:SecondEntity      "Washington" ;
        Property:SecondEntityType  DbO:Place ;
        Property:Sentence         "There is currently a nationwide temporary
restraining order in place, which was issued on Friday by a federal judge in
Washington." .

DbO:1b6e0104cc3e8765c0348553272a7e9d_0.527797
        a                          DbO:Person ;
        DbO:notablyAssociatedWith  "partOfMany" ;
        Property:FirstEntity       "Blair" ;
        Property:FirstEntityType   DbO:Person ;
        Property:Score             "0.527797" ;
        Property:SecondEntity      "their" ;
        Property:SecondEntityType  DbO:Person ;
        Property:Sentence           "Blair: Time to rise up against Brexit
The ex-prime minister says leaving the EU is \"not inevitable\" and Britons
could change their minds." .

DbO:b3369c198b218292ce67d80030798b44_0.377566
        a                          DbO:Organisation , DbO:Person ;
        DbO:eventOrganisation      "employedBy" ;
        Property:FirstEntity       "signatories" ;
        Property:FirstEntityType   DbO:Person ;
        Property:Score             "0.377566" ;
        Property:SecondEntity      "Microsoft" ;
        Property:SecondEntityType  DbO:Organisation ;
        Property:Sentence          "It was filed in Washington on Sunday and
also includes Apple, Facebook and Microsoft as signatories." .

DbO:occupation  a           owl:ObjectProperty ;
        rdfs:domain         DbO:Person ;
        rdfs:isDefinedBy    DbO:c0816cda414020c964ab90e1ae444a1f ;
        rdfs:label          "occupation"@en ;
        rdfs:subPropertyOf  DbO:notablyAssociatedWith .

DbO:Event  a              owl:Class ;
        rdfs:isDefinedBy  DbO:c0816cda414020c964ab90e1ae444a1f ;
        rdfs:label        "Event"@en .

DbO:parentOf  a           owl:AnnotationProperty ;
        rdfs:isDefinedBy  DbO:c0816cda414020c964ab90e1ae444a1f ;
        rdfs:label        "parentOf"@en .

Property:Sentence  a  owl:AnnotationProperty ;
        rdfs:comment  "This is the original text from the document where the
relation was detected."@en ;
        rdfs:label    "Sentence"@en ;
        rdfs:range    xsd:string .

DbO:bfc404453132521ee586978cf37bac99_0.497183
        a                          DbO:Event , DbO:Organisation ;
        DbO:eventTheme             "affectedBy" ;
        Property:FirstEntity       "Amazon" ;
        Property:FirstEntityType   DbO:Organisation ;
        Property:Score             "0.497183" ;
        Property:SecondEntity      "lawsuit" ;
        Property:SecondEntityType  DbO:Event ;
```

```
        Property:Sentence           "Amazon is not part of the amicus brief
but it is a witness in the original lawsuit brought by the Washington state
Attorney General." .

DbO:1b6e0104cc3e8765c0348553272a7e9d_0.534279
        a                           DbO:Event , DbO:Person ;
        DbO:occupation              "agentOf" ;
        Property:FirstEntity        "Blair" ;
        Property:FirstEntityType    DbO:Person ;
        Property:Score              "0.534279" ;
        Property:SecondEntity       "says" ;
        Property:SecondEntityType   DbO:Event ;
        Property:Sentence            "Blair: Time to rise up against Brexit
The ex-prime minister says leaving the EU is \"not inevitable\" and Britons
could change their minds." .

DbO:Facility  a           owl:Class ;
        rdfs:isDefinedBy  DbO:c0816cda414020c964ab90e1ae444a1f ;
        rdfs:label        "Facility"@en .

DbO:Person  a             owl:Class ;
        rdfs:isDefinedBy  DbO:c0816cda414020c964ab90e1ae444a1f ;
        rdfs:label        "Person"@en .

DbO:bfc404453132521ee586978cf37bac99_0.86232
        a                           DbO:Event , DbO:Person ;
        DbO:eventTheme              "affectedBy" ;
        Property:FirstEntity        "witness" ;
        Property:FirstEntityType    DbO:Person ;
        Property:Score              "0.86232" ;
        Property:SecondEntity       "lawsuit" ;
        Property:SecondEntityType   DbO:Event ;
        Property:Sentence           "Amazon is not part of the amicus brief
but it is a witness in the original lawsuit brought by the Washington state
Attorney General." .

DbO:28d913cbe19ff78908f424f5607b4e20_0.95153
        a                           DbO:Event , DbO:Organisation ;
        DbO:occupation              "agentOf" ;
        Property:FirstEntity        "firms" ;
        Property:FirstEntityType    DbO:Organisation ;
        Property:Score              "0.95153" ;
        Property:SecondEntity       "oppose" ;
        Property:SecondEntityType   DbO:Event ;
        Property:Sentence          "Thirty more tech firms oppose Trump ban" .

DbO:48eb488fe1fa13922eec2a1f3b6862a7_0.996752
        a                           DbO:Event , DbO:Person ;
        DbO:occupation              "agentOf" ;
        Property:FirstEntity        "Trump" ;
        Property:FirstEntityType    DbO:Person ;
        Property:Score              "0.996752" ;
        Property:SecondEntity       "invites" ;
        Property:SecondEntityType   DbO:Event ;
        Property:Sentence            "Trump invites top tech leaders to NYC
14 December 2016" .

DbO:eventTheme  a           owl:ObjectProperty ;
        rdfs:domain         DbO:Event ;
```

```
        rdfs:isDefinedBy    DbO:c0816cda414020c964ab90e1ae444a1f ;
        rdfs:label          "eventTheme"@en ;
        rdfs:subPropertyOf  DbO:notablyAssociatedWith .

DbO:bfc404453132521ee586978cf37bac99_0.767119
        a                           DbO:Place , DbO:Person ;
        DbO:eventOrganisation       "employedBy" ;
        Property:FirstEntity        "Attorney General" ;
        Property:FirstEntityType    DbO:Person ;
        Property:Score              "0.767119" ;
        Property:SecondEntity       "Washington state" ;
        Property:SecondEntityType   DbO:Place ;
        Property:Sentence           "Amazon is not part of the amicus brief
but it is a witness in the original lawsuit brought by the Washington state
Attorney General." .

