A Review of Distributional Semantic Modelling (DSM) Methods in Natural Language Processing

James Barnden
mail@jamqes.com

Keywords: Distributional Semantics, Natural Language Processing, Feature Selection, Dimensionality Reduction, Computational Linguistics, Count-based semantic modelling, predictive semantic modelling.

Abstract

This paper explores DSM, its origins, its perusal across different disciplines such as psycholinguistics and neurolinguistics, with an emphasis on computational linguistics. The different types of and general process of applying DSM is explored, and experiments with different configurations and combinations of methods are reviewed and contrasted. The paper concludes with the analysis of the various configurations, and the strengths and weaknesses of the methods used to evaluate the experimental applications.

Introduction

Many tasks on language data in both commercial and academic settings often make use of one or more models constructed from some diverse, domain-relevant collection of text (or corpus). They are said to model various types of semantic relationships between words or documents in some mathematical space, whereby differences and similarities between them become measurable. These are useful in regression tasks, such as predicting the next word (or phoneme) in a sequence (used in predictive text systems on mobile devices, and in speech recognition), in query processing/information extraction, in classification tasks (classifying documents), in sentiment analysis (understanding an author's feelings regarding some entity from text), and in machine translation. While there are, some language models constructed human linguists (e.g. WordNet), these may not lead to the best performance for some applications, specifically those requiring domain specific knowledge, and manually constructing such models may not be feasible in some settings. Research in to producing semantic models computationally has thrived since the early 1950s, and as such, there are countless methods available to do so. While techniques such as term frequency-inverse document frequency (TF-IDF), and Latent semantic analysis (LSA) have proven valuable and successful across many applications over the years, they can miss important relationships between words, and become less effective under certain conditions (e.g. classification tasks with numerous classes, situations where corpus texts contain only a small number of words). This is said to be due to their high dimensionality, essentially a lack of conciseness in the representations of words and relationships with their neighbours across contexts (Le & Mikolov, 2014) (Gupta, Karnick, Bansal, & Jhala, 2016).

Other methods have been deduced that are shown to produce better results in experimental settings, which are said to better capture the similarity in meaning (or semantic similarity) between words across a

corpus. While these methods are a staple of Natural Language Processing (NLP) and the computational linguist's tool-belt, there is speculation across other fields of linguistics as to whether statistical models can truly capture/represent meaning. The first section of this paper explores the origins of distributional semantics, and research conducted across sub-disciplines of linguistics (primarily computational-linguistics, psycholinguistics, and neurolinguistics). The second section explores experimental configurations and combinations of count-based and predictive methods for generating distributed semantic models, common steps in the implementation process, strengths, and weaknesses of methods, as well as different means of evaluating them. The final section of the paper discusses the methods used to select the literature used throughout this paper.

1 Distributional Semantics (DS) across sub-disciplines of linguistics

Linguistics is an incredibly broad field, which can be broken in to several branches, such as phonology (the study of the structure of sound in language), syntax (the rules behind the structure of sentences), and semantics (the study of meaning in language) among many others (Vater, 2006). There are also several sub-disciplines of linguistics such as mathematical linguistics, cognitive linguistics and computational linguistics to name a few. Each sub-discipline has unique motivations and perspectives in studying these branches of linguistics, with their own slightly differing questions they aim to answer. (Lenci, 2008) states that because of the different aims of each field, and the lack of collaboration between fields, opportunities for better understanding the effectiveness and limitations of semantic modelling methods have been missed.

According to (Vater, 2006), psycholinguistics aims to explore how language is stored in the human memory, how one acquires language, and how we produce and comprehend language. Other areas of study might include how language is lost (aphasia), and the relationships between language and other cognitive systems such as perception and thought. In the study of semantics, a key issue is defining precise, methodological criteria for the semantic content of words, and expressing the conditions in which words share similar meaning is important to both linguistic and psychological explorations of semantics. A proposed methodology, and one of the key concepts behind DSM, is the distributional hypothesis (DH) proposed by linguist and syntactician (Harris, 1954). The DH broadly states that the similarity in meaning between two units of a language is a function of the similarity of the contexts in which each unit appears. While Harris proposes the hypothesis in the context of phonemes, it is applicable at all linguistic levels including whole words, and even sequences of words.