DbO:03f767de583928b9154a5f9ac6a4cccc_0.834963
        a                           DbO:Person ;
        DbO:parentOf                "parentOf" ;
        Property:FirstEntity        "father" ;
        Property:FirstEntityType    DbO:Person ;
        Property:Score              "0.834963" ;
        Property:SecondEntity       "my" ;
        Property:SecondEntityType   DbO:Person ;
        Property:Sentence              "How I escaped my father's murderous
cult" .
```

Appendix II -- Full Connectivity Coefficient Result in Experiment One

X var	Y var	MIC (strength)	MIC-p^2 (nonlinearity)	Linear regression(p)
Cardinal	Date	0.3743100	-0.2562962	0.7941071
Cardinal	Facility	0.3560700	-0.2531488	0.7805247
Date	Facility	0.3285000	-0.2290696	0.7467058
EventViolence	Facility	0.3066100	-0.2274811	0.7308154
EventViolence	Cardinal	0.3009300	-0.2322719	0.7302067
EventElection	Cardinal	0.2841700	-0.2247647	0.7133966
Crime	Cardinal	0.2804400	-0.1578619	0.6620438
EventViolence	Date	0.2778800	-0.2092887	0.6979747
EventViolence	Crime	0.2747000	-0.1746272	0.6703187
Organisation	Place	0.2679800	-0.1895509	0.6764103
Cardinal	EventPersonnel	0.2656400	-0.1943409	0.6782189
EventElection	EventPersonnel	0.2534900	-0.1972711	0.6713874
Crime	Date	0.2522600	-0.1432052	0.6288602
EventElection	Date	0.2509800	-0.2145229	0.6822777
Crime	Facility	0.2501500	-0.1325292	0.6186107
EventElection	Facility	0.2446000	-0.1941173	0.6623573
EventPersonnel	Date	0.2442000	-0.2080491	0.6724947
EventPersonnel	Facility	0.2428200	-0.1802339	0.6504259
TitleWork	Facility	0.2362300	-0.1290639	0.6043955
EventPerformance	EventPersonnel	0.2238600	-0.1521745	0.6132165
TitleWork	EventViolence	0.2226100	-0.1167241	0.5825239
TitleWork	Cardinal	0.2203400	-0.1212543	0.5844607
EventPerformance	Date	0.2183100	-0.1646571	0.6188434
Award	Date	0.2176600	-0.1640686	0.6178419
Award	EventElection	0.2157000	-0.1806961	0.6295999
Award	Cardinal	0.2117000	-0.1680913	0.6162721
Vehicle	Cardinal	0.1980200	-0.1362945	0.5781994
TitleWork	Crime	0.1949000	-0.0917405	0.5353882
Organisation	Person	0.1936600	-0.1529523	0.5887379
Crime	EventPersonnel	0.1936000	-0.1140839	0.5546926
EventPerformance	Cardinal	0.1894400	-0.1465963	0.5796863
Vehicle	EventElection	0.1891800	-0.1280119	0.5631979
Vehicle	Date	0.1882300	-0.1168483	0.5523389
TitleWork	Weapon	0.1875900	-0.0666545	0.5042266
Vehicle	EventPersonnel	0.1867900	-0.1106533	0.5453836
EventViolence	EventElection	0.1864400	-0.1469524	0.5774015
EventElection	Crime	0.1851800	-0.0980068	0.5321530

Award	Facility	0.1830300	-0.1395844	0.5679915
Award	EventPersonnel	0.1814400	-0.1335640	0.5612522
TitleWork	Date	0.1814000	-0.0740140	0.5053850
EventPerformance	EventElection	0.1809800	-0.1359310	0.5629485
EventPerformance	Facility	0.1809600	-0.1099176	0.5393307
EventPerformance	Vehicle	0.1804100	-0.0986306	0.5282430
Weapon	Facility	0.1762100	-0.0548935	0.4807323
EventViolence	EventPersonnel	0.1757300	-0.1315371	0.5543168
Person	Place	0.1751300	-0.0963399	0.5210277
Event	Organisation	0.1750100	-0.1642934	0.5824975
Event	Place	0.1747000	-0.1338986	0.5555165
Award	EventPerformance	0.1727000	-0.1141162	0.5355523
EventCustody	Facility	0.1726400	-0.0501232	0.4719780
Vehicle	Facility	0.1721700	-0.0920177	0.5139919
Vehicle	Crime	0.1706200	-0.0605974	0.4808507
Substance	Cardinal	0.1698500	-0.0630235	0.4825697
EventCustody	Crime	0.1686400	-0.0468031	0.4641585
Vehicle	EventViolence	0.1628000	-0.0981364	0.5108193
TitleWork	EventPersonnel	0.1621800	-0.0541443	0.4651068
Event	Person	0.1598400	-0.1263142	0.5349339
EventCustody	Cardinal	0.1593100	-0.0338172	0.4394625
Award	Vehicle	0.1579100	-0.0994069	0.5072641
Substance	EventElection	0.1569700	-0.0442146	0.4485360
EventViolence	Weapon	0.1566500	-0.0501439	0.4547460
Award	EventViolence	0.1564500	-0.1084293	0.5146642
HealthCondition	Date	0.1548900	-0.0246047	0.4236681
Substance	Facility	0.1542100	-0.0370777	0.4373646
Award	Crime	0.1536600	-0.0651671	0.4677896
EventCustody	EventPersonnel	0.1531900	-0.0243805	0.4213911
Crime	Weapon	0.1483500	-0.0318172	0.4244610
TitleWork	EventElection	0.1460100	-0.0387884	0.4298819
EventCustody	EventViolence	0.1449600	-0.0203259	0.4065537
TitleWork	Substance	0.1414500	-0.0079059	0.3864659
Cardinal	Organisation	0.1413000	-0.1187399	0.5099411
Substance	Vehicle	0.1409700	-0.0287979	0.4120290
Date	Person	0.1408000	-0.0861908	0.4764355
EventPerformance	Crime	0.1403200	-0.0372911	0.4214393
Cardinal	Weapon	0.1402600	-0.0206243	0.4011039
Substance	EventViolence	0.1402600	-0.0316350	0.4146022
Cardinal	Place	0.1402000	-0.0958148	0.4858136
Award	HealthCondition	0.1386600	-0.0048271	0.3787970
EventCustody	Date	0.1384400	-0.0128194	0.3889208
EventPerformance	EventViolence	0.1381500	-0.0624378	0.4478703
Cardinal	Person	0.1379400	-0.0884291	0.4757826

Substance	Date	0.1379300	-0.0351449	0.4160227
TitleWork	EventCustody	0.1376000	-0.0227550	0.4004435
EventCustody	EventElection	0.1365600	-0.0176083	0.3926427
Date	Event	0.1364700	-0.1261644	0.5124787
Time	Date	0.1364400	-0.0216431	0.3975966
HealthCondition	Cardinal	0.1357200	-0.0132168	0.3859234
Facility	Place	0.1351100	-0.0948086	0.4794982
EventPerformance	HealthCondition	0.1350100	-0.0139261	0.3859224
HealthCondition	Crime	0.1349700	-0.0023254	0.3705340
HealthCondition	EventViolence	0.1341400	0.0016371	0.3640095
Facility	Organisation	0.1337200	-0.0924777	0.4756025
Time	Cardinal	0.1335800	-0.0120157	0.3815700
Award	Substance	0.1326400	-0.0309776	0.4044967
EventPerformance	Time	0.1318700	-0.0135485	0.3813378
Substance	EventPersonnel	0.1317600	-0.0274506	0.3990120
Date	Organisation	0.1309700	-0.1105856	0.4914831
EventElection	Organisation	0.1305800	-0.0803443	0.4592649
EventPersonnel	Event	0.1304800	-0.0857375	0.4649919
Date	Weapon	0.1271700	-0.0106557	0.3712489
Substance	Crime	0.1264100	-0.0122058	0.3723115
Cardinal	Event	0.1262200	-0.1102677	0.4863001
Date	Place	0.1258800	-0.0832004	0.4572531
Time	Facility	0.1250000	0.0020991	0.3505723
Award	Organisation	0.1249900	-0.0758321	0.4481318
HealthCondition	EventPersonnel	0.1243300	0.0047895	0.3457463
EventCustody	Weapon	0.1241800	0.0049166	0.3453454
HealthCondition	EventElection	0.1237300	0.0108999	0.3359019
EventPerformance	Event	0.1235600	-0.0710292	0.4411227
EventPerformance	SportingEvent	0.1232900	-0.0051946	0.3584476
EventElection	Place	0.1231100	-0.0621313	0.4303967
Time	EventPersonnel	0.1230300	-0.0006458	0.3516757
Award	Place	0.1228200	-0.0448698	0.4094994
EventBusiness	Facility	0.1221900	-0.0036251	0.3547043
EventBusiness	EventElection	0.1218800	0.0168034	0.3241553
Award	Person	0.1211500	-0.0370967	0.3978023
EventElection	Event	0.1206200	-0.0909390	0.4599554
HealthCondition	Facility	0.1204600	0.0023419	0.3436832
Facility	Person	0.1196500	-0.0625268	0.4268217
TitleWork	GeographicFeature	0.1195200	0.0175130	0.3193854
EventPersonnel	Organisation	0.1179000	-0.0640710	0.4265806
Award	Event	0.1178900	-0.0762531	0.4406167
EventElection	Person	0.1176700	-0.0488285	0.4080423
TitleWork	HealthCondition	0.1173700	0.0183124	0.3147342
EventViolence	Organisation	0.1161500	-0.0728985	0.4347971

Time	EventElection	0.1151400	0.0018023	0.3366566
EventPersonnel	Place	0.1148000	-0.0520991	0.4085329
TitleWork	EventPerformance	0.1148000	-0.0044647	0.3453472
EventViolence	GeographicFeature	0.1146000	0.0254488	0.2985820
TitleWork	Vehicle	0.1143000	-0.0091466	0.3513497
EventCustody	EventPerformance	0.1139500	0.0195035	0.3073215
Award	Time	0.1137600	0.0213070	0.3040608
TitleWork	Organisation	0.1132900	0.0039693	0.3306368
Award	TitleWork	0.1128800	-0.0103249	0.3510055
EventViolence	Person	0.1128200	-0.0411038	0.3923312
EventBusiness	EventEducation	0.1125400	0.0241060	-0.2973786
EventPerformance	Person	0.1123500	-0.0245194	0.3699586
EventBusiness	Cardinal	0.1121000	0.0116750	0.3168991
EventElection	Weapon	0.1108700	0.0014910	0.3307250
EventViolence	Place	0.1106200	-0.0527633	0.4042070
TitleWork	Person	0.1105400	0.0008754	0.3311565
EventBusiness	EventPersonnel	0.1097800	0.0265618	0.2884757
EventBusiness	EventViolence	0.1097000	0.0129411	0.3110610
Crime	Organisation	0.1093300	-0.0239722	0.3651058
EventBusiness	TitleWork	0.1092700	0.0275979	0.2857833
Time	Crime	0.1086300	0.0202111	0.2973532
EventPersonnel	Person	0.1082400	-0.0233118	0.3627008
Duration	EventDemonstration	0.1072200	0.0298985	0.2780675
Vehicle	Weapon	0.1063400	0.0289821	0.2781328
Facility	Event	0.1057800	-0.0691304	0.4182229
SportingEvent	Facility	0.1056900	0.0248412	0.2843392
Award	EventCustody	0.1055300	0.0206116	0.2914076
Substance	Person	0.1054300	0.0076702	0.3126656
TitleWork	Place	0.1046900	0.0072178	0.3122053
Vehicle	SportingEvent	0.1044400	0.0169003	0.2958712
Substance	Organisation	0.1041600	0.0029632	0.3181144
Substance	Place	0.1037900	0.0224508	0.2852003
EventPerformance	Organisation	0.1033200	-0.0275532	0.3617641
EventPersonnel	Weapon	0.1025500	0.0307364	0.2679807
EventBusiness	GeographicFeature	0.1024400	0.0246930	0.2788316
Cardinal	SportingEvent	0.1023800	0.0208196	0.2855879
Time	Vehicle	0.1019800	0.0310828	0.2662652
EventElection	SportingEvent	0.1018500	0.0278441	0.2720403
EventPerformance	Place	0.1017900	-0.0090831	0.3329761
Time	EventViolence	0.1016200	0.0117120	0.2998467
EventBusiness	Date	0.1015800	0.0232534	0.2798689
Vehicle	HealthCondition	0.1012600	0.0183993	0.2878553
Time	Organisation	0.1009000	0.0430486	0.2405232
Substance	Event	0.1008900	0.0174686	0.2888276

Date	SportingEvent	0.1007800	0.0178928	0.2879013
Award	SportingEvent	0.1004400	0.0397110	0.2464325
Substance	Weapon	0.1003100	0.0339958	0.2575155
Product	Event	0.0996300	0.0350219	-0.2541812
EventCustody	HealthCondition	0.0995700	0.0301482	0.2634802
Time	Event	0.0989800	0.0280949	0.2662425
Vehicle	Event	0.0975700	-0.0153711	0.3360671
Crime	Place	0.0973700	0.0041260	0.3053588
Crime	Person	0.0972100	0.0090287	0.2969533
NaturalDisaster	Crime	0.0968900	0.0445125	0.2288614
HealthCondition	Person	0.0963300	0.0489262	0.2177242
EventViolence	Event	0.0959900	-0.0176702	0.3371352
EventBusiness	Award	0.0958900	0.0418037	0.2325647
Vehicle	Person	0.0954600	0.0057884	0.2994521
Weapon	GeographicFeature	0.0951500	0.0532623	0.2046648
Product	Award	0.0950800	0.0313482	-0.2524516
Award	Weapon	0.0950700	0.0344081	0.2462963
SportingEvent	Event	0.0943800	0.0442090	0.2239888
EventBusiness	Crime	0.0942400	0.0375923	0.2380077
TitleWork	Event	0.0942200	0.0362848	0.2406973
Substance	EventPerformance	0.0937000	0.0405583	0.2305248
Crime	GeographicFeature	0.0933600	0.0337880	0.2440738
EventDemonstration	Vehicle	0.0932700	0.0418945	0.2266615
Crime	Event	0.0931100	-0.0004167	0.3058213
EventCustody	NaturalDisaster	0.0931000	0.0480481	0.2122543
EventCustody	Vehicle	0.0931000	0.0362010	0.2385351
GeographicFeature	Facility	0.0927100	0.0447252	0.2190544
Product	EventElection	0.0926900	0.0280505	-0.2542431
EventCustody	Organisation	0.0919400	0.0427391	0.2218128
EventPersonnel	SportingEvent	0.0916800	0.0323753	0.2435256
Vehicle	Organisation	0.0916000	0.0024899	0.2985132
Time	Place	0.0914900	0.0400437	0.2268179
Duration	EventPersonnel	0.0913700	0.0433507	0.2191332
NaturalDisaster	GeographicFeature	0.0911600	0.0429517	0.2195639
Vehicle	Place	0.0907100	-0.0009735	0.3027928
EventBusiness	EventCustody	0.0905200	0.0467713	0.2091620
Time	HealthCondition	0.0902100	0.0469267	0.2080463
TitleWork	SportingEvent	0.0901400	0.0409357	0.2218205
Duration	Vehicle	0.0898500	0.0508749	0.1974211
EventCustody	Place	0.0897800	0.0397072	0.2237695
EventViolence	SportingEvent	0.0891900	0.0421810	0.2168155
Product	Person	0.0891200	0.0470336	-0.2051497
EntertainmentAward	HealthCondition	0.0889500	0.0470378	0.2047248
Substance	Time	0.0889400	0.0602114	0.1694953

Product	EventPerformance	0.0889400	0.0425274	-0.2154359
EventBusiness	EventPerformance	0.0888700	0.0413519	0.2179864
TitleWork	Time	0.0888300	0.0487154	0.2002863
NaturalDisaster	Facility	0.0887400	0.0527227	0.1897822
EventBusiness	Place	0.0887000	0.0351346	0.2314419
EventBusiness	Person	0.0883700	0.0435645	0.2116730
EntertainmentAward	EventCustody	0.0882500	0.0542194	0.1844739
EventDemonstration	EventPersonnel	0.0882400	0.0449378	0.2080919
EventBusiness	Organisation	0.0880600	0.0404609	0.2181723
Substance	HealthCondition	0.0878900	0.0462242	0.2041219
HealthCondition	Organisation	0.0877800	0.0503409	0.1934919
Time	Person	0.0876100	0.0470018	0.2015147
EventCustody	Event	0.0873800	0.0484145	0.1973967
Time	SportingEvent	0.0873500	0.0498484	0.1936533
EventCustody	GeographicFeature	0.0871800	0.0444634	0.2066799
EventEducation	GeographicFeature	0.0871600	0.0473161	-0.1996094
EntertainmentAward	SportingEvent	0.0870000	0.0438316	0.2077702
Duration	EventElection	0.0868600	0.0468366	0.2000584
Substance	EventCustody	0.0867500	0.0479657	0.1969373
Weapon	Place	0.0864200	0.0419868	0.2107917
Product	Date	0.0863200	0.0469384	-0.1984479
Duration	Date	0.0862800	0.0581920	0.1675948
HealthCondition	Place	0.0861200	0.0505962	0.1884776
SportingEvent	Weapon	0.0859500	0.0457196	0.2005753
NaturalDisaster	Cardinal	0.0859000	0.0623742	0.1533812
Duration	HealthCondition	0.0858100	0.0585735	0.1650349
EntertainmentAward	Vehicle	0.0857500	0.0553934	0.1742313
TitleWork	EntertainmentAward	0.0857200	0.0487820	0.1921925
Vehicle	NaturalDisaster	0.0855200	0.0559791	0.1718748
EventDemonstration	Crime	0.0852800	0.0480937	0.1928376
EventBusiness	EntertainmentAward	0.0850700	0.0549419	0.1735746
EventBusiness	HealthCondition	0.0849800	0.0480844	0.1920823
EventViolence	NaturalDisaster	0.0849300	0.0623886	0.1501381
HealthCondition	Event	0.0848900	0.0615933	0.1526326
EntertainmentAward	Weapon	0.0838600	0.0545248	0.1712751
Weapon	Person	0.0838000	0.0522038	0.1777533
Crime	SportingEvent	0.0838000	0.0552424	0.1689900
EntertainmentAward	EventElection	0.0834900	0.0643322	0.1384119
EventBusiness	Vehicle	0.0834300	0.0493049	0.1847297
Product	SportingEvent	0.0832500	0.0531981	-0.1733549
Weapon	Organisation	0.0832200	0.0397395	0.2085197
Cardinal	GeographicFeature	0.0830600	0.0507962	0.1796212
Product	Organisation	0.0830300	0.0613568	-0.1472182
Weapon	Event	0.0830200	0.0702378	0.1130583

Product	EventPersonnel	0.0826300	0.0507965	-0.1784195
EntertainmentAward	Crime	0.0824700	0.0557873	0.1633484
Duration	EventPerformance	0.0824500	0.0530846	0.1713632
Product	NaturalDisaster	0.0820000	0.0532867	0.1694500
SportingEvent	Person	0.0818600	0.0582608	0.1536202
SportingEvent	Place	0.0818300	0.0597875	0.1484673
NaturalDisaster	EventPersonnel	0.0816100	0.0607133	0.1445570
EventCustody	Person	0.0815900	0.0556939	0.1609226
EventCustody	SportingEvent	0.0815100	0.0560233	0.1596455
EventBusiness	Weapon	0.0811100	0.0509879	0.1735573
Product	Vehicle	0.0807500	0.0592165	-0.1467431
Product	GeographicFeature	0.0807100	0.0553885	0.1591274
EventEducation	EventViolence	0.0806600	0.0605218	-0.1419092
Product	Cardinal	0.0805100	0.0530243	-0.1657880
SportingEvent	Organisation	0.0802700	0.0642144	0.1267107
EventDemonstration	Cardinal	0.0800500	0.0589699	0.1451900
Date	GeographicFeature	0.0798400	0.0628065	0.1305126
Duration	EventCustody	0.0796000	0.0528612	0.1635201
EventDemonstration	Facility	0.0792800	0.0525003	0.1636450
Duration	Crime	0.0790300	0.0510535	0.1672618
Award	EntertainmentAward	0.0789800	0.0630337	0.1262786
Product	Substance	0.0789700	0.0549611	-0.1549481
EntertainmentAward	Cardinal	0.0789200	0.0623142	0.1288636
Time	Weapon	0.0787300	0.0596588	0.1380987
EventBusiness	SportingEvent	0.0785500	0.0552515	0.1526386
EventCustody	EventDemonstration	0.0785400	0.0536371	0.1578064