Harris attributed meaning to the general characteristics of human activity, rather than being a property of language, and described the relationship between language and meaning as a complicated one which does not always conform to subjective experience. For example, while a person's perceptions (and meanings) change over the years, their language tends to remain fairly consistent. It is also possible for people to experience feelings which they cannot communicate with the language available to them. We also cannot say that a word has a central meaning, as many factors throughout time, as well as cultural factors, can affect meaning. While

the DH gives some insight into the semantic similarity between words, it does not explain/reveal meaning its self, which some prominent figures in the field believe will remain beyond the scope of linguistic research (Duan, 2017). (Lewis, 1970) indirectly criticizes distributional analysis, claiming that semantic models using probabilities as metrics is a naïve approach to understanding semantics in language, and is not sufficient for conveying semantic similarity alone. He believes that the ideal methods would allow for complex structures encompassing concepts and entities in the real world and in the minds of speakers.

In a sense, while Lewis indirectly criticises distributional methods, he and Harris agree that meaning extends to something greater and more complex beyond language alone. While distributional methods may come under criticism, they are considered a staple for modelling, and furthering the study of semantic relatedness in the computational linguistics community, and have proven invaluable in a variety of applications (explored in the next section). While the storage and representation of units of a language are an important area of study in both sub-fields of linguistics, each explores storage in terms of completely different devices (a computer vs. the human brain). Given the vastly different strengths and limitations in the properties of these systems, and how they operate, it would seem that there is little material that is immediately transferable here. Although the study of the mental lexicon would still be worthwhile to a computational linguist, as insights in to its structure could very well inspire a new method, in the same way that Artificial Neural Networks were loosely inspired by the structure of their biological counter parts. The complexity of meaning and its propensity to change in different environments highlighted by the psycholinguistics community, certainly does emphasises the need for dynamic methods to generate models of meaning, in order to be able to effectively keep up with the evolution of language.

2 A review of DSM methods

DSM methods fall in to two categories, count-based methods (CBMs) which are primarily statistical, and predictive methods (PMs) which make use of machine learning methods (primarily neural networks) to predict semantic similarity between words. (Baroni, Dinu, & Kruszewski, 2014) describe PMs as "the new kids on the distributional semantics block", in contrast to count-based methods, which are comparatively more mature. This section explores properties common to the application of all DSM techniques[1], applied examples, and literature exploring and evaluating different configurations of both types of method. PMs will be the primary focus of this review, as most recent research (at the time of writing) is directed at this family of methods. It is also worth noting that while some papers such as (Boom, Canneyt, & Bohez, 2015) don't refer to TF-IDF as a DSM method, this paper will consider it as a one, as the weighting method takes advantage of context (the presence or lack of the word across the entire corpus), and thus conforms to the distributional hypothesis.

[1] Common properties of methods observed across literature referred to by this paper.

2.1 Common properties across methods

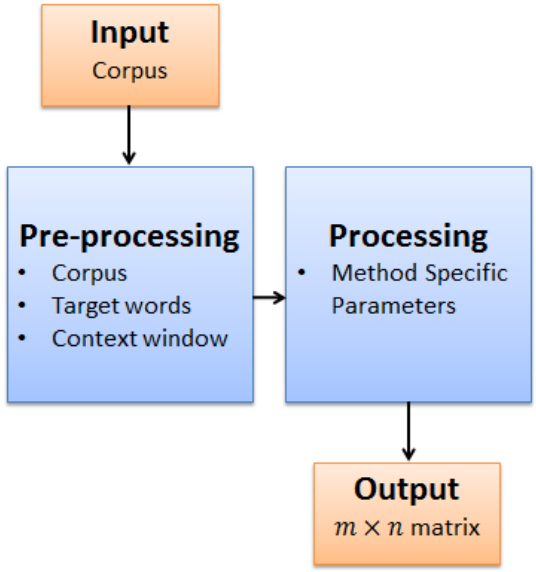

Figure 1: The general process of a DSM construction method, where input, pre-processing, and output are common across all methods, and processing tends to vary widely across methods.

Figure 1 is a broad illustration of the process involved in producing a DSM, where the input, pre-processing, and output are common across all methods, and the processing stage is very much method specific. The first pre-processing step is the selection of **target words** (the words we wish to model meaning for), and the definition of the **context space**. The selection of target words might be done based on thresholds applied to term frequency or to the values of the weights produced by a weighting scheme/method (e.g. choose words that occur more than 10 times in the corpus, but less than 10,000 times).