EventPerformance	Weapon	0.0784600	0.0570315	0.1463848
Duration	NaturalDisaster	0.0784400	0.0625197	0.1261758
EntertainmentAward	NaturalDisaster	0.0783500	0.0572838	0.1451419
Product	Time	0.0781700	0.0621507	-0.1265673
Product	Facility	0.0780100	0.0734730	-0.0673574
EventDemonstration	EventElection	0.0774100	0.0582532	0.1384080
EventEducation	Person	0.0773500	0.0619793	-0.1239785
EventBusiness	Duration	0.0773100	0.0615641	0.1254826
EntertainmentAward	GeographicFeature	0.0769200	0.0500609	0.1638875
EventBusiness	Event	0.0768300	0.0564127	0.1428890
Duration	Cardinal	0.0768200	0.0628703	0.1181090
Duration	Event	0.0767800	0.0687796	0.0894451
HealthCondition	Weapon	0.0767000	0.0552389	0.1464959
HealthCondition	SportingEvent	0.0766600	0.0571659	0.1396213
EntertainmentAward	Substance	0.0764200	0.0664528	0.0998358
GeographicFeature	Organisation	0.0763700	0.0684722	0.0888696
EventDemonstration	Date	0.0762400	0.0644742	0.1084702
EventDemonstration	SportingEvent	0.0761400	0.0659634	0.1008791

EventBusiness	NaturalDisaster	0.0761200	0.0550774	0.1450605
Product	Place	0.0757700	0.0554336	-0.1426058
NaturalDisaster	EventElection	0.0755200	0.0630431	0.1117002
EventPerformance	EventDemonstration	0.0754600	0.0622483	0.1149421
EventDemonstration	Time	0.0753900	0.0664438	0.0945846
EventEducation	Vehicle	0.0752700	0.0683029	-0.0834694
EventEducation	EntertainmentAward	0.0750400	0.0700166	-0.0708763
NaturalDisaster	Date	0.0750000	0.0717016	0.0574315
Award	EventDemonstration	0.0749300	0.0718215	-0.0557538
HealthCondition	NaturalDisaster	0.0749000	0.0677435	0.0845963
EntertainmentAward	Date	0.0747500	0.0666115	0.0902135
EntertainmentAward	EventPerformance	0.0746800	0.0673842	0.0854155
EventBusiness	Substance	0.0746700	0.0612166	0.1159888
HealthCondition	GeographicFeature	0.0746300	0.0614044	0.1150027
Duration	EventViolence	0.0746200	0.0663209	0.0910994
EventDemonstration	EventViolence	0.0741800	0.0579210	0.1275109
Product	EventViolence	0.0741800	0.0682916	-0.0767356
EntertainmentAward	Place	0.0739900	0.0739684	0.0046476
EntertainmentAward	EventViolence	0.0739600	0.0638490	0.1005535
Duration	EntertainmentAward	0.0737300	0.0613290	0.1113598
EventDemonstration	NaturalDisaster	0.0737000	0.0617788	0.1091843
Product	EventCustody	0.0736600	0.0717137	0.0441172
EventCustody	Time	0.0734500	0.0584671	0.1224048
EventPerformance	NaturalDisaster	0.0734300	0.0725662	0.0293900
EventPersonnel	GeographicFeature	0.0733500	0.0606754	0.1125815
EventEducation	EventDemonstration	0.0731900	0.0723415	-0.0291299
TitleWork	NaturalDisaster	0.0731700	0.0603570	0.1131944
EntertainmentAward	EventPersonnel	0.0731500	0.0650748	0.0898622
NaturalDisaster	Person	0.0730700	0.0720793	-0.0314753
Duration	Time	0.0730400	0.0720325	-0.0317408
Substance	GeographicFeature	0.0730200	0.0721132	-0.0301136
NaturalDisaster	Event	0.0729300	0.0724763	-0.0213007
NaturalDisaster	Weapon	0.0726300	0.0620000	0.1031017
NaturalDisaster	SportingEvent	0.0725900	0.0686669	-0.0626350
GeographicFeature	Event	0.0725400	0.0720201	-0.0228006
NaturalDisaster	Organisation	0.0724700	0.0724642	-0.0024090
Duration	Facility	0.0722000	0.0669836	0.0722245
Time	NaturalDisaster	0.0721800	0.0696318	0.0504795
GeographicFeature	Place	0.0721500	0.0617751	0.1018574
EventDemonstration	Place	0.0720900	0.0709790	0.0333323
EventElection	GeographicFeature	0.0720500	0.0686687	0.0581493
EntertainmentAward	Event	0.0718500	0.0718401	0.0031440
EventEducation	SportingEvent	0.0717900	0.0649408	0.0827599
SportingEvent	GeographicFeature	0.0716500	0.0626392	0.0949251

EventEducation	Facility	0.0716500	0.0679363	-0.0609405
Time	GeographicFeature	0.0715600	0.0715599	0.0003085
EventEducation	EventElection	0.0715200	0.0611821	-0.1016755
Duration	SportingEvent	0.0714500	0.0621004	0.0966934
EntertainmentAward	Person	0.0713400	0.0696114	0.0415762
EventBusiness	EventDemonstration	0.0712800	0.0709056	-0.0193496
EventDemonstration	Organisation	0.0712000	0.0676979	0.0591785
EventEducation	Cardinal	0.0710300	0.0673465	-0.0606922
EventEducation	Place	0.0709900	0.0692673	-0.0415055
EntertainmentAward	Organisation	0.0708900	0.0708777	-0.0035063
Product	EntertainmentAward	0.0707500	0.0688474	-0.0436187
EntertainmentAward	Facility	0.0706500	0.0627701	0.0887690
Product	HealthCondition	0.0705600	0.0666880	-0.0622257
EventEducation	Date	0.0705500	0.0703644	-0.0136240
GeographicFeature	Person	0.0705000	0.0652931	0.0721590
EntertainmentAward	EventDemonstration	0.0704900	0.0679829	0.0500712
Product	EventEducation	0.0704900	0.0671025	0.0582022
Duration	Substance	0.0704800	0.0696084	0.0295222
EventEducation	Award	0.0704500	0.0679203	-0.0502963
Duration	Award	0.0704200	0.0614815	0.0945436
Duration	GeographicFeature	0.0703700	0.0634498	0.0831877
Product	Crime	0.0703500	0.0664702	-0.0622877
EventEducation	Organisation	0.0702600	0.0667156	-0.0595352
EventDemonstration	HealthCondition	0.0702400	0.0686861	-0.0394201
EventEducation	Duration	0.0701800	0.0657584	0.0664948
Duration	Place	0.0700900	0.0695335	0.0235906
Product	Weapon	0.0699900	0.0680915	0.0435722
EventEducation	EventPersonnel	0.0698500	0.0676453	-0.0469545
Duration	Organisation	0.0698000	0.0681677	0.0404023
Award	NaturalDisaster	0.0697800	0.0697388	0.0064170
Vehicle	GeographicFeature	0.0697700	0.0631054	0.0816372
EventDemonstration	Weapon	0.0696000	0.0660522	0.0595637
Substance	SportingEvent	0.0694800	0.0606542	0.0939455
EventDemonstration	Event	0.0694600	0.0671147	0.0484283
EventEducation	TitleWork	0.0692900	0.0671071	-0.0467220
EventEducation	Substance	0.0689500	0.0685151	-0.0208547
Product	Duration	0.0688500	0.0684694	0.0195083
EventEducation	Weapon	0.0688200	0.0685658	-0.0159426
Duration	Person	0.0687200	0.0686480	-0.0084834
EventEducation	EventCustody	0.0685800	0.0682624	-0.0178222
EventDemonstration	GeographicFeature	0.0683300	0.0629765	0.0731678
Substance	NaturalDisaster	0.0682900	0.0668629	0.0377774
EventEducation	Crime	0.0682200	0.0663739	-0.0429664
Duration	Weapon	0.0682000	0.0665649	-0.0404367

EventBusiness	Time	0.0682000	0.0670120	0.0344676
EntertainmentAward	Time	0.0680900	0.0670769	-0.0318295
Award	GeographicFeature	0.0680300	0.0679281	0.0100956
NaturalDisaster	Place	0.0679900	0.0629296	0.0711369
EventEducation	Event	0.0679700	0.0677435	0.0150504
EventEducation	HealthCondition	0.0677300	0.0657426	0.0445805
Product	TitleWork	0.0676800	0.0656032	-0.0455724
EventBusiness	Product	0.0675800	0.0668908	-0.0262533
EventEducation	Time	0.0675700	0.0661329	0.0379094
EventDemonstration	Person	0.0674500	0.0670495	0.0200135
EventEducation	NaturalDisaster	0.0670500	0.0670433	0.0025802
EventPerformance	GeographicFeature	0.0668300	0.0655651	0.0355650
EventEducation	EventPerformance	0.0665700	0.0664352	-0.0116120
Product	EventDemonstration	0.0664200	0.0644246	0.0446704
TitleWork	EventDemonstration	0.0662700	0.0648278	0.0379765
Substance	EventDemonstration	0.0662000	0.0661429	-0.0075593
Duration	TitleWork	0.0640200	0.0640091	-0.0033062

Appendix III -- The CS' result of the second experiment

Concept	$\overline{CS}_{CW}$	$\overline{CS'}_{CW}$	CS	CS'
Disease_D010505	-0.0011829	0.8943114	0.0207074	0.9265831
Gene_853823	0.0006998	0.8891604	0.0293362	0.9255597
Disease_D058565	0.0013040	0.8942040	0.0951685	0.9201904
Gene_1509	0.0030898	0.8984119	0.1319365	0.9124130
Gene_855030	0.0077035	0.9156478	0.1217937	0.8874506
Disease_D003645	0.0037329	0.9009433	-0.1365934	0.8653799
Disease_D014689	0.0012572	0.8962939	-0.0907910	0.8605850
Disease_D007794	0.0059537	0.8974273	0.0340792	0.8475901
Disease_D001471	-0.0008832	0.8935587	0.0078127	0.8209251
Disease_D012480	0.0015221	0.8756122	0.0365848	0.7867964
Disease_C535390	-0.0016075	0.8954012	-0.0130027	0.7828508
Gene_7431	0.0067158	0.8877385	-0.1246973	0.7786208
Disease_D006192	0.0069363	0.9015389	0.1747824	0.7756270
Gene_1595	0.0079732	0.8874393	0.1003887	0.7480999
Gene_4155	0.0072611	0.9003241	0.0069378	0.7246535
Disease_C535342	0.0030218	0.8954569	0.1300061	0.7221987
Disease_D004405	0.0037021	0.9071112	0.0019479	0.7197927
Disease_D007640	0.0028883	0.9006598	-0.0841984	0.7059912
Disease_D012376	-0.0000348	0.9095467	0.1198208	0.6880114
Disease_D010211	0.0121139	0.8919884	0.2602795	0.6745116
Gene_4589	0.0019788	0.9081783	-0.0120068	0.6655252
Disease_D008228	-0.0009892	0.8750802	-0.0272254	0.6613191
Gene_850445	-0.0015653	0.9132180	-0.1061582	0.6571565
Disease_D009402	0.0040103	0.8860870	0.0982492	0.6529776
Disease_D004931	0.0023307	0.8875667	-0.1206361	0.6461974
Disease_D014006	0.0053524	0.8987087	-0.1859384	0.6445034
Disease_D006939	0.0084801	0.9076344	0.1290907	0.6413850
Disease_D009410	0.0018515	0.9078394	0.0079242	0.6377730
Gene_26302740	-0.0008296	0.8986007	-0.0975703	0.6352893
Disease_D001284	-0.0012928	0.8785918	0.0386573	0.6274108
Disease_D015863	0.0103406	0.8979422	-0.0626400	0.6205314
Gene_100862683	-0.0042703	0.8879001	-0.1142508	0.6193511
Disease_D000230	0.0049940	0.8957444	-0.0457284	0.6171082
Disease_D008223	0.0040812	0.9161638	0.0756063	0.6056874
Disease_D007008	0.0021507	0.8862924	-0.0023364	0.6040906
Gene_3347	0.0007604	0.8906444	0.0872100	0.6040651
Disease_D015299	0.0144504	0.8986586	0.0606599	0.5926765
Disease_D004401	0.0031732	0.9160118	-0.0980383	0.5869815
Gene_1555	0.0007660	0.8992061	0.0469462	0.5857412
Disease_D015835	0.0048326	0.9122005	0.0050797	0.5852621

Disease_D008180	0.0022519	0.8871992	0.0878263	0.5757224
Disease_D004673	-0.0003538	0.8912674	-0.1037612	0.5729796
Disease_D007565	0.0007236	0.8827867	-0.0908152	0.5683178
Gene_5443	-0.0079637	0.9018441	0.0195583	0.5574407
Gene_13080328	0.0000247	0.8814063	0.0659048	0.5493172
Gene_290	0.0029906	0.9051307	0.1682609	0.5412733
Disease_D016715	0.0014572	0.9022551	0.1690053	0.5307844
Disease_D009190	0.0034088	0.8825181	0.2142433	0.5291704
Disease_D001765	0.0057006	0.8973634	-0.0251760	0.5282308
Disease_D015746	0.0025333	0.8948125	0.0922873	0.5190388
Disease_D004927	0.0027326	0.8996806	0.0765415	0.5184601
Disease_D009196	-0.0039090	0.8859143	-0.0082503	0.5169005
Gene_851029	0.0011182	0.8857247	-0.1016669	0.5118726
Disease_C567712	-0.0021154	0.8842564	0.1015031	0.5058134
Gene_2335	0.0033509	0.8885094	0.0116101	0.5016741
Disease_D016585	0.0045780	0.8923382	0.2330750	0.5007839
Gene_856398	0.0002844	0.9038096	-0.1911460	0.4971650
Disease_C535590	0.0030156	0.9034694	-0.1383930	0.4919349
Disease_D054198	0.0040924	0.8950519	-0.1752640	0.4848480
Disease_D003680	0.0073067	0.9038474	0.1340089	0.4843278
Gene_851613	0.0025423	0.8838351	0.1416239	0.4772957
Disease_D010195	0.0043524	0.8942605	0.0462764	0.4728296
Disease_D014009	0.0030067	0.8974027	-0.0039644	0.4723367
Disease_C566808	0.0081537	0.9030870	-0.0554312	0.4706956
Disease_D006849	0.0124215	0.8958899	0.0006146	0.4645400
Disease_D011552	-0.0009807	0.8948833	0.0115193	0.4606926
Disease_D005955	0.0065283	0.8804665	0.0466543	0.4556530
Disease_D010585	0.0022162	0.8835682	-0.0093440	0.4532851
Disease_D002821	0.0029608	0.8964969	-0.1039864	0.4433463
Disease_D010532	0.0076193	0.9029395	0.0690020	0.4399263
Gene_100053958	-0.0010224	0.8871183	-0.1041384	0.4381101
Disease_D010538	-0.0018566	0.9037349	0.1001891	0.4364118
Disease_D009175	0.0058166	0.9045205	-0.1638986	0.4335186
Disease_D053717	0.0041774	0.9063810	0.0099709	0.4297025
Disease_D019337	-0.0028058	0.8950336	0.0066616	0.4263149
Gene_3558	-0.0021525	0.9067757	0.0196183	0.4250127
Disease_D014947	0.0059220	0.8848428	0.1452499	0.4244423
Gene_7124	-0.0005140	0.8936934	0.1346777	0.4221240
Disease_D016778	0.0023496	0.8889754	-0.1710435	0.4210698
Disease_D007710	0.0061374	0.8871147	-0.1343189	0.4206517
Disease_D003424	0.0024549	0.9045613	0.1410461	0.4192345
Disease_C531821	0.0036544	0.9026802	0.0625839	0.4081941
Disease_D014627	0.0014855	0.9038455	0.1459917	0.3996585
Disease_D009325	-0.0003800	0.8828473	0.0169536	0.3989931

Disease_D055732	0.0007336	0.9149358	0.1380944	0.3974881
Disease_D010493	0.0067061	0.8954273	-0.0229720	0.3933054
Disease_D003731	-0.0023618	0.8966439	-0.0485491	0.3920448
Disease_D006258	0.0085826	0.9011031	0.0495744	0.3913070
Disease_C565534	0.0034066	0.8868754	0.0349908	0.3866487
Disease_D017676	-0.0011883	0.8924623	0.0089493	0.3834364
Gene_16196	0.0000438	0.8963575	-0.0435383	0.3824297
Gene_54205	0.0053051	0.9020213	0.0629328	0.3795628
Disease_D018792	0.0047616	0.8937764	-0.1984504	0.3786583
Disease_D001437	0.0041519	0.8946904	0.0652567	0.3784619
Disease_D015658	0.0048538	0.8839761	0.0202406	0.3777641
Gene_3552	0.0061777	0.8983711	-0.0445707	0.3756001
Disease_C565043	-0.0008922	0.9022868	-0.0750668	0.3738498
Gene_3553	0.0025187	0.9046556	0.2372601	0.3731180
Disease_D006105	0.0018362	0.8991905	-0.0739635	0.3730036
Gene_920	0.0064979	0.9006984	-0.0193718	0.3673436
Disease_D009503	0.0026825	0.8844903	0.0335272	0.3669779
Disease_D017827	-0.0002030	0.8886069	0.0930585	0.3630571
Disease_D014245	-0.0003038	0.9129846	0.2440524	0.3592929
Disease_D006967	0.0018142	0.8854276	0.0775194	0.3565315
Disease_D006944	0.0027018	0.9168078	0.1266506	0.3561426
Gene_3574	0.0068124	0.8947782	-0.0494295	0.3496241
Disease_D005764	0.0007770	0.9106218	-0.0522938	0.3483781
Disease_D008583	0.0031991	0.8901023	-0.0123282	0.3479486
Disease_D013281	0.0018113	0.9075710	0.0386020	0.3461025
Disease_D001523	-0.0000312	0.8924987	0.2363585	0.3455120
Disease_D009164	-0.0017848	0.8893575	-0.1262717	0.3404954
Gene_3569	0.0110732	0.9054776	0.1282116	0.3403705
Disease_D015470	0.0090482	0.9070770	0.0902506	0.3356720
Disease_D014008	-0.0001249	0.8970042	0.1546466	0.3288156
Gene_3576	0.0050932	0.9067104	-0.2112243	0.3288121
Disease_C565469	0.0021589	0.9045888	-0.0404509	0.3246768
Disease_D005891	0.0135138	0.8856338	0.2766025	0.3199268
Disease_D009877	-0.0015819	0.8952766	0.0399220	0.3182461
Gene_10678	0.0064191	0.8869236	0.1915046	0.3159889
Disease_D015179	0.0009704	0.8875442	0.0315663	0.3133993
Gene_3605	0.0012142	0.8970609	-0.1695068	0.3122140
Disease_D013203	-0.0017343	0.8918076	0.0280824	0.3115327
Gene_81502	0.0014324	0.9136146	0.0964298	0.3110681
Disease_D020766	0.0027604	0.8849156	-0.1624973	0.3075759
Disease_D007249	0.0073476	0.9057811	-0.1279940	0.3051620
Disease_D056650	0.0010671	0.8887495	0.1246756	0.3051120
Disease_D009057	-0.0050742	0.9081832	0.1620225	0.3047931
Gene_4353	0.0097166	0.8951633	0.1290068	0.3022480

Disease_D009062	0.0013271	0.8931413	-0.1023200	0.3016444
Disease_D000163	-0.0014743	0.9031728	0.1632469	0.2998188
Disease_D013771	0.0114847	0.8857936	-0.0243175	0.2977647
Disease_D011537	-0.0020791	0.8870040	0.0707689	0.2966995
Disease_D019283	0.0073516	0.9011289	0.0038007	0.2915949
Disease_D011655	0.0009089	0.9045111	-0.0764608	0.2907934
Disease_D007752	-0.0049157	0.8931038	0.0715579	0.2898719
Disease_D013280	0.0031256	0.8840833	0.1013889	0.2864139
Disease_D003453	-0.0023525	0.8926173	0.1033818	0.2848285
Disease_D005334	0.0084230	0.8981992	0.1645252	0.2830223
Disease_C566419	0.0057308	0.9075477	0.0292328	0.2824848
Disease_D015821	0.0006346	0.9088278	0.0403166	0.2820600
Disease_D003428	0.0045737	0.8898337	0.0958808	0.2792485
Gene_853188	0.0070462	0.8912614	0.2009902	0.2704372
Disease_C536972	0.0074379	0.8983325	0.1071857	0.2699538
Gene_1440	-0.0067097	0.9079861	-0.1145544	0.2662130
Disease_D003643	0.0041456	0.8933332	-0.2422104	0.2642030
Disease_D004696	-0.0029358	0.8708941	-0.0326165	0.2633628
Disease_D002179	0.0018481	0.8907608	0.0397356	0.2578590
Disease_D028361	0.0034056	0.8963504	-0.1752625	0.2556720
Disease_D001249	0.0018110	0.9020606	-0.1524644	0.2544242
Disease_D016919	0.0084947	0.8926480	0.0461518	0.2494883