A context is often some number of words either side of a target word (referred to in literature as a context window), or perhaps all words in documents containing the target word. This stage may also include noise reduction, such as omitting certain word classes (e.g. conjunctions, determiners) from the corpus. The processing stage involves parsing the selected corpus with some algorithm, using the specified target and context parameters. Dimensionality, and/or noise reduction techniques are inherent in some methods, while others might require additional processing steps to achieve this effect. The selection of these parameters would depend largely on the application of the model.

The output of all modelling methods would be a $m \times n$ matrix, where each row (or word vector) m represents each target word, and each column n corresponds to the indices of all context words (see Figure 2 for an example of word vectors) (Jurafsky, 2018). To measure the similarity of (or distance between) two word vectors from a model, any distance metric (e.g. cosine similarity, Euclidean distance, L_1 etc.) can be applied, although cosine similarity is the most commonly applied metric, to measure the angle between two word vectors. Theoretically, from these outputted matrices, semantic relationships between pairs or sets of words may present themselves as numerical patterns (e.g. some common numerical offset for each type of semantic relation) (Mikolov, Yih, & Zweig, 2013). The output is then evaluated with some application specific dataset, which can either be created computationally from a corpus, or manually created by humans (e.g. a group of people assign similarity scores to pairs of words manually)

	cheese	cat	eat	purr	soft
cheese	1	0	0.6	0	0.5
cat	0	1	0.4	0.8	0.6
…	…	…	…	…	…

Figure 2: A fictitious example of two word vectors for words "cat" and "cheese", and the scores strengths of their relationships to the five words in the corpus. In this example, a score of one indicates that the words always occur together.

2.2 Tools, Techniques and Experiments

Identifying the intended meaning behind words with multiple possible meanings (word sense disambiguation), and being able to interpret textual queries computationally is an important component of many modern applications. Algorithmic methods are not the only way of obtaining models of meaning usable by such applications, WordNet is an example of a semantic model constructed by lexicographers. WordNet is a database, which groups interchangeable/synonymous and collocated words in to sets (synsets), these sets may then be connected to each other based on their semantic relationships (see Table 1 for definitions of semantic relationships) (Princeton University, 2010).

Relation	Description	Example
Synonymy	Words that share the same meaning (or nearly share the same meaning)	Sick, ill
Hypernymy	A hypernym is superordinate to its referent words	'Device' is a hypernym of 'Computer'.
Hyponymy	A hyponym is subordinate to its referent words	'Computer' is a hyponym of 'device'
Meronymy	A meronym is a part of its referent word(s)	'Page' is a meronym of 'book'
Holynymy	A holonym has each referent as a part of its self	'Book' is a holonym of 'page'
Troponymy	A way of doing the referent	'Whisper' is a troponym of 'speak'
Coordinate terms	Words that share hypernyms	'welshman' and 'scottsman' share the hypernym 'nationality'

Table 1: A tale of definitions and examples of different semantic relationships that can occur between words.

While WordNet is a powerful tool, it can require a certain level of maintenance to keep up with the evolution of language, or to adapt its vocabulary to a particular domain. There is also no "one size fits all" lexical resource that will perform well across all domains, work by (Meyer & Gurevych, 2011) and (Bentivogli, Bocco, & Pianta, 2004) highlight the need to extend the WordNet database to effectively account for domain-specific terminology. (Pekar & Staab, 2003) state that such maintenance would be incredibly time consuming to attempt manually, and in some settings may be unfeasible. They propose a more feasible, CBM for extending thesauri semi-automatically. To test the method, they construct a dataset of word vectors from nouns present in their test corpora: The British National Corpus (1807 nouns forming 233 synsets), and the Associated Press Corpus (816 nouns forming 137 synsets), where each vector links a noun (the target word) to a verb (the context) by a conditional probability value (likelihood of occurring together).