Gene_6998	0.0032849	0.8767342	-0.0217207	0.2490076
Disease_D001927	0.0017098	0.8984905	-0.1072646	0.2483580
Disease_C565957	0.0013780	0.8996228	0.0517947	0.2482154
Disease_D003141	0.0100822	0.9079441	-0.1961620	0.2461143
Disease_D013927	0.0055517	0.9005787	0.1136265	0.2455560
Disease_D008206	0.0003801	0.8879705	0.0818977	0.2443289
Disease_D058387	-0.0018869	0.8990942	-0.0462237	0.2432994
Disease_D008288	-0.0005907	0.8883825	-0.1669071	0.2415256
Disease_D012131	0.0037642	0.8857274	-0.1660457	0.2397418
Disease_D008107	0.0031974	0.8860818	0.0888342	0.2393866
Disease_D004194	0.0043155	0.9084082	-0.0711561	0.2392038
Disease_D016469	0.0115589	0.8848442	0.0550667	0.2374599
Gene_3458	-0.0004801	0.9054303	-0.1536555	0.2358160
Disease_D001327	-0.0036210	0.9054132	0.0472853	0.2312348
Gene_3557	-0.0023594	0.9107422	0.1804935	0.2306896
Disease_D006402	0.0052832	0.8841648	0.0633175	0.2273141
Disease_D004941	-0.0042791	0.8961423	-0.1351770	0.2260987
Disease_D012769	0.0009956	0.8897666	-0.0613034	0.2228792
Disease_D003092	0.0014614	0.9022573	0.1208685	0.2226566
Disease_D011014	0.0032156	0.8949156	-0.0310856	0.2211311
Disease_D008581	0.0063288	0.8971836	-0.1783828	0.2200317
Disease_D018410	0.0017768	0.8983662	0.1055039	0.2200055

Gene_856845	0.0006666	0.8958329	-0.1186880	0.2182947
Disease_D002761	-0.0038538	0.9009648	0.0629479	0.2162094
Disease_D015817	0.0017699	0.8932233	0.0395523	0.2154403
Disease_D018805	0.0026517	0.8917569	-0.1999609	0.2102765
Disease_D008607	0.0055508	0.8895359	-0.0409692	0.2067233
Gene_25712	0.0025616	0.9157341	-0.0167556	0.2065563
Disease_D014564	0.0000068	0.8925312	0.0620877	0.2062627
Disease_D058365	-0.0016503	0.8941932	0.0049331	0.2041426
Disease_D007154	0.0026851	0.8971616	0.0287850	0.1993341
Disease_D006099	-0.0009592	0.8942521	0.1160670	0.1987979
Disease_D016638	0.0061219	0.9024056	0.0274302	0.1984141
Disease_D015473	0.0109810	0.9081823	0.0422122	0.1948072
Disease_D009894	0.0011033	0.8966110	-0.0755115	0.1932927
Disease_D016720	-0.0069831	0.9065383	-0.1263191	0.1928290
Disease_D003872	-0.0024920	0.8911837	0.0615061	0.1865065
Disease_D009362	0.0023890	0.8988374	0.0838476	0.1838308
Disease_D018458	-0.0019034	0.8855517	-0.0507843	0.1818774
Disease_D063646	-0.0029208	0.9054620	-0.1484406	0.1796018
Disease_D003316	-0.0060491	0.8997373	0.0597885	0.1785407
Disease_211750	0.0095884	0.8866579	-0.0166975	0.1783186
Disease_D005767	0.0033518	0.9029871	-0.0383118	0.1780738
Gene_16171	-0.0031302	0.8840044	0.0762585	0.1778327
Disease_D005928	0.0003837	0.9014751	-0.0875659	0.1749260
Disease_D055499	0.0021643	0.8995992	-0.1967128	0.1746785
Disease_D034721	0.0055179	0.8882807	-0.1338196	0.1705870
Disease_D002825	0.0087010	0.9016247	-0.1777219	0.1689432
Disease_D008103	0.0079483	0.9113895	-0.0417050	0.1681110
Disease_D005355	-0.0038580	0.8908054	0.0309154	0.1678546
Gene_850930	0.0012895	0.9093814	-0.1535226	0.1666101
Disease_D010518	0.0006699	0.9031187	-0.0225813	0.1661447
Disease_D016399	0.0026609	0.8839922	0.0729352	0.1656946
Disease_D003881	0.0024370	0.8915679	0.0653261	0.1652684
Gene_116562	0.0041329	0.9008074	0.0701447	0.1634586
Disease_D010304	-0.0002781	0.8933246	-0.0264335	0.1620985
Disease_D013180	0.0079839	0.8917956	0.0943342	0.1603444
Disease_C569516	0.0056544	0.9061484	-0.1733517	0.1597735
Disease_D009135	-0.0010470	0.9078082	-0.2021753	0.1594230
Disease_D009959	0.0043015	0.8905930	0.0063988	0.1582459
Disease_D003920	0.0059190	0.9089696	0.0809575	0.1577707
Disease_D001424	-0.0045559	0.9084181	-0.1340423	0.1570340
Disease_D001261	-0.0043667	0.9102763	0.0932998	0.1528293
Disease_D014848	0.0025182	0.8935870	0.2261116	0.1508480
Disease_D016109	-0.0000387	0.8893305	0.0226784	0.1498547
Disease_D020096	0.0036452	0.9019439	0.0630276	0.1483978

Disease_D010212	0.0091806	0.9027825	-0.0857742	0.1483475
Disease_D018798	0.0048875	0.8738609	0.1209812	0.1477956
Disease_D006333	-0.0025833	0.8887412	-0.1878154	0.1474760
Disease_D004342	-0.0020402	0.9054458	0.0716715	0.1468868
Disease_D014839	0.0034551	0.9114629	0.0658541	0.1460806
Disease_D014010	0.0024235	0.9006133	0.0161287	0.1451022
Disease_D056486	0.0063480	0.9026492	-0.0879740	0.1426325
Disease_D009800	0.0025667	0.8829629	0.1645708	0.1424309
Disease_D016470	0.0014874	0.9015528	-0.0142854	0.1422865
Disease_D001228	0.0018828	0.8911045	-0.1089094	0.1418365
Disease_D009091	-0.0007415	0.8864245	-0.0747372	0.1403311
Disease_D012749	-0.0029171	0.9012211	0.0477464	0.1398574
Disease_D001022	0.0008864	0.9061601	-0.0363546	0.1376691
Disease_D002972	0.0015460	0.8811453	-0.0605690	0.1328908
Disease_D003072	-0.0020984	0.9078284	-0.0411274	0.1284396
Disease_D009436	0.0050317	0.8878696	0.0319237	0.1281795
Disease_D008171	0.0029207	0.8850517	0.1241577	0.1272305
Disease_D052016	0.0065333	0.8816571	0.0282857	0.1242859
Disease_D002180	0.0024140	0.8789904	0.0082594	0.1238879
Disease_D013282	0.0005557	0.8964590	0.0448877	0.1231785
Disease_D001855	0.0085716	0.8852547	-0.0118788	0.1204642
Disease_D013568	0.0050643	0.8977237	0.1009821	0.1186691
Disease_D007153	0.0061887	0.8829486	-0.0768442	0.1178064
Disease_D009181	0.0006602	0.8975615	-0.1003750	0.1165198
Disease_D002181	0.0028134	0.8905449	-0.0367444	0.1154500
Disease_D014123	0.0063918	0.8791374	0.1601160	0.1149262
Disease_D003677	0.0104649	0.8955312	0.0738607	0.1140376
Disease_D014860	0.0086623	0.8978710	0.0331412	0.1079847
Disease_D014777	0.0018049	0.8956620	0.1295195	0.1079781
Disease_D007674	0.0041673	0.8948707	-0.0331973	0.1078845
Disease_D015212	0.0037051	0.8964104	0.1522763	0.1077075
Disease_D007239	-0.0004153	0.8861839	-0.0553505	0.1059428
Gene_1401	0.0026583	0.8959628	0.0120931	0.1039948
Disease_D008100	0.0024199	0.8893169	-0.1412591	0.1019119
Disease_D029424	0.0036209	0.8845069	-0.2084028	0.1007097
Disease_D014005	0.0027509	0.8840659	-0.1317030	0.0991100
Disease_D009765	0.0002098	0.8907895	-0.0633302	0.0978003
Disease_D059413	0.0005808	0.8818710	0.1218434	0.0941037
Disease_D007970	0.0046206	0.9106109	0.0510347	0.0871873
Disease_D005128	0.0050918	0.8928823	-0.0864452	0.0861382
Disease_C536777	0.0050025	0.8857571	0.1930629	0.0853676
Disease_D011020	0.0054200	0.8806857	-0.0054173	0.0795301
Disease_C531782	0.0090489	0.9092966	0.0959704	0.0756843
Disease_D002178	0.0065628	0.8895678	-0.1217183	0.0756492

Disease_D009422	0.0002447	0.8987676	0.1207259	0.0721963
Disease_D012772	0.0031740	0.8944064	-0.0320387	0.0720161
Disease_D006323	-0.0036006	0.8832853	-0.0273585	0.0679986
Disease_D006331	-0.0001754	0.9035385	0.1406092	0.0677552
Disease_D012871	0.0024578	0.8861975	-0.0026239	0.0661130
Disease_D003866	-0.0009578	0.9094549	-0.0223817	0.0558543
Disease_D007938	0.0058981	0.8750523	0.0061133	0.0545156
Disease_D000796	0.0048549	0.9071806	-0.0752672	0.0539396
Disease_D012163	0.0033020	0.9035703	-0.0337218	0.0505162
Disease_D001943	0.0053124	0.8927961	-0.0751027	0.0485214
Disease_D018149	0.0093598	0.9072549	0.0625385	0.0476234
Disease_D013746	0.0003723	0.8967765	0.2407378	0.0461639
Disease_D004487	0.0114862	0.8896298	-0.1167651	0.0440167
Disease_D002294	0.0045730	0.9047235	-0.0216802	0.0406151
Disease_D003967	0.0039021	0.8886622	0.0633168	0.0386489
Disease_D000038	-0.0021233	0.9024011	-0.0961478	0.0376632
Disease_D008659	-0.0011726	0.8854360	0.0435192	0.0369156
Disease_D014456	0.0011755	0.8953575	-0.0084077	0.0335922
Disease_D009260	0.0039936	0.8786699	0.1173023	0.0333579
Disease_D010146	0.0003258	0.8772149	-0.0695371	0.0329676
Disease_D005402	0.0026627	0.8887464	0.1172660	0.0324770
Disease_D000740	0.0008198	0.8950556	0.0583392	0.0323323
Disease_D014376	0.0028918	0.8996525	0.0209833	0.0313777
Disease_D011776	-0.0065099	0.8874653	-0.0022689	0.0296694
Disease_D014552	-0.0014149	0.9028077	-0.1586994	0.0286411
Disease_D003586	0.0052116	0.9027060	0.0074726	0.0260189
Disease_D002177	0.0026270	0.8763758	0.0461562	0.0230245
Disease_D014987	0.0035145	0.8849034	-0.0092579	0.0199556
Disease_D004802	0.0001916	0.8907286	-0.1265019	0.0191407
Disease_D010019	0.0001999	0.8892638	0.0480246	0.0145823
Disease_D010190	0.0017218	0.8958375	-0.0789058	0.0110123
Disease_C566367	0.0023071	0.8879994	0.0592059	0.0104703
Disease_D003110	0.0054052	0.8910705	-0.0412367	0.0103889
Disease_D006223	-0.0038696	0.8934729	0.0988301	0.0095141
Disease_D001660	0.0115872	0.9068950	0.0589708	0.0073333
Disease_D004211	0.0024150	0.8965697	-0.0094508	0.0052502
Disease_C566273	-0.0108822	0.8964763	-0.0610634	0.0052242
Gene_113246	-0.0066445	0.8952228	0.1180810	0.0030460
Disease_D001791	-0.0011611	0.8918024	-0.0707165	-0.0019918
Disease_C565742	0.0041685	0.8742913	-0.1591411	-0.0054683
Disease_D060737	0.0071415	0.9048180	0.0244134	-0.0076567
Disease_D018307	0.0051751	0.9121355	0.1584386	-0.0103551
Disease_D008175	0.0020288	0.9047405	-0.0623586	-0.0107575
Disease_D006069	0.0039959	0.8942817	-0.0692713	-0.0168996

Disease_C564973	0.0032578	0.8951313	0.2134532	-0.0210736
Disease_D009336	0.0042730	0.9074938	0.2062178	-0.0212112
Disease_D009103	-0.0029152	0.8728112	0.0242697	-0.0255016
Disease_D003093	0.0041083	0.8864434	-0.1064257	-0.0276504
Disease_D002908	-0.0055627	0.8955210	-0.1082146	-0.0295295
Disease_D003876	0.0047777	0.8947603	-0.1306300	-0.0359047
Disease_D019966	-0.0057436	0.8933763	-0.1832246	-0.0375546
Disease_D051437	0.0009283	0.8916548	0.1923070	-0.0408030
Disease_D006551	0.0067339	0.9016141	0.1226051	-0.0496040
Disease_D006470	-0.0072529	0.8862549	-0.0489917	-0.0671171
Disease_D007634	-0.0001395	0.8911835	0.0926303	-0.0671612
Disease_D002277	0.0062931	0.8982762	-0.0198302	-0.0682807
Disease_D007246	0.0053367	0.9045538	0.1509045	-0.0859313
Disease_D009369	-0.0031710	0.8700444	0.1313590	-0.0871864
Disease_D064420	0.0068292	0.8858863	0.0744598	-0.0926655
Disease_D012640	-0.0019874	0.9069298	0.2241908	-0.0963407
Disease_D007410	0.0064562	0.8985659	-0.0124876	-0.0974590
Disease_D012421	-0.0003782	0.8859102	0.0772908	-0.1119310
Disease_D006461	0.0036593	0.8867765	-0.1367328	-0.1158345
Disease_D017093	0.0096169	0.9169186	0.0897075	-0.1186729
Disease_D060085	-0.0019043	0.8947738	0.1812467	-0.1195401
Disease_D002690	0.0014324	0.8970910	-0.0921807	-0.1244361
Disease_D003550	0.0056175	0.8804083	-0.0258367	-0.1301926
Disease_D004890	0.0005876	0.8977826	-0.0532873	-0.1541878
Disease_D015431	0.0054331	0.8873496	0.1161244	-0.1950300
Disease_D030342	0.0050934	0.8979007	-0.0058214	-0.2093056

Appendix IV -- Confidence Score

Concept	Confidence Score
Gene_920	0.999173343
Disease_D008581	0.999064371
Disease_D003586	0.998979641
Disease_D009877	0.998834133
Disease_C567712	0.998654366
Disease_D054198	0.998138458
Gene_853823	0.997654319
Disease_D005402	0.997251395
Disease_D007410	0.996747606
Gene_4155	0.996588171
Disease_D014376	0.996405136
Disease_D063646	0.996249706
Disease_D015821	0.995808154
Gene_3458	0.995397717
Disease_D010532	0.99521789
Disease_D009369	0.995211862
Disease_D059413	0.995171905
Gene_1509	0.995003223
Disease_D001261	0.994862333
Disease_D006069	0.99446347
Disease_C536777	0.992157772
Disease_D009196	0.991606832
Disease_D009260	0.991149586
Disease_D016720	0.990766078
Disease_D005891	0.990368485
Disease_D009164	0.990034074
Disease_D003866	0.989515621
Gene_113246	0.9893841
Disease_D007565	0.989238679
Disease_D001284	0.988872826
Gene_850445	0.988434196
Disease_D005767	0.988135561
Gene_7431	0.987979412
Disease_D006223	0.987784391
Disease_D001471	0.987747252
Disease_D009410	0.987586439
Disease_D010585	0.987205803
Disease_D005764	0.986610323
Disease_C535342	0.985995948
Disease_D010518	0.985892639

Concept	Confidence Score
Disease_D005128	0.985577829
Disease_D006967	0.985546559
Disease_D004941	0.985475555
Disease_D016638	0.985413522
Disease_C565469	0.985350132
Disease_D010304	0.984953314
Disease_D020096	0.984586418
Disease_D007008	0.984299898
Disease_D002177	0.982657155
Disease_D010019	0.982331175
Disease_D014689	0.982221365
Disease_D056650	0.982056469
Disease_D011537	0.981877416
Disease_D003677	0.98185908
Gene_3557	0.981825277
Disease_D018798	0.980177402
Disease_D010212	0.980142996
Gene_855030	0.980041385
Disease_D008228	0.979685426
Disease_D058387	0.979365543
Gene_1440	0.97911492
Disease_D003645	0.978744626
Disease_D018149	0.978653975
Disease_D015473	0.978554398
Disease_C531782	0.977893431
Disease_D008103	0.977129728
Disease_D002277	0.977036439
Disease_D003881	0.976773843
Disease_D007794	0.976630569
Disease_D013568	0.976490609
Disease_D004487	0.975732837
Disease_D007970	0.975038823
Disease_D014777	0.974844381
Gene_3569	0.974020958
Disease_D016919	0.973917365
Gene_16171	0.97291252
Disease_C564973	0.972767634
Disease_D001943	0.97271087
Disease_C565957	0.972698942
Disease_D018792	0.971732825
Disease_D013282	0.971650355
Disease_D001228	0.971268445
Disease_D052016	0.970682442

Concept	Confidence Score
Gene_1595	0.970264256
Disease_D015179	0.96963641
Disease_D019337	0.968588591
Gene_3574	0.9685103
Disease_D009190	0.96827364
Disease_D011014	0.967835337
Disease_D006258	0.96780178
Disease_D006331	0.966605924
Disease_D011020	0.965891309
Gene_13080328	0.96561265
Disease_D007640	0.965151727
Disease_D009325	0.964827806
Disease_D003643	0.964265794
Gene_851613	0.964024335
Disease_D012769	0.963804737
Disease_D004401	0.963402331
Gene_3347	0.962929964
Disease_D016470	0.962780818
Disease_D056486	0.962779798
Disease_D008175	0.962456777
Disease_D012640	0.962217808
Gene_16196	0.962118149
Disease_D003731	0.962009907
Disease_D004802	0.961108189
Disease_D011655	0.960910499
Disease_D055499	0.960675091
Disease_D008583	0.96041435
Disease_D010211	0.960302651
Disease_C566367	0.960083799
Disease_D003072	0.959845781
Gene_290	0.959825456
Disease_D001927	0.959504724
Disease_D006192	0.959240615
Gene_3553	0.958304584
Disease_D003920	0.957509324
Disease_D002180	0.957296751
Disease_D005355	0.956982344
Disease_D012376	0.956683934
Disease_D012772	0.956441417
Disease_D012131	0.956127554
Disease_C531821	0.955790848
Disease_D007674	0.954862125
Disease_D058365	0.954712436

Concept	Confidence Score
Disease_D004696	0.954630882
Disease_D003092	0.954560414
Gene_81502	0.954439789
Disease_D003876	0.954125121
Disease_D010146	0.95399414
Disease_D004890	0.953659907
Disease_D009057	0.953586161
Disease_D008171	0.952817872
Gene_100862683	0.952713013
Disease_D008288	0.95261915
Disease_D002825	0.952307954
Disease_D015658	0.951855034
Disease_D003453	0.951663792
Disease_D000163	0.951531768
Disease_D009422	0.94967212
Disease_D008206	0.948222294
Disease_D018307	0.948135174
Disease_D001791	0.947783233
Disease_D002294	0.947580013
Disease_D007154	0.946992397
Disease_D017093	0.94681792
Disease_D010505	0.946678042
Disease_D013771	0.946635872
Disease_D014008	0.946415812
Disease_D003680	0.94636482
Disease_D012749	0.944644511
Disease_D010195	0.944222331
Disease_D008223	0.943657398
Disease_D000740	0.943292869
Disease_D003316	0.942688838
Disease_D008107	0.942617476
Disease_D001424	0.942501381
Disease_D010538	0.941231042
Disease_D014987	0.940485414
Disease_D004405	0.939363718
Disease_D009800	0.938730702
Disease_D016109	0.938448071
Disease_D029424	0.938272998
Disease_D013280	0.93586424
Disease_C565742	0.935699215
Disease_D014456	0.93568169
Disease_D004211	0.934393738
Disease_D007246	0.93339799

Concept	Confidence Score
Disease_D009175	0.933352858
Disease_D012871	0.933204892
Gene_100053958	0.93266359
Gene_2335	0.932626605
Disease_D000230	0.932242632
Disease_D019283	0.931854427
Disease_D001523	0.931409657
Disease_D009181	0.928383842
Disease_D009103	0.928096687
Gene_6998	0.92797552
Disease_D011552	0.927759171
Disease_D007634	0.927290426
Disease_D007249	0.926744625
Disease_D010493	0.923798233
Disease_D003424	0.923630506
Disease_D018805	0.921915099
Gene_851029	0.921823144
Disease_D064420	0.921444319
Gene_1555	0.921381831
Disease_D018458	0.921246484
Disease_D006402	0.920689806
Disease_D006105	0.920570612
Disease_D034721	0.918893769
Disease_D002181	0.918733232
Gene_3558	0.918459684
Disease_D014947	0.917936563
Disease_D014564	0.916918524
Disease_D006323	0.916257711
Disease_D015746	0.916004121
Disease_D003110	0.915236213
Disease_D028361	0.914479792
Disease_D013281	0.914304137
Gene_4589	0.913838506
Gene_25712	0.913691692
Disease_C535590	0.913273454
Disease_D014245	0.912890822
Disease_D009765	0.911596388
Disease_D058565	0.910835266
Disease_D016399	0.909050725
Disease_D009894	0.908046268
Disease_C566808	0.907706112
Disease_D009091	0.907461274
Disease_D002972	0.907371923

Concept	Confidence Score
Disease_D012480	0.90661031
Disease_D015470	0.906356752
Disease_D013180	0.905700922
Disease_D009135	0.904418334
Disease_D003428	0.903400853
Disease_D017827	0.902809143
Disease_D015299	0.902337492
Gene_4353	0.90096733
Disease_D008607	0.900402114
Disease_D009062	0.899859935
Disease_D000038	0.89955565
Disease_D003967	0.898598995
Gene_5443	0.898470283
Disease_D009336	0.897342484
Disease_D009959	0.895478636
Disease_D001249	0.8953747
Gene_7124	0.894174933
Disease_D002908	0.893542778
Disease_D000796	0.892870475
Disease_D005334	0.892699361
Gene_856398	0.892654777
Disease_C566419	0.891860977
Disease_D007153	0.89185182
Disease_D020766	0.89179185
Disease_D060737	0.890558999
Disease_D015863	0.889942884
Disease_D016469	0.885847062
Disease_D001437	0.885717332
Gene_116562	0.88534458
Disease_D014848	0.885106042
Disease_D014860	0.884370273
Disease_D006099	0.883696347
Disease_D006470	0.882803656
Disease_D001022	0.88187727
Disease_D016715	0.881347746
Disease_D002179	0.880346745
Disease_D006939	0.880061269
Disease_D016585	0.878776193
Disease_D002821	0.877620459
Disease_D005955	0.877342701
Disease_C535390	0.876799643
Disease_D009402	0.875564992
Disease_D016778	0.874962717

Concept	Confidence Score
Gene_3605	0.874912083
Disease_C536972	0.871516779
Disease_D053717	0.871352494
Disease_D013927	0.870508745
Disease_D003872	0.869166404
Disease_D003141	0.868870839
Disease_D009503	0.868653446
Disease_D009362	0.868129544
Disease_D004342	0.867080903
Gene_856845	0.866941959
Disease_D006333	0.864608783
Disease_D012163	0.86458309
Disease_D006849	0.863074362
Disease_D007239	0.862165727
Disease_D001765	0.861489594
Disease_D006461	0.860035369
Disease_D014006	0.856590748
Disease_D015835	0.856413007
Gene_850930	0.853394732
Disease_D019966	0.852983292
Gene_1401	0.850844413
Gene_10678	0.850721508
Disease_D015817	0.849420309
Disease_D014627	0.847705036
Disease_D004927	0.845641851
Disease_D001855	0.845601279
Disease_D060085	0.844303213
Gene_3552	0.840120018
Disease_D030342	0.837675158
Disease_D018410	0.837510377
Disease_D007710	0.833808154
Disease_D014005	0.82935822
Disease_D001327	0.828954656
Disease_D015212	0.824599288
Disease_D008100	0.819817334
Disease_D001660	0.818843336
Disease_C565043	0.818551421
Disease_D014009	0.815868467
Disease_D012421	0.813538894
Disease_D007752	0.80967588
Disease_D013746	0.8092275
Disease_D017676	0.808355749
Disease_C566273	0.807814041

Concept	Confidence Score
Gene_3576	0.801529542
Disease_D002178	0.793663532
Disease_D002761	0.793379321
Disease_D006944	0.790743411
Disease_D014839	0.783685401
Disease_D003550	0.78189858
Disease_D003093	0.780889787
Disease_D055732	0.780842528
Gene_26302740	0.779418588
Disease_D013203	0.778812021
Disease_D014123	0.778339908
Disease_C565534	0.773516268
Disease_D008659	0.773195341
Disease_D015431	0.771867048
Disease_D004673	0.76884234
Disease_D051437	0.766326666
Disease_D004194	0.762579177
Disease_D010190	0.755662408
Disease_D002690	0.746149115
Disease_D009436	0.742194459
Gene_54205	0.738560401
Disease_D014552	0.72955923
Disease_C569516	0.72609558
Disease_D014010	0.721884906
Disease_211750	0.721029818
Disease_D005928	0.708744749
Gene_853188	0.692995608
Disease_D004931	0.689247012
Disease_D011776	0.66335341
Disease_D007938	0.618494276
Disease_D008180	0.565453202
Disease_D006551	0.469778184

Appendix V -- Final IC Result in Experiment Two

ID	IC	Name
Disease_D010505	0.877175998777040000	Familial Mediterranean Fever
Disease_D003645	0.846985897189434000	Death, Sudden
Disease_D014689	0.845284948123037000	Venous Insufficiency
Disease_D058565	0.838141970905325000	Cerebral Ventriculitis
Disease_D007794	0.827782448538365000	Lameness, Animal
Disease_D001471	0.810866410200234000	Barrett Esophagus
Disease_D006192	0.744012879928159000	Haemophilus Infections
Disease_D012480	0.713317720021823000	Salmonella Infections
Disease_C535342	0.712084957708331000	Cataract, zonular
Disease_C535390	0.686403303698487000	Aspergillus niger infection
Disease_D007640	0.681388634322932000	Keratoconus
Disease_D004405	0.676147113115760000	Dysentery, Bacillary
Disease_D012376	0.658209460118499000	Rodent Diseases
Disease_D008228	0.647884661527491000	Lymphoma, Non-Hodgkin
Disease_D010211	0.647735288568664000	Papilledema
Disease_D009410	0.629855884509403000	Nerve Degeneration
Disease_D001284	0.620429578606490000	Atrophy
Disease_D007008	0.594606346573229000	Hypokalemia
Disease_D000230	0.575294541049416000	Adenocarcinoma
Disease_D009402	0.571724314959986000	Nephrosis, Lipoid
Disease_D008223	0.571561433240418000	Lymphoma
Disease_D004401	0.565499378959244000	Dysarthria
Disease_D006939	0.564458113461050000	Hyperemesis Gravidarum
Disease_D007565	0.562201980190341000	Jaundice
Disease_D015863	0.552237539407798000	Iridocyclitis
Disease_D014006	0.552075610862970000	Tinea Capitis
Disease_D015299	0.534794352215698000	Discitis
Disease_D009196	0.512562106099437000	Myeloproliferative Disorders
Disease_D009190	0.512381743354296000	Myelodysplastic Syndromes
Disease_C567712	0.505132720326173000	Retinitis Pigmentosa, Concentric
Disease_D015835	0.501226040701993000	Ocular Motility Disorders
Disease_D054198	0.483945397785190000	Precursor Cell Lymphoblastic Leukemia-Lymphoma
Disease_D015746	0.475441676641615000	Abdominal Pain
Disease_D016715	0.467805606620665000	Proteus Syndrome
Disease_D003680	0.458350728352232000	Deglutition Disorders
Disease_D001765	0.455065376987814000	Blind Loop Syndrome
Disease_C535590	0.449271081029067000	Carrington syndrome
Disease_D010585	0.447485649991632000	Phagocyte Bactericidal Dysfunction
Disease_D010195	0.446456304785520000	Pancreatitis
Disease_D004931	0.445389653558421000	Esophageal Achalasia

ID	IC	Name
Disease_D004673	0.440530952659187000	Encephalomyelitis, Acute Disseminated
Disease_D016585	0.440076934439524000	Vaginosis, Bacterial
Disease_D004927	0.438431554565610000	Escherichia coli Infections
Disease_D010532	0.437822549991596000	Peritoneal Diseases
Disease_D011552	0.427411853027763000	Pseudomonas Infections
Disease_C566808	0.427253286303407000	Phagocytosis, Plasma-Related Defect in
Disease_D019337	0.412923652240273000	Hematologic Neoplasms
Disease_D010538	0.410764331436963000	Peritonitis
Disease_D009175	0.404625785722090000	Mycoplasma Infections
Disease_D006849	0.400932542613529000	Hydrocephalus
Disease_D005955	0.399763817924522000	Glucosephosphate Dehydrogenase Deficiency
Disease_C531821	0.390148151436946000	Stenotrophomonas maltophilia bacteremia
Disease_D014947	0.389611098239583000	Wounds and Injuries
Disease_D002821	0.389089802067402000	Chorioamnionitis
Disease_D003424	0.387217758906799000	Crohn Disease
Disease_D014009	0.385364627130764000	Onychomycosis
Disease_D009325	0.384959612962378000	Nausea
Disease_D006258	0.378707607397355000	Head and Neck Neoplasms
Disease_D003731	0.377151022135564000	Dental Caries
Disease_D053717	0.374422362701053000	Pneumonia, Ventilator-Associated
Disease_D016778	0.368420351049792000	Malaria, Falciparum
Disease_D018792	0.367954665452438000	Encephalitis, Viral
Disease_D010493	0.363334797877934000	Pericarditis
Disease_D015658	0.359576693997225000	HIV Infections
Disease_D006967	0.351378393838661000	Hypersensitivity
Disease_D007710	0.350742845643799000	Klebsiella Infections
Disease_D005764	0.343713510200387000	Gastroesophageal Reflux
Disease_D006105	0.343376126744622000	Granulomatous Disease, Chronic
Disease_D014627	0.338792499000544000	Vaginitis
Disease_D009164	0.337101996396221000	Mycobacterium Infections
Disease_D001437	0.335210288538912000	Bacteriuria
Disease_D008583	0.334174810247033000	Meningitis, Haemophilus
Disease_D014245	0.327995240194883000	Trichomonas Infections
Disease_D017827	0.327771329225697000	Machado-Joseph Disease
Disease_D008180	0.325544139630695000	Lupus Erythematosus, Systemic
Disease_D001523	0.321813243688957000	Mental Disorders
Disease_C565469	0.319920339257379000	Immune Deficiency Disease
Disease_D009503	0.318776591944553000	Neutropenia
Disease_D009877	0.317875063798126000	Endophthalmitis
Disease_D005891	0.316845448245995000	Gingivitis
Disease_D013281	0.316442980318804000	Stomatitis, Aphthous
Disease_D014008	0.311196291721800000	Tinea Pedis
Disease_D055732	0.310375602927053000	Pulmonary Aspergillosis

ID	IC	Name
Disease_D017676	0.309953051666283000	Lichen Planus, Oral
Disease_C565043	0.306015268203619000	Arene Oxide Detoxification Defect
Disease_D015470	0.304238548560899000	Leukemia, Myeloid, Acute
Disease_D015179	0.303883357828440000	Colorectal Neoplasms
Disease_D056650	0.299637188221411000	Vulvodynia
Disease_C565534	0.299079070521221000	Granulomatous Disease with Defect in Neutrophil Chemotaxis
Disease_D011537	0.291322474204455000	Pruritus
Disease_D009057	0.290646472279546000	Stomatognathic Diseases
Disease_D000163	0.285287069213410000	Acquired Immunodeficiency Syndrome
Disease_D007249	0.282807227282387000	Inflammation
Disease_D013771	0.281874735804365000	Tetralogy of Fallot
Disease_D006944	0.281617398767199000	Hyperglycemic Hyperosmolar Nonketotic Coma
Disease_D015821	0.280877675023230000	Eye Infections, Fungal
Disease_D011655	0.279426363361805000	Pulmonary Embolism
Disease_D020766	0.274293691039964000	Intracranial Embolism
Disease_D019283	0.271723991650004000	Pancreatitis, Acute Necrotizing
Disease_D009062	0.271437723385690000	Mouth Neoplasms
Disease_D003453	0.271061011970168000	Cryptococcosis
Disease_D013280	0.268044534320573000	Stomatitis
Disease_D003643	0.254761955754958000	Death
Disease_D005334	0.252653785881147000	Fever
Disease_D003428	0.252273338443631000	Cross Infection
Disease_C566419	0.251937195907186000	Orofacial Cleft 2
Disease_D004696	0.251414257267097000	Endocarditis
Disease_D016919	0.242981040080678000	Meningitis, Cryptococcal
Disease_D013203	0.242625393179709000	Staphylococcal Infections
Disease_C565957	0.241438849369138000	Amyotrophic Lateral Sclerosis 2, Juvenile
Disease_D001927	0.238300642028280000	Brain Diseases
Disease_D058387	0.238279058424467000	Candidemia
Disease_C536972	0.235269255227497000	Torulopsis
Disease_D007752	0.234702262556063000	Obstetric Labor, Premature
Disease_D028361	0.233806911796197000	Mitochondrial Diseases
Disease_D008206	0.231678083010187000	Lymphatic Diseases
Disease_D008288	0.230081916911917000	Malaria
Disease_D012131	0.229223700244429000	Respiratory Insufficiency
Disease_D001249	0.227804978032502000	Asthma
Disease_D002179	0.227005350245739000	Candidiasis, Cutaneous
Disease_D008107	0.225649995727482000	Liver Diseases
Disease_D004941	0.222814699165292000	Esophagitis
Disease_D008581	0.219825810654872000	Meningitis
Disease_D012769	0.214812029810655000	Shock
Disease_D011014	0.214018479461337000	Pneumonia
Disease_D003141	0.213841563214146000	Communicable Diseases

ID	IC	Name
Disease_D013927	0.213758577902975000	Thrombosis
Disease_D003092	0.212539211995557000	Colitis
Disease_D016469	0.210353179351705000	Fungemia
Disease_D006402	0.209285829280895000	Hematologic Diseases
Disease_D016638	0.195519924482702000	Critical Illness
Disease_D058365	0.194897507593441000	Candidiasis, Invasive
Disease_D018805	0.193857079758269000	Sepsis
Disease_D001327	0.191683179513503000	Autoimmune Diseases
Disease_D016720	0.191048459615235000	Pneumocystis Infections
Disease_D015473	0.190629429294772000	Leukemia, Promyelocytic, Acute
Disease_D014564	0.189126083766272000	Urogenital Abnormalities
Disease_D007154	0.188767891230284000	Immune System Diseases
Disease_D008607	0.186134058383162000	Intellectual Disability
Disease_D018410	0.184256899533252000	Pneumonia, Bacterial
Disease_D015817	0.182999394148495000	Eye Infections
Disease_D004194	0.182411811068103000	Disease
Disease_D063646	0.178928288311771000	Carcinogenesis
Disease_D005767	0.175961106946680000	Gastrointestinal Diseases
Disease_D006099	0.175676935522358000	Granuloma
Disease_D009894	0.175518694332894000	Opportunistic Infections
Disease_D002761	0.171536052489976000	Cholangitis
Disease_D003316	0.168308331210049000	Corneal Diseases
Disease_D055499	0.167809316726573000	Catheter-Related Infections
Disease_D018458	0.167553905628042000	Persistent Vegetative State
Disease_D008103	0.164266223081316000	Liver Cirrhosis
Disease_D010518	0.163800835540810000	Periodontitis
Disease_D003872	0.162105218333672000	Dermatitis
Disease_D003881	0.161429870903752000	Dermatomycoses
Disease_D002825	0.160885907335831000	Chorioretinitis
Disease_D005355	0.160633866903778000	Fibrosis
Disease_D010304	0.159659466652241000	Paronychia
Disease_D009362	0.159588972421326000	Neoplasm Metastasis
Disease_D034721	0.156751348099385000	Mastocytosis, Systemic
Disease_D001261	0.152044162728631000	Pulmonary Atelectasis
Disease_D003920	0.151066938419593000	Diabetes Mellitus
Disease_D016399	0.150624818310839000	Lymphoma, T-Cell
Disease_D001424	0.148004785052141000	Bacterial Infections
Disease_D020096	0.146110461548467000	Zygomycosis
Disease_D010212	0.145401803918888000	Papilloma
Disease_D013180	0.145224090721356000	Sprains and Strains
Disease_D018798	0.144865953751875000	Anemia, Iron-Deficiency
Disease_D009135	0.144185051431650000	Muscular Diseases
Disease_D009959	0.141705828270990000	Oropharyngeal Neoplasms

ID	IC	Name
Disease_D016109	0.140630844679268000	Epidermolysis Bullosa, Junctional
Disease_D001228	0.137761325682020000	Aspergillosis
Disease_D056486	0.137323731884601000	Chemical and Drug Induced Liver Injury
Disease_D016470	0.136990678799740000	Bacteremia
Disease_D009800	0.133704246161242000	Oculocerebrorenal Syndrome
Disease_D014848	0.133516530083213000	Vulvovaginitis
Disease_D012749	0.132115578180491000	Sexually Transmitted Diseases
Disease_211750	0.128573009706355000	C SYNDROME
Disease_D006333	0.127509060031396000	Heart Failure
Disease_D004342	0.127362735458771000	Drug Hypersensitivity
Disease_D009091	0.127345015792450000	Mucormycosis
Disease_D005928	0.123977871955642000	Glossitis
Disease_D003072	0.123282241846674000	Cognition Disorders
Disease_D001022	0.121407224930842000	Aortic Valve Insufficiency
Disease_D008171	0.121227461328765000	Lung Diseases
Disease_D052016	0.120642147302708000	Mucositis
Disease_D002972	0.120581385871687000	Cleft Palate
Disease_D013282	0.119686437482142000	Stomatitis, Denture
Disease_D002180	0.118597445236216000	Candidiasis, Oral
Disease_C569516	0.116010852830392000	Trichophyton infection
Disease_D013568	0.115879290974721000	Pathological Conditions, Signs and Symptoms
Disease_D014839	0.114481243896024000	Vomiting
Disease_D003677	0.111968804952655000	Deficiency Diseases
Disease_D009181	0.108175073162177000	Mycoses
Disease_D002181	0.106067753672412000	Candidiasis, Vulvovaginal
Disease_D014777	0.105261871274100000	Virus Diseases
Disease_D007153	0.105065836626068000	Immunologic Deficiency Syndromes