They split this data in to test and training sets, where their classification process must assign nouns in the test set to the correct synset (or class), where a single noun may belong to multiple classes. Firstly, the process uses a k-nearest neighbours classifier, which utilizes the L_1 distance metric to measure semantic similarity between word vectors, and produces a ranked list of candidate classes for a given word vector. Secondly, the process uses a measurement of the semantic similarity between candidate classes to influence the final decision of the overall method. This is done by measuring

the lowest common hypernym (the lowest level of hypernym shared by the two synsets), favouring those with higher similarity, and disfavouring those with lower similarity. While the method showed improvement over methods using only word vector similarity, learning accuracy still remained below 50%. They found across all their experiments, that greater values for k in the kNN classifier (beyond 60), degraded performance of the methods, as this would have resulted in the kNN classifier outputting a greater number of candidate classes to consider. They refer to related experiments with distributed methods, which performed well for a small number of classes (around 5), but performed poorly for larger numbers of classes.

(Agirre, Alfonseca, Hall, Kravalova, Pasca, & Soroa, 2009) take a similar, but slightly different approach in a different application area, using distributional methods to work alongside WordNet, rather than extend it. They explore and compare the results of Distributional and WordNet-based methods for estimating graded word similarity and relatedness, as well as constructing a method that combines both approaches, such that one method makes up for the weaknesses of another. They apply their methods to the Rubenstein & Goodenough and WordSim353 datasets, which are both comprised of word pairs, where each word pair has a similarity score assigned by human subjects (the average of all human assigned scores). In their WordNet methods, they represent WordNet as a graph, where each node is a synset, and each edge connecting the nodes represents a semantic relationship between the synsets. For each word in a word pair, they apply a variation of the PageRank algorithm (iterative graph based algorithm producing relevance scores based on references between nodes) to produce a word vector, and then measure the distance between the two vectors to produce their similarity score. They use three variations/configurations of distributional CBMs: a **bag-of-words** (BoW) approach, a **context-window** approach, and a **syntactic dependency** approach. In the BoW approach, the frequency of each word in the context window is a unique feature in the word vector. In the **context-window** approach, the frequency of a particular sequence of words (n-gram) within the context window forms a single feature, given the multiple uses of the term "context window", for clarity, this approach will be referred to as the n-gram approach. In the **syntactic dependency** approach, the word vector is constructed via dependency parsing, which models relationships between modifiers and nouns hierarchically (see Figure 3 for an example of a dependency grammar), the context window in this approach is the depth of the tree. The resultant vectors from each approach are measured with cosine distance to produce similarity scores for a word pair.

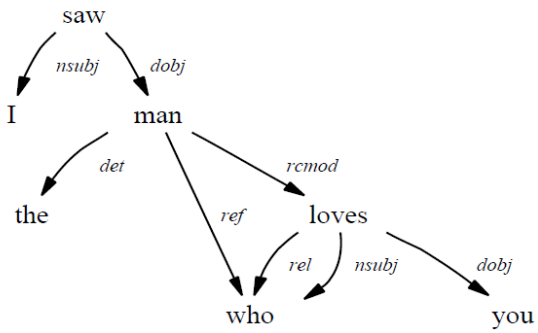

Figure 3: Illustration of a dependency grammar for the sentence "I saw the man who loves you" taken from (Marneffe, MacCartney, & Manning, 2006). The grammar models the relationships between and functional properties of the words in the sentence.

They use a corpus of 4 billion English documents scraped from the internet, for training in their DSMs. They find that their WordNet-based methods are better at capturing relatedness (e.g. meronymy) better than their distributional counterparts,

however their results were impacted by the presence of unknown words (particularly in the WordSim353 dataset). They experimented with training their distributional methods (n-gram and BoW) on varying sizes of corpora, and found that as corpus size increased, so did the quality of the resulting model. In modifying the size of their context windows, they also confirm that larger context windows may result in poorer models if a corpus is too sparse. To see if these methods could complement each other, they trained a Support Vector Machine (SVM) classifier to select the most appropriate output between two distributional, and a single WordNet method, which showed marginal improvements, although they did not optimise all parameters of the classifier, which could have led to greater improvements. This seems like a more sensible approach than merely extending WordNet, as it would seem from Agirre et al.'s experiments that distributional methods could sometimes present better results than WordNet-based ones, even when WordNet contains the word. Using a multilingual version of WordNet (Multilingual central repository), they also show that their methods are applicable to cross-lingual semantic similarity grading with only minor loss of performance, emphasising the applicability of these methods to machine translation tasks.

In an attempt to better understand the characteristics of the output of DSMs, (Mikolov, Yih, & Zweig, 2013) explore the degree to which different syntactic and semantic relationships are captured within language models generated by different techniques. They produced a language model with a Recurrent Neural Network (RNN) language-modelling toolkit, where batches of word vectors were fed to the network, which then outputs some probability distribution over words. The network's output is compared with the expected values to test the suitability of the learned distribution, and small changes are propagated backwards through the network to maximise the likelihood function (an objective function that assesses the suitability of a distribution). They show that different relationships (e.g. male/female, plural/singluar) between words (and their vectors) present themselves in the form of consistent offsets, such that "$King - Man + Woman \approx Queen$" or "$apple - apples \approx car - cars \approx family - families$", where each word represents their respective vectors produced by the network (see Figure 4 for a visual illustration of the offsets). They argue that graded smiliarity is more informative than just discretely classifying relationships, for example "*dog:bark* is more similar to *cat:meow* than *car:vroom*" is more informative than simply classifying all three word pairs as an "*ENTITY:SOUND*" relationship. They hypothesize that these offsets can be used to produce graded similarities, and answer syntactic analogy questions in the form of "*a* is to *b* as *c* is to *d*" where *d* is unknown. To test the theory, they construct their own dataset of questions to test an understanding of types of adjectives (positive, comparative, superlative), nouns (possessive, non-possessive), and verbs (base, past, and 3rd person present tense) from a corpus of 267M words from newspaper texts. They part-of-speech tag all words in the corpus, select 100 of the most common types of each type of noun, verb and adjective, and randomly match them with other words from the same category (see Table 2 for example questions). To test the ability to produce graded similarity, they use the SemEval-2012 task 2 dataset, composed of data from human subjects, which contains a test set of word pairs of the same relation, to be ordered by the degree/strength of adherance to said

relation (University of York Department of Computer Science, 2012).

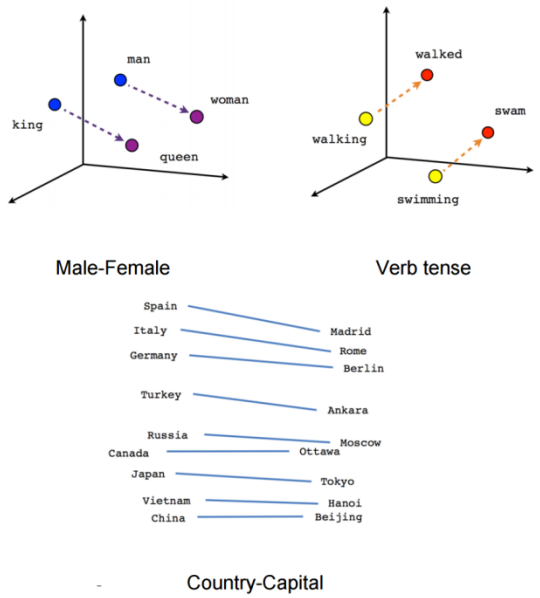

Figure 4: Illustration of word vector offsets for the male/female relationship, offsets for verb tenses, and offsets for country-capitol relationships from (TensorFlow, 2018)

Category	Relation	Example
Adjectives	Comparative/ Superlative	better:best rougher:___
Nouns	Singular/ Plural	year:years law:___
Verbs	Past/ 3rd Person Singular Present	saw:sees returned:___

Table 2: A sample of test set patterns for the graded syntactic similarity problem from (Mikolov, Yih, & Zweig, 2013)

To answer the syntactic analogy questions[2], they calculate vector $y = b - a + c$, and if no corresponding word exists for vector y, the word vector with the shortest cosine distance is retrieved as the answer. To answer semantic similarity questions, they use the cosine similarity metric on the words provided and order words accordingly. They experimented with word vectors from the RNN model of varying dimensionalities, and compared the results with word vectors from a count based method (Latent Semantic Analysis), and two other predictive methods with different neural network architectures (feed forward with a single hidden layer, and a convolutional neural network). They found that the RNN vectors vastly outperformed the LSA vectors across both tasks, and that the performance of the RNN vectors increased as their dimensionalities increased. The RNN vectors were also tested against vectors from two other PMs utilising different neural network architectures, one language model generated by a convolutional neural network (CNNLM) (Collobert & Weston, 2008), and another generated by a multilayer perceptron trained on a hierarchical, tree-based representation of its training corpus (MLPLM) (Mnih & Hinton, 2008). The RNN vectors outperformed the CNNLM across both tasks, outperformed the MLPLM vectors in the semantic similarity test, and showed near equal performance with the MLPLM in the syntactical analogy task. Given that the models were trained specifically for the syntactical analogy questions, Mikolov et al. were surprised to find that the RNN vectors were not only transferable to the semantic similarity task, but also outperformed the previously best performing system.