Disease_D014010	0.104747111651514000	Tinea Versicolor
Disease_D007674	0.103014812444439000	Kidney Diseases
Disease_D001855	0.101864711684694000	Bone Marrow Diseases
Disease_D014860	0.095498497539789700	Warts
Disease_D009436	0.095134140814177300	Neural Tube Defects
Disease_D029424	0.094493170298353100	Pulmonary Disease, Chronic Obstructive
Disease_D059413	0.093649396834409800	Intraabdominal Infections
Disease_D007239	0.091340238858317300	Infection
Disease_D014123	0.089451651060216900	Toxoplasmosis
Disease_D009765	0.089154399755845500	Obesity
Disease_D015212	0.088815558951961400	Inflammatory Bowel Diseases
Disease_D007970	0.085010978257174400	Leukopenia
Disease_D005128	0.084895947910599200	Eye Diseases
Disease_C536777	0.084698118098565500	Systemic candidiasis
Disease_D008100	0.083549144193266800	Liver Abscess
Disease_D014005	0.082197674520113400	Tinea

ID	IC	Name
Disease_D011020	0.076817394046165200	Pneumonia, Pneumocystis
Disease_C531782	0.074011174015031900	Endemic treponematosis caused by Treponema carateum
Disease_D012772	0.068879149960410300	Shock, Septic
Disease_D009422	0.068562782433520500	Nervous System Diseases
Disease_D006331	0.065492577676395800	Heart Diseases
Disease_D006323	0.062304203372209300	Heart Arrest
Disease_D012871	0.061696998322026500	Skin Diseases
Disease_D002178	0.060040036367452900	Candidiasis, Chronic Mucocutaneous
Disease_D003866	0.055268693183090600	Depressive Disorder
Disease_D000796	0.048161049235165900	Angiolymphoid Hyperplasia with Eosinophilia
Disease_D001943	0.047197318094728600	Breast Neoplasms
Disease_D018149	0.046606803837246500	Glucose Intolerance
Disease_D012163	0.043675416271926100	Retinal Detachment
Disease_D004487	0.042948492539427800	Edema
Disease_D002294	0.038486015034977900	Carcinoma, Squamous Cell
Disease_D013746	0.037357065596585500	Tetany
Disease_D003967	0.034729889161150700	Diarrhea
Disease_D000038	0.033880160205060700	Abscess
Disease_D007938	0.033717602805337500	Leukemia
Disease_D009260	0.033062631480676400	Nail Diseases
Disease_D005402	0.032387717568354300	Fistula
Disease_D010146	0.031450887475870600	Pain
Disease_D014456	0.031431604620914500	Ulcer
Disease_D014376	0.031264915507368700	Tuberculosis
Disease_D000740	0.030498825597766900	Anemia
Disease_D008659	0.028542944247126700	Metabolic Diseases
Disease_D003586	0.025992326129990900	Cytomegalovirus Infections
Disease_D002177	0.022625227518729600	Candidiasis
Disease_D014552	0.020895351026056500	Urinary Tract Infections
Disease_D011776	0.019681278083880400	Pyuria
Disease_D014987	0.018767918897289200	Xerostomia
Disease_D004802	0.018396297719345300	Eosinophilia
Disease_D010019	0.014324613682271200	Osteomyelitis
Disease_C566367	0.010052327823558200	Light Fixation Seizure Syndrome
Disease_D003110	0.009508292862788260	Colonic Neoplasms
Disease_D006223	0.009397917114059030	Hamartoma Syndrome, Multiple
Disease_D010190	0.008321612739809600	Pancreatic Neoplasms
Disease_D001660	0.006004807504687420	Biliary Tract Diseases
Disease_D004211	0.004905794185937070	Disseminated Intravascular Coagulation
Disease_C566273	0.004220164029946660	alpha-1-Antitrypsin Deficiency, Autosomal Recessive
Disease_D001791	-0.001887794453472090	Blood Platelet Disorders
Disease_C565742	-0.005116708489009650	Hyaluronan Metabolism, Defect in
Disease_D060737	-0.006818747629817080	Reproductive Tract Infections

ID	IC	Name
Disease_D018307	-0.009818077840822670	Neoplasms, Squamous Cell
Disease_D008175	-0.010353608454065800	Lung Neoplasms
Disease_D006069	-0.016806028067039400	Gonorrhea
Disease_D009336	-0.019033675323604500	Necrosis
Disease_C564973	-0.020499680049883200	Myopathy due to Malate-Aspartate Shuttle Defect
Disease_D003093	-0.021591886647895700	Colitis, Ulcerative
Disease_D006551	-0.023302888652986800	Hernia, Hiatal
Disease_D009103	-0.023667934494263500	Multiple Sclerosis
Disease_D002908	-0.026385915401217100	Chronic Disease
Disease_D051437	-0.031268401508325100	Renal Insufficiency
Disease_D019966	-0.032033474384521300	Substance-Related Disorders
Disease_D003876	-0.034257567293875800	Dermatitis, Atopic
Disease_D006470	-0.059251203696436600	Hemorrhage
Disease_D007634	-0.062277919261648000	Keratitis
Disease_D002277	-0.066712765310635500	Carcinoma
Disease_D007246	-0.080208089807874100	Infertility
Disease_D064420	-0.085386120020793300	Drug-Related Side Effects and Adverse Reactions
Disease_D009369	-0.086768987532377600	Neoplasms
Disease_D012421	-0.091060194597316600	Rupture
Disease_D012640	-0.092700781114316900	Seizures
Disease_D002690	-0.092847921372249100	Chlamydia Infections
Disease_D007410	-0.097142013350430500	Intestinal Diseases
Disease_D006461	-0.099621745094229300	Hemolysis
Disease_D060085	-0.100928130609846000	Coinfection
Disease_D003550	-0.101797415105162000	Cystic Fibrosis
Disease_D017093	-0.112361642308086000	Liver Failure
Disease_D004890	-0.147042749996819000	Erythema
Disease_D015431	-0.150537256296315000	Weight Loss
Disease_D030342	-0.175330088494539000	Genetic Diseases, Inborn

Appendix VI – Final CC Result in Experiment Two

ID	CC	Name
Disease_D016638	1.350126488	Critical Illness
Disease_D058365	1.32513689	Candidiasis, Invasive
Disease_D058387	1.313930559	Candidemia
Disease_D002177	1.313675349	Candidiasis
Disease_D010195	1.312786191	Pancreatitis
Disease_D015821	1.309977631	Eye Infections, Fungal
Disease_D002277	1.306686762	Carcinoma
Disease_D009877	1.306294747	Endophthalmitis
Disease_D020096	1.295726448	Zygomycosis
Disease_D018805	1.295391116	Sepsis
Disease_D009181	1.295182288	Mycoses
Disease_D018798	1.28989653	Anemia, Iron-Deficiency
Disease_D003110	1.285475248	Colonic Neoplasms
Disease_D019283	1.283878346	Pancreatitis, Acute Necrotizing
Disease_D015473	1.283307433	Leukemia, Promyelocytic, Acute
Disease_D003680	1.282857012	Deglutition Disorders
Disease_D009196	1.279424199	Myeloproliferative Disorders
Disease_D014008	1.278808952	Tinea Pedis
Disease_D007239	1.278224088	Infection
Disease_D008171	1.277859771	Lung Diseases
Disease_D016720	1.275446486	Pneumocystis Infections
Disease_D014860	1.275037901	Warts
Disease_C565469	1.274978553	Immune Deficiency Disease
Disease_D009190	1.27497021	Myelodysplastic Syndromes
Disease_D010538	1.274557774	Peritonitis
Disease_D015658	1.274261455	HIV Infections
Disease_D001471	1.274097848	Barrett Esophagus
Disease_D007246	1.273247195	Infertility
Disease_D011020	1.273028772	Pneumonia, Pneumocystis
Disease_D009091	1.272649483	Mucormycosis
Disease_D004211	1.272041591	Disseminated Intravascular Coagulation
Disease_D009503	1.271982323	Neutropenia
Disease_D014376	1.269510456	Tuberculosis
Disease_D003920	1.269502404	Diabetes Mellitus
Disease_D016919	1.269172284	Meningitis, Cryptococcal
Disease_D011655	1.268880823	Pulmonary Embolism
Disease_D007565	1.268003193	Jaundice
Disease_D008581	1.267647328	Meningitis
Disease_D054198	1.267092207	Precursor Cell Lymphoblastic Leukemia-Lymphoma
Disease_D052016	1.264482707	Mucositis

ID	CC	Name
Disease_D004941	1.264300967	Esophagitis
Disease_D004696	1.263324188	Endocarditis
Disease_D009410	1.262352116	Nerve Degeneration
Disease_D015470	1.261134004	Leukemia, Myeloid, Acute
Disease_D010532	1.260963408	Peritoneal Diseases
Disease_D019337	1.260267466	Hematologic Neoplasms
Disease_D006258	1.259417676	Head and Neck Neoplasms
Disease_C567712	1.259314705	Retinitis Pigmentosa, Concentric
Disease_D005764	1.259159447	Gastroesophageal Reflux
Disease_D006849	1.259093296	Hydrocephalus
Disease_D001228	1.258940988	Aspergillosis
Disease_D005334	1.25815027	Fever
Disease_D056650	1.257828388	Vulvodynia
Disease_D003645	1.25763912	Death, Sudden
Disease_D059413	1.257181172	Intraabdominal Infections
Disease_D006331	1.256724999	Heart Diseases
Disease_D001424	1.256505498	Bacterial Infections
Disease_C535342	1.255763487	Cataract, zonular
Disease_C535590	1.254871787	Carrington syndrome
Disease_D009369	1.254792483	Neoplasms
Disease_D010304	1.253684553	Paronychia
Disease_D011014	1.253667603	Pneumonia
Disease_D007640	1.253353029	Keratoconus
Disease_D016469	1.251906139	Fungemia
Disease_D008223	1.251418054	Lymphoma
Disease_D000163	1.249519206	Acquired Immunodeficiency Syndrome
Disease_D015179	1.248415076	Colorectal Neoplasms
Disease_D009325	1.24834619	Nausea
Disease_D006192	1.247953079	Haemophilus Infections
Disease_D056486	1.247668923	Chemical and Drug Induced Liver Injury
Disease_D010212	1.247668698	Papilloma
Disease_D010146	1.246498186	Pain
Disease_D007634	1.245719064	Keratitis
Disease_D009959	1.245215633	Oropharyngeal Neoplasms
Disease_D008103	1.244326518	Liver Cirrhosis
Disease_D016470	1.244137178	Bacteremia
Disease_D007674	1.242443433	Kidney Diseases
Disease_D009260	1.242349075	Nail Diseases
Disease_D006939	1.242168097	Hyperemesis Gravidarum
Disease_D014245	1.242151255	Trichomonas Infections
Disease_D012749	1.241032631	Sexually Transmitted Diseases
Disease_D003316	1.240666936	Corneal Diseases
Disease_D012131	1.238971308	Respiratory Insufficiency

ID	CC	Name
Disease_D006069	1.238463866	Gonorrhea
Disease_D007970	1.237961203	Leukopenia
Disease_D013568	1.237507784	Pathological Conditions, Signs and Symptoms
Disease_D029424	1.23734676	Pulmonary Disease, Chronic Obstructive
Disease_D034721	1.237220338	Mastocytosis, Systemic
Disease_D010518	1.237010064	Periodontitis
Disease_D003586	1.235956876	Cytomegalovirus Infections
Disease_D003881	1.235535842	Dermatomycoses
Disease_D010019	1.235066992	Osteomyelitis
Disease_D002821	1.23475908	Chorioamnionitis
Disease_D014689	1.234111877	Venous Insufficiency
Disease_D009894	1.234023955	Opportunistic Infections
Disease_D010585	1.233941433	Phagocyte Bactericidal Dysfunction
Disease_D005891	1.233702996	Gingivitis
Disease_D002825	1.233617041	Chorioretinitis
Disease_D005767	1.233244681	Gastrointestinal Diseases
Disease_D008206	1.233202197	Lymphatic Diseases
Disease_D006967	1.232592435	Hypersensitivity
Disease_D007794	1.23084086	Lameness, Animal
Disease_D055499	1.230013878	Catheter-Related Infections
Disease_D002180	1.229738465	Candidiasis, Oral
Disease_D013280	1.229694349	Stomatitis
Disease_D008107	1.229107301	Liver Diseases
Disease_D001523	1.228358793	Mental Disorders
Disease_C566419	1.227771185	Orofacial Cleft 2
Disease_D063646	1.226831037	Carcinogenesis
Disease_D010190	1.226378577	Pancreatic Neoplasms
Disease_D003453	1.225575687	Cryptococcosis
Disease_D008100	1.224451773	Liver Abscess
Disease_D001284	1.224125889	Atrophy
Disease_D006402	1.224103803	Hematologic Diseases
Disease_D010211	1.222522666	Papilledema
Disease_D014456	1.222069245	Ulcer
Disease_D004405	1.221325328	Dysentery, Bacillary
Disease_D013282	1.22110823	Stomatitis, Denture
Disease_C536777	1.220975658	Systemic candidiasis
Disease_D008228	1.220962173	Lymphoma, Non-Hodgkin
Disease_D003866	1.220392075	Depressive Disorder
Disease_D016109	1.220084303	Epidermolysis Bullosa, Junctional
Disease_D012772	1.219396962	Shock, Septic
Disease_D009062	1.218574973	Mouth Neoplasms
Disease_D003876	1.218270281	Dermatitis, Atopic
Disease_D009164	1.218206286	Mycobacterium Infections

ID	CC	Name
Disease_D007410	1.217786677	Intestinal Diseases
Disease_C531782	1.217629194	Endemic treponematosis caused by Treponema carateum
Disease_D016399	1.217623286	Lymphoma, T-Cell
Disease_D003967	1.217534386	Diarrhea
Disease_D014848	1.217140684	Vulvovaginitis
Disease_D009057	1.216953334	Stomatognathic Diseases
Disease_D005355	1.216771459	Fibrosis
Disease_D013281	1.216735859	Stomatitis, Aphthous
Disease_D003428	1.216305339	Cross Infection
Disease_D018458	1.216142427	Persistent Vegetative State
Disease_D007008	1.215935586	Hypokalemia
Disease_D005402	1.215770267	Fistula
Disease_D007249	1.215340915	Inflammation
Disease_D001022	1.215276468	Aortic Valve Insufficiency
Disease_D014564	1.214536829	Urogenital Abnormalities
Disease_D003731	1.213141416	Dental Caries
Disease_D004487	1.21311292	Edema
Disease_D012640	1.212996336	Seizures
Disease_D001927	1.212679423	Brain Diseases
Disease_D001791	1.211983352	Blood Platelet Disorders
Disease_D003677	1.211876339	Deficiency Diseases
Disease_D008583	1.21185983	Meningitis, Haemophilus
Disease_C566808	1.211713834	Phagocytosis, Plasma-Related Defect in
Disease_D015863	1.211559018	Iridocyclitis
Disease_D011537	1.211109977	Pruritus
Disease_D001261	1.210852721	Pulmonary Atelectasis
Disease_C531821	1.210840328	Stenotrophomonas maltophilia bacteremia
Disease_D001943	1.210606571	Breast Neoplasms
Disease_D009422	1.210518224	Nervous System Diseases
Disease_C566367	1.20991228	Light Fixation Seizure Syndrome
Disease_D002294	1.20950814	Carcinoma, Squamous Cell
Disease_D007153	1.208540935	Immunologic Deficiency Syndromes
Disease_D000230	1.208228546	Adenocarcinoma
Disease_D018307	1.20736708	Neoplasms, Squamous Cell
Disease_D018792	1.207365989	Encephalitis, Viral
Disease_D012376	1.207290107	Rodent Diseases
Disease_D016585	1.206608225	Vaginosis, Bacterial
Disease_D003072	1.206213861	Cognition Disorders
Disease_D003643	1.205992613	Death
Disease_D008175	1.205832364	Lung Neoplasms
Disease_D001249	1.20580594	Asthma
Disease_D012480	1.205320101	Salmonella Infections
Disease_D002179	1.204451506	Candidiasis, Cutaneous

ID	CC	Name
Disease_D005128	1.203556702	Eye Diseases
Disease_D006223	1.203508237	Hamartoma Syndrome, Multiple
Disease_D016778	1.2033537	Malaria, Falciparum
Disease_D018149	1.203328393	Glucose Intolerance
Disease_D003424	1.202875655	Crohn Disease
Disease_D008288	1.202802563	Malaria
Disease_D004890	1.202569084	Erythema
Disease_D010505	1.202340568	Familial Mediterranean Fever
Disease_D003872	1.201431224	Dermatitis
Disease_D014777	1.201112215	Virus Diseases
Disease_D002908	1.200860465	Chronic Disease
Disease_D015746	1.200382091	Abdominal Pain
Disease_D014005	1.200280309	Tinea
Disease_D006099	1.199631208	Granuloma
Disease_D012871	1.199600501	Skin Diseases
Disease_D015299	1.198573701	Discitis
Disease_D064420	1.198488606	Drug-Related Side Effects and Adverse Reactions
Disease_D009402	1.19770869	Nephrosis, Lipoid
Disease_D000740	1.197484777	Anemia
Disease_D014627	1.197326401	Vaginitis
Disease_D014009	1.196924736	Onychomycosis
Disease_D012769	1.196767583	Shock
Disease_D006323	1.196764373	Heart Arrest
Disease_D004802	1.196239899	Eosinophilia
Disease_C564973	1.194977141	Myopathy due to Malate-Aspartate Shuttle Defect
Disease_D009175	1.193520901	Mycoplasma Infections
Disease_D004401	1.193215212	Dysarthria
Disease_C565957	1.192469078	Amyotrophic Lateral Sclerosis 2, Juvenile
Disease_D000038	1.191750256	Abscess
Disease_D014987	1.191347103	Xerostomia
Disease_D014947	1.190984754	Wounds and Injuries
Disease_D016715	1.188114357	Proteus Syndrome
Disease_D007154	1.187966582	Immune System Diseases
Disease_D017093	1.187895481	Liver Failure
Disease_D003092	1.185984417	Colitis
Disease_D013771	1.185702696	Tetralogy of Fallot
Disease_D006105	1.184757611	Granulomatous Disease, Chronic
Disease_D009800	1.183834223	Oculocerebrorenal Syndrome
Disease_D002181	1.183477161	Candidiasis, Vulvovaginal
Disease_D011552	1.183138805	Pseudomonas Infections
Disease_D009765	1.181890137	Obesity
Disease_D001327	1.181822905	Autoimmune Diseases
Disease_D006333	1.181681138	Heart Failure

ID	CC	Name
Disease_D028361	1.181521864	Mitochondrial Diseases
Disease_D058565	1.181332058	Cerebral Ventriculitis
Disease_D009103	1.181056185	Multiple Sclerosis
Disease_D060085	1.181025444	Coinfection
Disease_D001660	1.180382867	Biliary Tract Diseases
Disease_D013927	1.180337371	Thrombosis
Disease_C535390	1.179057067	Aspergillus niger infection
Disease_D012163	1.178708594	Retinal Detachment