Focusing specifically on working with smaller samples of texts (as one might working with data from social media for example), (Boom, Canneyt, & Bohez, 2015) conduct an analysis on CBM (specifically TF-IDF) and PM models perform on shorter texts. Instead of using a human collated dataset of short texts (e.g. SemEval2015 Twitter Paraphrase dataset), they construct their own dataset of short texts from Wikipedia (10, 20, and 30 words long), where some pairs are semantically related (within close proximity in the same article), and others aren't (from completely

[2] Questions in the form of "*a* is to *b* as *x* is to *y*", where *y* is unknown.

random articles). They avoid the human collated datasets as they argue that the semantic relationships within are often too narrow. Using TF-IDF and checking cosine similarity between pairs yields low similarity values for similar word pairs, which they attribute to the short length of texts (term frequency is unlikely to be greater than one for words within a text). They also find many non-pairs with high similarity values, for which they believe frequently occurring, non-informative words (e.g. conjunctions, determiners), are responsible (Figure 5). The distribution of similarity measurements with vectors from the PM are much better (Figure 6), although a significant number of unrelated pairs still return high similarity values, again likely due to non-informative words. The authors find that using IDF alone, a measure of uniqueness of a particular word across a corpus, to consider only the top 30% of unique words, and to weight the word vectors produced by the PM, provides better performance than either method alone.

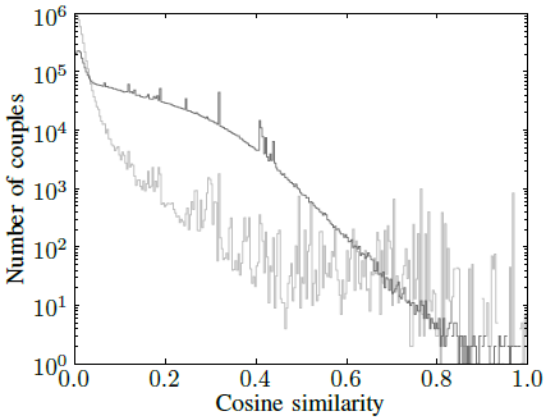

Figure 5: Histogram plot of the cosine similarity values for TF-IDF word vectors measured from couples of text, where dark grey represents related pairs, and light grey represents unrelated pairs. Figure from (Boom, Canneyt, & Bohez, 2015)

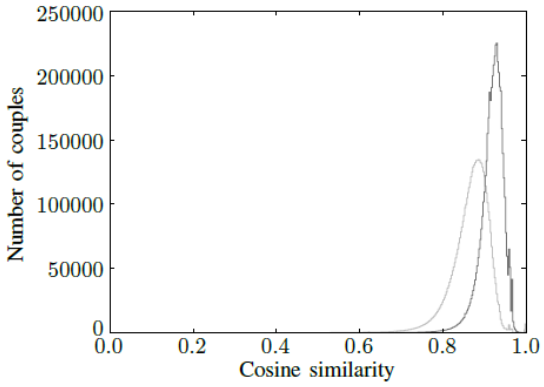

Figure 6: Histogram plot of the cosine similarity values for PM word vectors measured from couples of text, where dark grey represents related pairs, and light grey represents unrelated pairs. Figure from (Boom, Canneyt, & Bohez, 2015)

2.3 Review and Recommendations

While (Baroni, Dinu, & Kruszewski, 2014) suggest that PMs may be the future of DSM, (Boom, Canneyt, & Bohez, 2015) show that combining them with CBM can result in a stronger model, where one methods strengths can alleviate the weaknesses of another. As was seen in the work by (Agirre, Alfonseca, Hall, Kravalova, Pasca, & Soroa, 2009) and (Boom, Canneyt, & Bohez, 2015), combining completely different methods to work together can lead to improvements. Some works explored in this section evaluated their work against datasets created with human subjects; others used datasets created computationally, while others used a combination of the two. (Agirre, Alfonseca, Hall, Kravalova, Pasca, & Soroa, 2009) justify the use of human datasets with a study showing that similarity scores assigned by groups of humans for language tend to be reasonably consistent. (Boom, Canneyt, & Bohez, 2015) argue that the semantic relations within can be too narrow, and use this to justify the use of their computationally generated dataset. Although they note the disadvantage of using Wikipedia to generate data for testing/training DSMs for texts from

social media, they don't acknowledge that in automatically generating a dataset from millions of texts, there could be errors in the result which could easily be missed[3], and checking the entire dataset manually would be unfeasible. For this reason, the author concludes that evaluating models with both machine created and human created datasets would be the most robust means of testing DSMs.

3 Conclusion

To conclude, the literature explored throughout the first section showed that while the motivations of each sub-discipline of linguistics are independent, researchers could definitely benefit from evaluating literature across domains. While explorations of how language is stored and used in the human brain might not be directly applicable in computational linguistics, it could potentially inspire an Artificial Neural Network architecture specifically for DSM, in the same way that one of the most successful neural network architectures for image processing was biologically inspired (Matsugu, Mori, Mitari, & Kaneda, 2003). It also highlights the cross-domain understanding that true meaning is incredibly complex, and cannot be captured in textual data alone. Although from the second section, we can see that DSM definitely captures enough of the properties of meaning in language to be able to identify consistent patterns representing relationships. While the different methods reviewed in this literature couldn't be more different, we see that in multiple instances, combinations of them can yield improved performance over one of the methods alone. Given that these methods are only capturing essences of meaning, it's quite possible that different methods could be capturing completely different essences of meaning, which combined form a broader picture.

4 Research methods

This section contains an overview of the methods used to select papers referred to throughout this review. The first two papers were found via the article search on the University "Find It" system and Google scholar, these were (Baroni, Dinu, & Kruszewski, 2014) and (Lenci, 2008) , which both served as a basis for better understanding the material covered in each section, as well as establishing a more refined set of keywords with which to further search academic databases. I found papers surveying multiple methods to be helpful, and decided to further pursue some of their sources, whilst being wary of preventing bias in my own selection of sources. In trying to be as diverse as possible, I also utilized a service called "Iris.ai". Given a link to an academic paper, Iris.ai attempts to locate relevant papers, grouping them within a hierarchy of categories, where the top level of the hierarchy would be the most broad (which likely, and somewhat ironically, will have utilised some distributional method) (See Figures 7 & 8).

While this method provided useful, relevant papers, they did not make it in to the final selection of papers, but this was due to their lack of diversity when contrasted with the papers already chosen. The papers used for the second section were organized by type (PM or CBM), and when selecting papers, I tried to choose papers that were different from each other in terms of configurations (which hopefully resulted in a broader exploration of experiments and configurations), but also those that had

[3] For example, there is a small chance of generating a pair of related texts when non-related texts were intended and vis-versa.

conflicting/contrasting views. I personally feel that this ultimately lead to a more interesting survey.

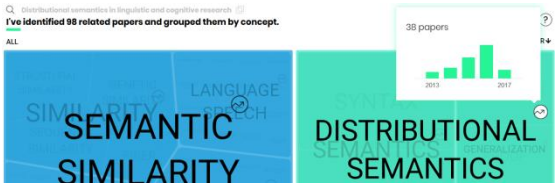

Figure 7: A screenshot of the results from Iris.ai after being presented with (Lenci, 2008). This shows the highest level in the category structure.

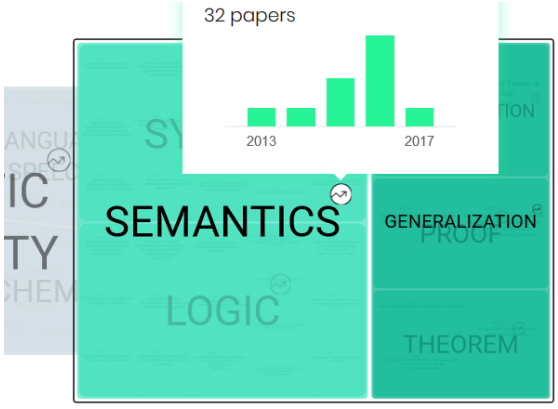

Figure 8: A screenshot of the results from Iris.ai after being presented with (Lenci, 2008). A view of categories that fell under the "Distributional Semantics" category.