Disease_D009336	1.178351921	Necrosis
Disease_D013180	1.178289644	Sprains and Strains
Disease_D003141	1.177664966	Communicable Diseases
Disease_D053717	1.177507787	Pneumonia, Ventilator-Associated
Disease_D002761	1.17690185	Cholangitis
Disease_D010493	1.176060921	Pericarditis
Disease_C565742	1.175663966	Hyaluronan Metabolism, Defect in
Disease_D017827	1.172830956	Machado-Joseph Disease
Disease_D014839	1.172820924	Vomiting
Disease_D001765	1.172585486	Blind Loop Syndrome
Disease_D009362	1.17249233	Neoplasm Metastasis
Disease_D006470	1.172326656	Hemorrhage
Disease_D015835	1.171586922	Ocular Motility Disorders
Disease_D004927	1.171556832	Escherichia coli Infections
Disease_D060737	1.170257252	Reproductive Tract Infections
Disease_D055732	1.170172953	Pulmonary Aspergillosis
Disease_D009135	1.169760022	Muscular Diseases
Disease_D002972	1.168838484	Cleft Palate
Disease_D051437	1.168769605	Renal Insufficiency
Disease_D000796	1.168319025	Angiolymphoid Hyperplasia with Eosinophilia
Disease_D020766	1.168017772	Intracranial Embolism
Disease_D008607	1.16465357	Intellectual Disability
Disease_D007710	1.160318517	Klebsiella Infections
Disease_D001437	1.159899863	Bacteriuria
Disease_D006461	1.159899055	Hemolysis
Disease_C536972	1.159086898	Torulopsis
Disease_D014006	1.15814673	Tinea Capitis
Disease_D018410	1.156672367	Pneumonia, Bacterial
Disease_D005955	1.154980696	Glucosephosphate Dehydrogenase Deficiency
Disease_D001855	1.153890658	Bone Marrow Diseases
Disease_D015212	1.152745551	Inflammatory Bowel Diseases
Disease_D004342	1.149738426	Drug Hypersensitivity
Disease_D012421	1.149686117	Rupture
Disease_D019966	1.148389061	Substance-Related Disorders
Disease_D017676	1.145533117	Lichen Planus, Oral

ID	CC	Name
Disease_C565043	1.145095179	Arene Oxide Detoxification Defect
Disease_D030342	1.143125594	Genetic Diseases, Inborn
Disease_D008659	1.142136968	Metabolic Diseases
Disease_D014123	1.136653275	Toxoplasmosis
Disease_D015817	1.134451536	Eye Infections
Disease_D013746	1.131511897	Tetany
Disease_D013203	1.130404699	Staphylococcal Infections
Disease_D004673	1.128814016	Encephalomyelitis, Acute Disseminated
Disease_D007752	1.126393202	Obstetric Labor, Premature
Disease_C565534	1.126296856	Granulomatous Disease with Defect in Neutrophil Chemotaxis
Disease_D015431	1.122213186	Weight Loss
Disease_D002178	1.122199709	Candidiasis, Chronic Mucocutaneous
Disease_D009436	1.120524601	Neural Tube Defects
Disease_D002690	1.115728362	Chlamydia Infections
Disease_D014552	1.113682005	Urinary Tract Infections
Disease_C566273	1.112808878	alpha-1-Antitrypsin Deficiency, Autosomal Recessive
Disease_D006944	1.11245515	Hyperglycemic Hyperosmolar Nonketotic Coma
Disease_D003550	1.109769579	Cystic Fibrosis
Disease_D003093	1.107429663	Colitis, Ulcerative
Disease_D005928	1.093094035	Glossitis
Disease_C569516	1.091197648	Trichophyton infection
Disease_D004931	1.088749181	Esophageal Achalasia
Disease_D004194	1.088569359	Disease
Disease_D014010	1.086530392	Tinea Versicolor
Disease_D007938	1.079613705	Leukemia
Disease_D011776	1.070948145	Pyuria
Disease_211750	1.066694513	C SYNDROME
Disease_D008180	0.988735149	Lupus Erythematosus, Systemic
Disease_D006551	0.899521962	Hernia, Hiatal

Appendix VII -- Final SI Result in Experiment Two

ID	Concept	SI	Name	Categories
1	Disease_D001471	0.768272568	Barrett Esophagus	Cancer | Digestive system disease
2	Disease_D003645	0.766064234	Death, Sudden	Pathology (process)
3	Disease_D014689	0.712235519	Venous Insufficiency	Cardiovascular disease
4	Disease_D007794	0.688346988	Lameness, Animal	Animal disease
5	Disease_D010505	0.672027439	Familial Mediterranean Fever	Genetic disease (inborn)
6	Disease_D006192	0.646732547	Haemophilus Infections	Bacterial infection or mycosis
7	Disease_C535342	0.633730526	Cataract, zonular	Eye disease
8	Disease_D007640	0.599216211	Keratoconus	Eye disease
9	Disease_D058565	0.588317644	Cerebral Ventriculitis	Nervous system disease
10	Disease_D009410	0.570225406	Nerve Degeneration	Pathology (process)
11	Disease_D004405	0.523159095	Dysentery, Bacillary	Bacterial infection or mycosis | Digestive system disease
12	Disease_D012480	0.522955819	Salmonella Infections	Bacterial infection or mycosis
13	Disease_D007565	0.518487636	Jaundice	Pathology (process) | Signs and symptoms
14	Disease_D010195	0.507900372	Pancreatitis	Digestive system disease
15	Disease_D010211	0.498821664	Papilledema	Eye disease | Nervous system disease
16	Disease_D009196	0.496670098	Myeloproliferative Disorders	Blood disease
17	Disease_D008228	0.49550053	Lymphoma, Non-Hodgkin	Cancer | Immune system disease | Lymphatic disease
18	Disease_D008223	0.49057365	Lymphoma	Cancer | Immune system disease | Lymphatic disease
19	Disease_D009190	0.486614258	Myelodysplastic Syndromes	Blood disease
20	Disease_D001284	0.476436141	Atrophy	Pathology (anatomical condition)
21	Disease_D012376	0.474968596	Rodent Diseases	Animal disease
22	Disease_D006939	0.463296915	Hyperemesis Gravidarum	Pregnancy complication | Signs and symptoms
23	Disease_D003680	0.452781389	Deglutition Disorders	Digestive system disease | Ear-nose-throat disease
24	Disease_C567712	0.444983528	Retinitis Pigmentosa, Concentric	Eye disease | Genetic disease (inborn)
25	Disease_D054198	0.442113323	Precursor Cell Lymphoblastic Leukemia-Lymphoma	Cancer | Immune system disease | Lymphatic disease
26	Disease_C535390	0.439100078	Aspergillus niger infection	Bacterial infection or mycosis | Respiratory tract disease
27	Disease_D007008	0.433724889	Hypokalemia	Metabolic disease

28	Disease_D000230	0.398272709	Adenocarcinoma	Cancer
29	Disease_D010538	0.38915094	Peritonitis	Bacterial infection or mycosis | Digestive system disease
30	Disease_D010532	0.384690112	Peritoneal Diseases	Digestive system disease
31	Disease_D015863	0.383757066	Iridocyclitis	Eye disease
32	Disease_C535590	0.382048765	Carrington syndrome	Blood disease | Respiratory tract disease
33	Disease_D009402	0.371534496	Nephrosis, Lipoid	Urogenital disease (female) | Urogenital disease (male)
34	Disease_D009877	0.371327579	Endophthalmitis	Bacterial infection or mycosis | Eye disease
35	Disease_D019337	0.359488904	Hematologic Neoplasms	Blood disease | Cancer
36	Disease_D004401	0.355648045	Dysarthria	Nervous system disease | Signs and symptoms
37	Disease_D016638	0.352349534	Critical Illness	Pathology (process)
38	Disease_D006849	0.3454902	Hydrocephalus	Nervous system disease
39	Disease_D015821	0.344349076	Eye Infections, Fungal	Bacterial infection or mycosis | Eye disease
40	Disease_D015658	0.339859232	HIV Infections	Immune system disease | Viral disease
41	Disease_D015299	0.338366415	Discitis	Bacterial infection or mycosis | Musculoskeletal disease
42	Disease_D010585	0.333902879	Phagocyte Bactericidal Dysfunction	Blood disease | Immune system disease
43	Disease_D006258	0.325093872	Head and Neck Neoplasms	Cancer
44	Disease_D058387	0.312648066	Candidemia	Bacterial infection or mycosis | Pathology (process)
45	Disease_D009325	0.306463689	Nausea	Signs and symptoms
46	Disease_D014008	0.303984389	Tinea Pedis	Bacterial infection or mycosis | Signs and symptoms | Skin disease
47	Disease_C565469	0.303772644	Immune Deficiency Disease	Immune system disease
48	Disease_D058365	0.296300228	Candidiasis, Invasive	Bacterial infection or mycosis
49	Disease_D009503	0.296036597	Neutropenia	Blood disease
50	Disease_D005764	0.291272415	Gastroesophageal Reflux	Digestive system disease
51	Disease_D015746	0.285988007	Abdominal Pain	Signs and symptoms
52	Disease_D002821	0.280234767	Chorioamnionitis	Fetal disease | Pregnancy complication
53	Disease_D019283	0.27773144	Pancreatitis, Acute Necrotizing	Digestive system disease
54	Disease_D016585	0.266204788	Vaginosis, Bacterial	Bacterial infection or mycosis | Urogenital disease (female)
55	Disease_C566808	0.265351446	Phagocytosis, Plasma-Related Defect in	Blood disease
56	Disease_D014006	0.265068384	Tinea Capitis	Bacterial infection or mycosis | Skin disease
57	Disease_D015470	0.258148777	Leukemia, Myeloid, Acute	Cancer

58	Disease_D011655	0.251766449	Pulmonary Embolism	Cardiovascular disease | Respiratory tract disease
59	Disease_D016715	0.251507817	Proteus Syndrome	Cancer | Congenital abnormality | Musculoskeletal disease
60	Disease_D015835	0.246582656	Ocular Motility Disorders	Eye disease | Nervous system disease
61	Disease_D056650	0.246441033	Vulvodynia	Urogenital disease (female)
62	Disease_D006967	0.239596406	Hypersensitivity	Immune system disease
63	Disease_D014245	0.238593031	Trichomonas Infections	Parasitic disease
64	Disease_D015179	0.229584941	Colorectal Neoplasms	Cancer | Digestive system disease
65	Disease_D018805	0.229298665	Sepsis	Bacterial infection or mycosis | Pathology (process)
66	Disease_C531821	0.228158899	Stenotrophomonas maltophilia bacteremia	Bacterial infection or mycosis
67	Disease_D003731	0.220916768	Dental Caries	Mouth disease
68	Disease_D016919	0.217786085	Meningitis, Cryptococcal	Bacterial infection or mycosis | Nervous system disease
69	Disease_D000163	0.214366691	Acquired Immunodeficiency Syndrome	Immune system disease | Viral disease
70	Disease_D004696	0.212820264	Endocarditis	Cardiovascular disease
71	Disease_D005891	0.209250745	Gingivitis	Mouth disease
72	Disease_D003424	0.207699087	Crohn Disease	Digestive system disease
73	Disease_D001765	0.20494085	Blind Loop Syndrome	Digestive system disease | Metabolic disease
74	Disease_D009175	0.203478256	Mycoplasma Infections	Bacterial infection or mycosis
75	Disease_D005334	0.202515787	Fever	Signs and symptoms
76	Disease_D001523	0.202110639	Mental Disorders	Mental disorder
77	Disease_D011552	0.202087203	Pseudomonas Infections	Bacterial infection or mycosis
78	Disease_D015473	0.199415456	Leukemia, Promyelocytic, Acute	Cancer
79	Disease_D018792	0.19936207	Encephalitis, Viral	Nervous system disease | Viral disease
80	Disease_D009164	0.194105827	Mycobacterium Infections	Bacterial infection or mycosis
81	Disease_D014009	0.192731903	Onychomycosis	Bacterial infection or mycosis | Skin disease
82	Disease_D008581	0.192401752	Meningitis	Nervous system disease
83	Disease_D016778	0.190900343	Malaria, Falciparum	Parasitic disease
84	Disease_D004941	0.18781518	Esophagitis	Digestive system disease
85	Disease_D004927	0.186854006	Escherichia coli Infections	Bacterial infection or mycosis
86	Disease_D020096	0.184678167	Zygomycosis	Bacterial infection or mycosis
87	Disease_D014947	0.183584307	Wounds and Injuries	Wounds and injuries
88	Disease_D016720	0.182368219	Pneumocystis Infections	Bacterial infection or mycosis

89	Disease_D008583	0.177240353	Meningitis, Haemophilus	Bacterial infection or mycosis | Nervous system disease
90	Disease_D013281	0.17121419	Stomatitis, Aphthous	Mouth disease
91	Disease_D018798	0.170557723	Anemia, Iron-Deficiency	Blood disease | Metabolic disease
92	Disease_D011014	0.155859752	Pneumonia	Respiratory tract disease
93	Disease_D013280	0.153988196	Stomatitis	Mouth disease
94	Disease_D014627	0.149374427	Vaginitis	Urogenital disease (female)
95	Disease_D016469	0.14846815	Fungemia	Bacterial infection or mycosis | Pathology (process)
96	Disease_D003453	0.147713889	Cryptococcosis	Bacterial infection or mycosis
97	Disease_D009181	0.147427617	Mycoses	Bacterial infection or mycosis
98	Disease_D009057	0.147187215	Stomatognathic Diseases	Mouth disease
99	Disease_D053717	0.139244627	Pneumonia, Ventilator-Associated	Bacterial infection or mycosis | Respiratory tract disease
100	Disease_D012131	0.137691841	Respiratory Insufficiency	Respiratory tract disease
101	Disease_D007249	0.136160698	Inflammation	Pathology (process)
102	Disease_D011537	0.134861703	Pruritus	Signs and symptoms | Skin disease
103	Disease_C566419	0.134416437	Orofacial Cleft 2	Congenital abnormality | Mouth disease | Musculoskeletal disease
104	Disease_D009062	0.132535535	Mouth Neoplasms	Cancer | Mouth disease
105	Disease_D003920	0.131189911	Diabetes Mellitus	Endocrine system disease | Metabolic disease
106	Disease_D008206	0.127220709	Lymphatic Diseases	Lymphatic disease
107	Disease_D006105	0.125836291	Granulomatous Disease, Chronic	Blood disease | Genetic disease (inborn) | Immune system disease
108	Disease_D010493	0.12549924	Pericarditis	Cardiovascular disease
109	Disease_D008171	0.121386025	Lung Diseases	Respiratory tract disease
110	Disease_D009091	0.11563551	Mucormycosis	Bacterial infection or mycosis
111	Disease_D005955	0.113328838	Glucosephosphate Dehydrogenase Deficiency	Blood disease | Genetic disease (inborn) | Metabolic disease
112	Disease_D008107	0.112405783	Liver Diseases	Digestive system disease
113	Disease_D003428	0.109290335	Cross Infection	Bacterial infection or mycosis | Pathology (process)
114	Disease_D002177	0.107186137	Candidiasis	Bacterial infection or mycosis
115	Disease_D010304	0.104250148	Paronychia	Bacterial infection or mycosis | Skin disease
116	Disease_D001424	0.099437239	Bacterial Infections	Bacterial infection or mycosis
117	Disease_D001228	0.095109731	Aspergillosis	Bacterial infection or mycosis
118	Disease_D004673	0.093992065	Encephalomyelitis, Acute Disseminated	Immune system disease | Nervous system disease

119	Disease_D007239	0.093798291	Infection	Bacterial infection or mycosis
120	Disease_D052016	0.091142997	Mucositis	Digestive system disease | Mouth disease
121	Disease_D014860	0.090678185	Warts	Skin disease | Viral disease
122	Disease_D003643	0.08876838	Death	Pathology (process)
123	Disease_D001927	0.087967909	Brain Diseases	Nervous system disease
124	Disease_D008103	0.087859355	Liver Cirrhosis	Digestive system disease
125	Disease_D006402	0.085753976	Hematologic Diseases	Blood disease
126	Disease_D017827	0.084541821	Machado-Joseph Disease	Genetic disease (inborn) | Nervous system disease
127	Disease_D003316	0.083578322	Corneal Diseases	Eye disease
128	Disease_D007710	0.078599249	Klebsiella Infections	Bacterial infection or mycosis
129	Disease_D010212	0.077353125	Papilloma	Cancer
130	Disease_D009894	0.075686608	Opportunistic Infections	Bacterial infection or mycosis | Parasitic disease | Viral disease
131	Disease_D005767	0.074377527	Gastrointestinal Diseases	Digestive system disease
132	Disease_D010518	0.071180197	Periodontitis	Mouth disease
133	Disease_D056486	0.06967854	Chemical and Drug Induced Liver Injury	Digestive system disease
134	Disease_D013771	0.069500378	Tetralogy of Fallot	Cardiovascular disease | Congenital abnormality
135	Disease_D011020	0.06847028	Pneumonia, Pneumocystis	Bacterial infection or mycosis | Respiratory tract disease
136	Disease_D009959	0.068397585	Oropharyngeal Neoplasms	Cancer | Ear-nose-throat disease | Mouth disease
137	Disease_D003881	0.065655858	Dermatomycoses	Bacterial infection or mycosis | Skin disease
138	Disease_D034721	0.064949023	Mastocytosis, Systemic	Cancer | Immune system disease
139	Disease_D063646	0.06296329	Carcinogenesis	Cancer | Pathology (process)
140	Disease_D001437	0.062912489	Bacteriuria	Bacterial infection or mycosis | Urogenital disease (female) | Urogenital disease (male)
141	Disease_D001249	0.062741952	Asthma	Immune system disease | Respiratory tract disease
142	Disease_D055732	0.062115187	Pulmonary Aspergillosis	Bacterial infection or mycosis | Respiratory tract disease
143	Disease_D016470	0.061524309	Bacteremia	Bacterial infection or mycosis | Pathology (process)
144	Disease_D002825	0.060880776	Chorioretinitis	Eye disease
145	Disease_D055499	0.0594625	Catheter-Related Infections	Bacterial infection or mycosis