Works Cited

Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Pasca, M., & Soroa, A. (2009). A study on similarity and relatedness using distributional and WordNet-based approaches. *09 Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, (pp. 19-27). Boulder.

Baroni, M., Dinu, G., & Kruszewski, G. (2014). Don't count, predict! A systematic comparison of context-counting vs. context-predicting semantic vectors. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, *1*, pp. 238-247. Baltimore.

Bentivogli, L., Bocco, A., & Pianta, E. (2004). ArchiWordNet: Integrating WordNet with Domain-Specific Knowledge. *Second International WordNet Conference, GWC*, (pp. 39-46). Brno.

Boom, C. D., Canneyt, S. V., & Bohez, S. (2015). Learning Semantic Similarity for Very Short Texts. *IEEE International Conference on Data Mining Workshop (ICDMW)*, (pp. 1229-1234).

Collobert, R., & Weston, J. (2008). A unified architecture for natural language processing: deep neural networks with multitask learning. *ICML '08 Proceedings of the 25th international conference on Machine learning* , (pp. 160-167). Helsinki.

Duan, S. (2017). Bloomfield's Concept of Meaning. *Journal of Language Teaching and Research, 8*(2), 343-348.

Gupta, V., Karnick, H., Bansal, A., & Jhala, P. (2016). Product Classification in E-commerce using Distributional Semantics. *COLING*, (pp. 536-546). Osaka.

Harris, Z. S. (1954). Distributional Structure. *WORD, 10*(2-3), 146-162.

Jurafsky, D. (2018, 01 25). *Word Similarity Distributional Similarity*. Retrieved 02 21, 2018, from Stanford University Natural Language Processing: https://www.youtube.com/watch?v=swDoFpuHpzQ

Le, Q., & Mikolov, T. (2014). *Distributed Representations of Senteces and*

Documents. Google Inc. Mountain View: arXiv.

Lenci, A. (2008). Distributional semantics in linguistic and cognitive research. *Rivista di Linguistica*, 1-31.

Lewis, D. (1970). General Semantics. *Synthese, 22*(1-2), 18-67.

Marneffe, M., MacCartney, B., & Manning, C. D. (2006). Generating Typed Dependency Parses from Phrase Structure Parses. *Proceedings of the Fifth International Conference on Language Resources and Evaluation.* Genoa.

Meyer, C. M., & Gurevych, I. (2011). What Psycholinguists Know about Chemistry: Aligning Wiktionary and WordNet for Increased Domain Coverage. *The Fifth International Joint Conference on Natural Language Processing*, (pp. 883-892). Chiang Mai.

Mikolov, T., Yih, W., & Zweig, G. (2013). Linguistic Regularities in Continuous Space Word Representations. *Proceedings of NAACL-HLT 2013* (pp. 746-751). Atlanta: Association for Computational Linguistics.

Mnih, A., & Hinton, G. (2008). A scalable hierarchical distributed language model. *NIPS'08 Proceedings of the 21st International Conference on Neural Information Processing Systems* , (pp. 1081-1088). Vancouver.

Pekar, V., & Staab, S. (2003). Word classification based on combined measures of distributional and semantic similarity. *EACL '03 Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics. 2*, pp. 147-150. Budapest: Association for Compuational Linguistics.

Princeton University. (2010). *About WordNet*. Retrieved March 22, 2018, from WordNet: https://wordnet.princeton.edu/

Pushpa, C., Deepak, G., Zakir, M., Thriveni, J., & Venugopal, K. (2016). Enhanced Neighborhood Normalized Pointwise Mutual Information Algorithm for Constraint Aware Data Clustering. *ICTACT Journal on Soft Computing, 6*(4), 1287-1292.

TensorFlow. (2018, March 29). *Vector Representations of Words*. Retrieved April 3, 2018, from TensorFlow: https://www.tensorflow.org/tutorials/word2vec

University of York Department of Computer Science. (2012, April 10). *Measuring Degrees of Relational Similarity*. Retrieved April 3, 2018, from University of York Department of Computer Science: https://www.cs.york.ac.uk/semeval-2012/task2.html

Vater, H. (2006). On the Mental Lexicon. *Studi Linguistici e Filologici Online, 4*(1), 175-205.