146	Disease_D002179	0.058976373	Candidiasis, Cutaneous	Bacterial infection or mycosis | Skin disease
147	Disease_D008288	0.058240085	Malaria	Parasitic disease
148	Disease_D012749	0.05000266	Sexually Transmitted Diseases	Bacterial infection or mycosis | Urogenital disease (female) | Urogenital disease (male) | Viral disease

149	Disease_D059413	0.049292948	Intraabdominal Infections	Bacterial infection or mycosis
150	Disease_C565957	0.046097938	Amyotrophic Lateral Sclerosis 2, Juvenile	Metabolic disease | Nervous system disease
151	Disease_D014564	0.045368543	Urogenital Abnormalities	Congenital abnormality | Urogenital disease (female) | Urogenital disease (male)
152	Disease_D003110	0.03214074	Colonic Neoplasms	Cancer | Digestive system disease
153	Disease_D012769	0.030338862	Shock	Pathology (process)
154	Disease_D018458	0.02843574	Persistent Vegetative State	Nervous system disease | Signs and symptoms
155	Disease_D013568	0.026753905	Pathological Conditions, Signs and Symptoms	
156	Disease_D007674	0.025484559	Kidney Diseases	Urogenital disease (female) | Urogenital disease (male)
157	Disease_D005355	0.02325692	Fibrosis	Pathology (process)
158	Disease_D020766	0.02305037	Intracranial Embolism	Cardiovascular disease | Nervous system disease
159	Disease_D006331	0.02152843	Heart Diseases	Cardiovascular disease
160	Disease_D014376	0.017382276	Tuberculosis	Bacterial infection or mycosis
161	Disease_D016399	0.015637571	Lymphoma, T-Cell	Cancer | Immune system disease | Lymphatic disease
162	Disease_D028361	0.014552229	Mitochondrial Diseases	Metabolic disease
163	Disease_D002180	0.012094441	Candidiasis, Oral	Bacterial infection or mycosis | Mouth disease
164	Disease_D016109	0.011603754	Epidermolysis Bullosa, Junctional	Congenital abnormality | Genetic disease (inborn) | Skin disease
165	Disease_D004931	0.00969486	Esophageal Achalasia	Digestive system disease
166	Disease_D017676	0.007031985	Lichen Planus, Oral	Mouth disease | Skin disease
167	Disease_D002277	0.006795557	Carcinoma	Cancer
168	Disease_D029424	0.00607729	Pulmonary Disease, Chronic Obstructive	Respiratory tract disease
169	Disease_D003092	0.004248983	Colitis	Digestive system disease
170	Disease_C565043	0.002318754	Arene Oxide Detoxification Defect	Genetic disease (inborn) | Metabolic disease
171	Disease_D001261	0.00196058	Pulmonary Atelectasis	Respiratory tract disease
172	Disease_D006099	-0.000488835	Granuloma	Lymphatic disease | Pathology (process)
173	Disease_D007970	-0.001568256	Leukopenia	Blood disease
174	Disease_D014848	-0.001688262	Vulvovaginitis	Urogenital disease (female)
175	Disease_D004211	-0.002044677	Disseminated Intravascular Coagulation	Blood disease
176	Disease_D013282	-0.006023464	Stomatitis, Denture	Mouth disease
177	Disease_D013927	-0.007124636	Thrombosis	Cardiovascular disease
178	Disease_D003872	-0.009388848	Dermatitis	Skin disease

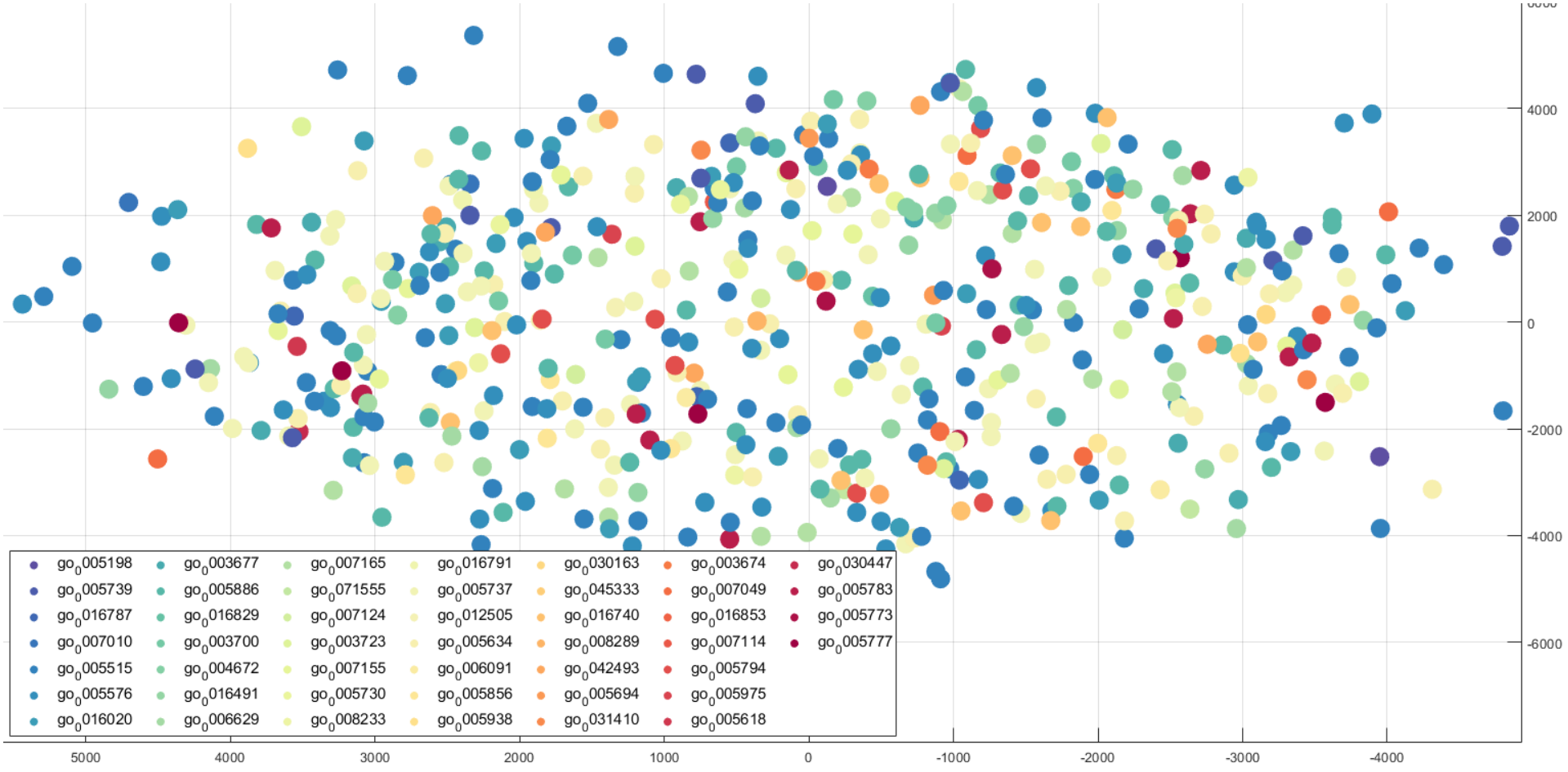
179	Disease_D003141	-0.012976501	Communicable Diseases	Bacterial infection or mycosis
180	Disease_D007154	-0.013937547	Immune System Diseases	Immune system disease
181	Disease_D001022	-0.017330618	Aortic Valve Insufficiency	Cardiovascular disease
182	Disease_D001327	-0.024802018	Autoimmune Diseases	Immune system disease
183	Disease_D008100	-0.032937852	Liver Abscess	Bacterial infection or mycosis | Digestive system disease
184	Disease_D010146	-0.033510799	Pain	Signs and symptoms
185	Disease_D003677	-0.033843893	Deficiency Diseases	Nutrition disorder
186	Disease_C536972	-0.033846999	Torulopsis	Bacterial infection or mycosis
187	Disease_D003072	-0.035661236	Cognition Disorders	Mental disorder
188	Disease_C536777	-0.039560538	Systemic candidiasis	Bacterial infection or mycosis
189	Disease_D009260	-0.041187338	Nail Diseases	Skin disease
190	Disease_C565534	-0.045989427	Granulomatous Disease with Defect in Neutrophil Chemotaxis	Blood disease | Genetic disease (inborn) | Immune system disease
191	Disease_D007153	-0.047804546	Immunologic Deficiency Syndromes	Immune system disease
192	Disease_D002761	-0.054865043	Cholangitis	Digestive system disease
193	Disease_C531782	-0.057140948	Endemic treponematosis caused by Treponema carateum	Bacterial infection or mycosis | Skin disease
194	Disease_D012772	-0.058093856	Shock, Septic	Bacterial infection or mycosis | Pathology (process)
195	Disease_D003586	-0.062090758	Cytomegalovirus Infections	Viral disease
196	Disease_D014777	-0.064104402	Virus Diseases	Viral disease
197	Disease_D008607	-0.068177189	Intellectual Disability	Mental disorder | Nervous system disease | Signs and symptoms
198	Disease_D003866	-0.068816947	Depressive Disorder	Mental disorder
199	Disease_D010019	-0.075151273	Osteomyelitis	Bacterial infection or mycosis | Musculoskeletal disease
200	Disease_D009800	-0.075424959	Oculocerebrorenal Syndrome	Congenital abnormality | Genetic disease (inborn) | Metabolic disease | Nervous system disease | Urogenital disease (female) | Urogenital disease (male)
201	Disease_D009362	-0.076001931	Neoplasm Metastasis	Cancer | Pathology (process)
202	Disease_D013180	-0.076784573	Sprains and Strains	Wounds and injuries
203	Disease_D005128	-0.078029438	Eye Diseases	Eye disease
204	Disease_D009422	-0.078098499	Nervous System Diseases	Nervous system disease

205	Disease_D007246	-0.080236978	Infertility	Urogenital disease (female) | Urogenital disease (male)
206	Disease_D006333	-0.086089301	Heart Failure	Cardiovascular disease
207	Disease_D018410	-0.087672939	Pneumonia, Bacterial	Bacterial infection or mycosis | Respiratory tract disease
208	Disease_D014456	-0.087742828	Ulcer	Pathology (process)
209	Disease_D014005	-0.087864203	Tinea	Bacterial infection or mycosis | Skin disease
210	Disease_D013203	-0.09051052	Staphylococcal Infections	Bacterial infection or mycosis
211	Disease_D006944	-0.09329805	Hyperglycemic Hyperosmolar Nonketotic Coma	Endocrine system disease
212	Disease_D003967	-0.094673026	Diarrhea	Signs and symptoms
213	Disease_D004487	-0.096676721	Edema	Signs and symptoms
214	Disease_D009135	-0.096701038	Muscular Diseases	Musculoskeletal disease | Nervous system disease
215	Disease_D006069	-0.097190427	Gonorrhea	Bacterial infection or mycosis | Urogenital disease (female) | Urogenital disease (male)
216	Disease_D001943	-0.098202048	Breast Neoplasms	Cancer | Skin disease
217	Disease_D010190	-0.10013649	Pancreatic Neoplasms	Cancer | Digestive system disease | Endocrine system disease
218	Disease_D005402	-0.100813359	Fistula	Pathology (anatomical condition)
219	Disease_D002181	-0.102475164	Candidiasis, Vulvovaginal	Bacterial infection or mycosis | Urogenital disease (female)
220	Disease_D007752	-0.106940847	Obstetric Labor, Premature	Pregnancy complication
221	Disease_D012871	-0.10885083	Skin Diseases	Skin disease
222	Disease_D002294	-0.108916452	Carcinoma, Squamous Cell	Cancer
223	Disease_D006323	-0.114567969	Heart Arrest	Cardiovascular disease
224	Disease_D018149	-0.114915135	Glucose Intolerance	Metabolic disease
225	Disease_D014839	-0.118130164	Vomiting	Signs and symptoms
226	Disease_D002972	-0.121172321	Cleft Palate	Congenital abnormality | Mouth disease | Musculoskeletal disease
227	Disease_D009765	-0.122066755	Obesity	Nutrition disorder | Signs and symptoms
228	Disease_D007634	-0.124292841	Keratitis	Eye disease
229	Disease_D009369	-0.127426026	Neoplasms	Cancer
230	Disease_C566367	-0.135034794	Light Fixation Seizure Syndrome	Congenital abnormality | Eye disease | Genetic disease (inborn) | Mental disorder | Nervous system disease | Signs and symptoms
231	Disease_D015817	-0.138181064	Eye Infections	Bacterial infection or mycosis | Eye disease
232	Disease_D001791	-0.141783055	Blood Platelet Disorders	Blood disease
233	Disease_D000740	-0.143187932	Anemia	Blood disease

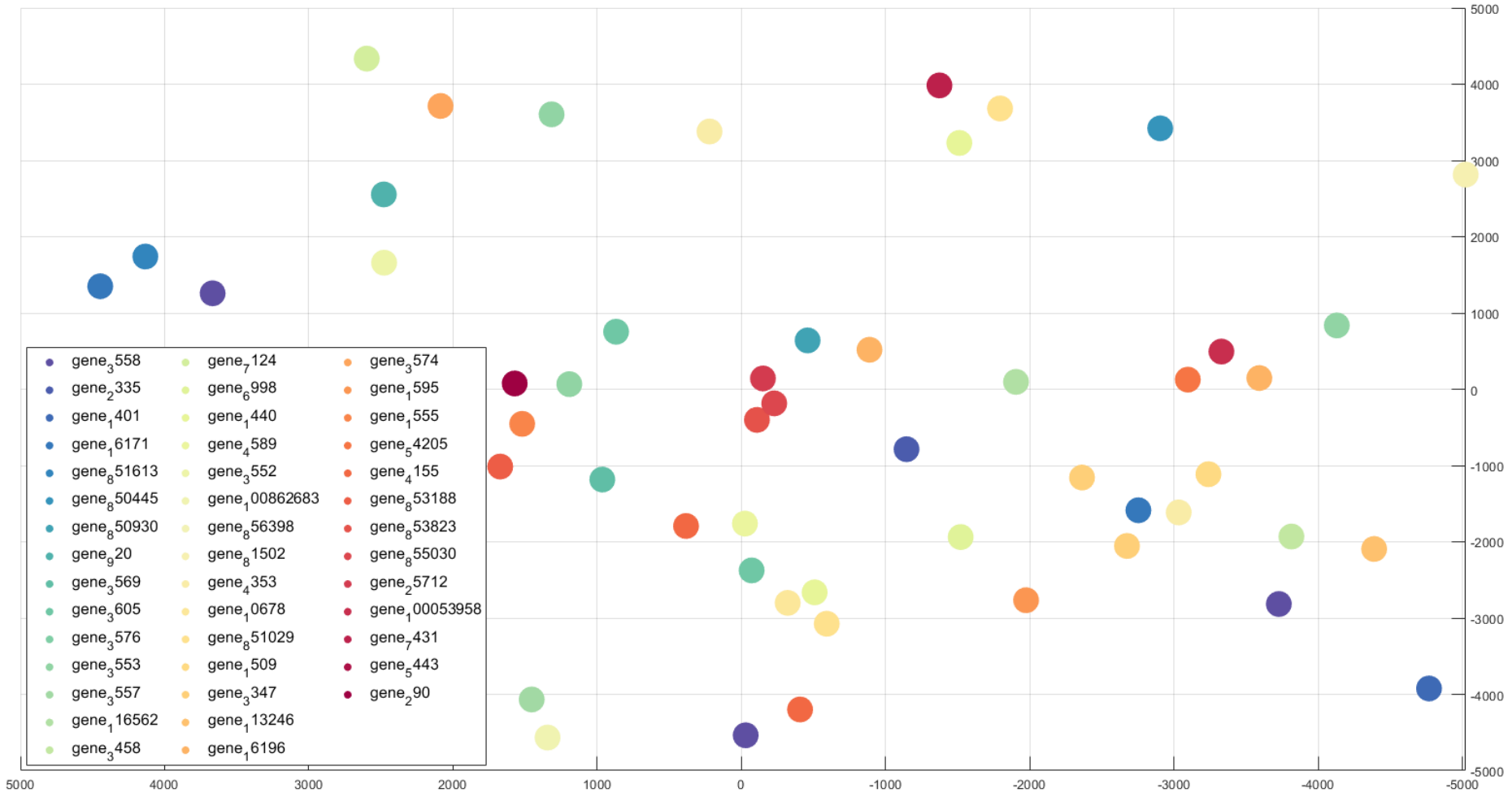
234	Disease_D006223	-0.149868672	Hamartoma Syndrome, Multiple	Cancer | Genetic disease (inborn)
235	Disease_D000038	-0.152701562	Abscess	Bacterial infection or mycosis | Pathology (process)
236	Disease_D004342	-0.157116888	Drug Hypersensitivity	Immune system disease
237	Disease_D004802	-0.157449387	Eosinophilia	Blood disease
238	Disease_D003876	-0.158585797	Dermatitis, Atopic	Genetic disease (inborn) | Immune system disease | Skin disease
239	Disease_D018307	-0.159562342	Neoplasms, Squamous Cell	Cancer
240	Disease_D008175	-0.163477059	Lung Neoplasms	Cancer | Respiratory tract disease
241	Disease_D014987	-0.167954595	Xerostomia	Mouth disease
242	Disease_D001855	-0.172128106	Bone Marrow Diseases	Blood disease
243	Disease_D012163	-0.172337548	Retinal Detachment	Eye disease
244	Disease_D015212	-0.187067549	Inflammatory Bowel Diseases	Digestive system disease
245	Disease_D002908	-0.189743406	Chronic Disease	Pathology (process)
246	Disease_D000796	-0.19113264	Angiolymphoid Hyperplasia with Eosinophilia	Blood disease | Lymphatic disease | Skin disease
247	Disease_C564973	-0.197207333	Myopathy due to Malate-Aspartate Shuttle Defect	Genetic disease (inborn) | Metabolic disease | Musculoskeletal disease | Nervous system disease
248	Disease_D001660	-0.204413279	Biliary Tract Diseases	Digestive system disease
249	Disease_D007410	-0.219406373	Intestinal Diseases	Digestive system disease
250	Disease_D014123	-0.222175826	Toxoplasmosis	Parasitic disease
251	Disease_C565742	-0.225452356	Hyaluronan Metabolism, Defect in	Genetic disease (inborn) | Metabolic disease | Skin disease
252	Disease_D012640	-0.225817607	Seizures	Nervous system disease | Signs and symptoms
253	Disease_D009103	-0.231111489	Multiple Sclerosis	Immune system disease | Nervous system disease
254	Disease_D009336	-0.232709829	Necrosis	Pathology (process)
255	Disease_D060737	-0.239068285	Reproductive Tract Infections	Bacterial infection or mycosis
256	Disease_D004194	-0.24056294	Disease	Pathology (process)
257	Disease_D064420	-0.251064012	Drug-Related Side Effects and Adverse Reactions	
258	Disease_D009436	-0.252570231	Neural Tube Defects	Congenital abnormality
259	Disease_D051437	-0.265599677	Renal Insufficiency	Urogenital disease (female) | Urogenital disease (male)
260	Disease_D008659	-0.267876361	Metabolic Diseases	Metabolic disease
261	Disease_D002178	-0.282196136	Candidiasis, Chronic Mucocutaneous	Bacterial infection or mycosis | Skin disease
262	Disease_D013746	-0.283081538	Tetany	Metabolic disease | Nervous system disease | Signs and symptoms

263	Disease_D006470	-0.284292559	Hemorrhage	Pathology (process)
264	Disease_D005928	-0.286040464	Glossitis	Mouth disease
265	Disease_C569516	-0.297818564	Trichophyton infection	Bacterial infection or mycosis
266	Disease_D017093	-0.300202511	Liver Failure	Digestive system disease
267	Disease_D004890	-0.300589229	Erythema	Signs and symptoms | Skin disease
268	Disease_D060085	-0.30458564	Coinfection	Bacterial infection or mycosis | Parasitic disease | Viral disease
269	Disease_D019966	-0.31155592	Substance-Related Disorders	Mental disorder
270	Disease_D014010	-0.318878158	Tinea Versicolor	Bacterial infection or mycosis | Skin disease
271	Disease_D008180	-0.326127247	Lupus Erythematosus, Systemic	Connective tissue disease | Immune system disease
272	Disease_D014552	-0.338290859	Urinary Tract Infections	Bacterial infection or mycosis | Urogenital disease (female) | Urogenital disease (male)
273	Disease_211750	-0.340261462	C SYNDROME	Congenital abnormality
274	Disease_D006461	-0.350228973	Hemolysis	Pathology (process)
275	Disease_C566273	-0.356071853	alpha-1-Antitrypsin Deficiency, Autosomal Recessive	Digestive system disease | Genetic disease (inborn) | Pathology (process) | Respiratory tract disease
276	Disease_D012421	-0.364759535	Rupture	Wounds and injuries
277	Disease_D003093	-0.392533995	Colitis, Ulcerative	Digestive system disease
278	Disease_D007938	-0.40171403	Leukemia	Cancer
279	Disease_D011776	-0.434281094	Pyuria	Bacterial infection or mycosis | Urogenital disease (female) | Urogenital disease (male)
280	Disease_D002690	-0.441818489	Chlamydia Infections	Bacterial infection or mycosis | Urogenital disease (female) | Urogenital disease (male)
281	Disease_D030342	-0.459384853	Genetic Diseases, Inborn	Genetic disease (inborn)
282	Disease_D003550	-0.463545507	Cystic Fibrosis	Digestive system disease | Genetic disease (inborn) | Infant-newborn disease | Respiratory tract disease
283	Disease_D015431	-0.482238513	Weight Loss	Signs and symptoms
284	Disease_D006551	-0.855556925	Hernia, Hiatal	Pathology (anatomical condition)

Appendix VIII -- t-SNE plot for the old approach


go0005198
go0005739
go0016787
go0007010
go0005515
go0005576
go0016020
go0003677
go0005886
go0016829
go0003700
go0004672
go0016491
go0006629
go0007165
go0071555
go0007124
go0003723
go0007155
go0005730
go0008233
go0016791
go0005737
go0012505
go0005634
go0006091
go0005856
go0005938
go0030163
go0045333
go0016740
go0008289
go0042493
go0005694
go0031410
go0003674
go0007049
go0016853
go0007114
go0005794
go0005975
go0005618
go0030447
go0005783
go0005773
go0005777
5000
4000
3000
2000
1000
0
-1000
-2000
-3000
-4000
4000
2000
0
-2000
-4000
-6000

Appendix IX – t-SNE plot for the new approach


gene3558
gene2335
gene1401
gene16171
gene851613
gene850445
gene850930
gene920
gene3569
gene3605
gene3576
gene3553
gene3557
gene116562
gene3458
gene7124
gene6998
gene1440
gene4589
gene3552
gene100862683
gene856398
gene81502
gene4353
gene10678
gene851029
gene1509
gene3347
gene113246
gene16196
gene3574
gene1595
gene1555
gene54205
gene4155
gene853188
gene853823
gene855030
gene25712
gene100053958
gene7431
gene5443
gene290
5000
4000
3000
2000
1000
0
-1000
-2000
-3000
-4000
-5000

Appendix X -- List of keywords associated with cancer category

systemic_illness
transformation_and_cloning_systems
systemic_diseases
breast_tumor
breast_cancer
proteus_vulgaris
proteus
proteus_and_fungi_of_the_genus_candida
oral_carcinoma
oral_candidosis
pancreatitis
nhl
malignant_lymphomas
malignant_lymphoma
lymphoid_leucosis
role_in_oral_carcinogenesis
barretts_esophagus
glandular_papilloma
chlamydia_and_human_papilloma
stbp
squamous_papillomas
tracheobronchial_papilloma
solitary_tracheobronchial_papilloma
papillomas
increased_lung_tumor
leukemia
leukaemia
acute_leukaemia
leukaemias
squamous_carcinoma
squamous_cell_hyperplasia
squamous_cell_carcinoma
colonization
deficient_t-cell_function
pulmonary_adenocarcinoma
carcinomas
submandibular_gland_carcinoma
carcinoma
head_and_neck_cancer

hematologic_malignancies
hematological_malignancy
hematopoietic
hematological_malignancies
reactive_squamous_metaplasia
cancer
neoplastic_disease
malignant_disease
non-oral_cancer
infection_and_malignancy
neutropenic_cancer
neoplasm
submucosal_tumors
neoplasms
malignancy
cem_neoplasms
cancer_of_the_cervix
intraluminal_tumors
oral_cancer
tumour
solid_tumors
malignant_tumour
non-neutropenic_cancer
febrile_cancer
tumor
oral_and_oesophageal_cancers
cancers
benign_tumor
malignancies
crown_gall_tumor
haematological_malignancy
invasive_aspergillosis
invasive_disease
submucosal_invasion
invasive_mucormycosis
fungal_invasion
invasive_mycosis
invasive_candidiasis
metastasis
ic
ia
oropharyngeal_candidiasis
oropharyngeal_candidosis
opc

vaginal_candida_colonization
colon_cancer
colonic_mycobiota
oral_candida_colonization
cd
acute_promyelocytic_leukemia
aml
promyelocytic_leukemia
acute_leukemia
acute_myelogenous_leukemia
acute_myeloblastic_leukemia
acute_nonlymphocytic_leukemia
acute_lymphoblastic_leukemia