WordNet Embeddings

Chakaveh Saedi, António Branco, João António Rodrigues, João Ricardo Silva
University of Lisbon
NLX-Natural Language and Speech Group, Department of Informatics
Faculdade de Ciências
Campo Grande, 1749-016 Lisboa, Portugal
{chakaveh.saedi, antonio.branco, joao.rodrigues, jsilva}@di.fc.ul.pt

Abstract

Semantic networks and semantic spaces have been two prominent approaches to represent lexical semantics. While a unified account of the lexical meaning relies on one being able to convert between these representations, in both directions, the conversion direction from semantic networks into semantic spaces started to attract more attention recently. In this paper we present a methodology for this conversion and assess it with a case study. When it is applied over WordNet, the performance of the resulting embeddings in a mainstream semantic similarity task is very good, substantially superior to the performance of word embeddings based on very large collections of texts like word2vec.

1 Introduction

The study of lexical semantics has been at the core of the research on language science and technology as the meaning of linguistic forms results from the meaning of their lexical units and from the way these are combined (Pelletier, 2016). How to represent lexical semantics has thus been a central topic of inquiry. Three broad families of approaches have emerged in this respect, namely those advocating that lexical semantics is represented as a semantic network (Quillan, 1966), a feature-based model (Minsky, 1975; Bobrow and Norman, 1975), or a semantic space (Harris, 1954; Osgood et al., 1957).

In terms of data structures, under a semantic network approach, the meaning of a lexical unit is represented as a node in a graph whose edges between nodes encode different types of semantic relations holding among the units (e.g. hypernymy, meronymy, etc.). In a feature-based model, the semantics of a lexicon is represented by a hash table where a key is the lexical unit of interest and the respective value is a set of other units denoting typical characteristics of the denotation of the unit in the key (e.g. role, usage or shape, etc.). Under a semantic space perspective, in turn, the meaning of a lexical unit is represented by a vector in a high-dimensional space, where each component is based on some frequency level of co-occurrence with the other units in contexts of language usage.

The motivation for these three families of lexical representation is to be found in their different suitability and success in explaining a wide range of empirical phenomena, in terms of how these are manifest in ordinary language usage and how they are elicited in laboratory experimentation. These phenomena are related to the acquisition, storage and retrieval of lexical knowledge (e.g. the spread activation effect (Meyer and Schvaneveldt, 1971), the fan effect (Anderson, 1974), among many others) and to how this knowledge interacts with other cognitive faculties or tasks, including categorization (Estes, 1994), reasoning (Rips, 1975), problem solving (Holyoak and Koh, 1987), learning (Ross, 1984), etc.

In the scope of the formal and computational modeling of lexical semantics, these approaches have inspired a number of initiatives to build repositories of lexical knowledge. Popular examples of such repositories are, for semantic networks, WordNet (Fellbaum, 1998), for feature-based models, Small World of Words (De Deyne et al., 2013), and for the semantic space, word2vec (Mikolov et al., 2013a), among many others. Interestingly, to achieve the highest quality, repositories of different types typically resort to different empirical sources of data. For instance, WordNet is constructed on the basis of systematic lexical intuitions handled by human experts; the informa-

Proceedings of the 3rd Workshop on Representation Learning for NLP, pages 122–131
Melbourne, Australia, July 20, 2018. ©2018 Association for Computational Linguistics

tion encoded in Small World of Words is evoked from laypersons; and word2vec is built on the basis of the co-occurrence frequency of lexical units in a collection of documents.

Even when motivated in the first place by psycholinguistic research goals, these repositories of lexical knowledge have been extraordinarily important for language technology. They have been instrumental for major advances in language processing tasks and applications such as word sense disambiguation, part-of-speech tagging, named entity recognition, sentiment analysis (e.g. (Li and Jurafsky, 2015)), parsing (e.g. (Socher et al., 2013)), textual entailment (e.g. (Baroni et al., 2012)), discourse analysis (e.g. (Ji and Eisenstein, 2014)), among many others.[1]

The proliferation of different types of representation for the same object of research is common in science, and searching for a unified rendering of a given research domain has been a major goal in many disciplines. To a large extent, such search focuses on finding ways of converting from one type of representation into another. Once this is made possible, it brings not only the theoretical satisfaction of getting a better unified insight into the research object, but also important instrumental rewards of reapplying results, resources and tools that had been obtained under one representation to the other representations, thus opening the potential for further research advances.

This is the case also in what concerns the research on lexical semantics. Establishing whether and how any given lexical representation can be converted into another representation is important for a more unified account of it. On the language science side, this will likely enhance the plausibility of our empirical modeling about how the mind-brain handles lexical meaning. On the language technology side, in turn, this will permit to reuse resources and find new ways to combine different sources of lexical information for better application results.

In the present paper, we seek to contribute towards a unified account of lexical semantics. We report on the methodology we used to convert from a semantic network based representation of lexical meaning into a semantic space based one, and on the successful evaluation results obtained when applying that methodology. We resorted to Princeton WordNet version 3 as a repository of the lexical semantics of the English language, represented as a semantic graph, and converted a subgraph of it with half of its concepts into wnet2vec, a collection of vectors in a high-dimension space. These WordNet embeddings were evaluated under the same conditions that semantic space based repositories like word2vec are, namely under the processing task of determining the semantic similarity between pairs of lexical units. The evaluation results obtained for wnet2vec are around 15% superior to the results obtained for word2vec with the same mainstream evaluation data set SimLex-999 (Hill et al., 2016).

2 Distributional vectors from ontological graphs

For a given word w, its distributional representation $\vec{w}$ (aka word embedding) is a high dimension vector whose elements $\vec{w}_i$ record real valued scores expressing the strength of the semantic affinity of w with other words in the vocabulary. The usual source of these scores, and ultimately the empirical base of word embeddings, has been the frequency of co-occurrence between words taken from large collections of text.

The goal here instead is to use semantic networks as the empirical source of word embeddings. This will permit that the lexical knowledge that is encoded in a semantic graph be re-encoded as an embeddings matrix compiling the distributional vectors of the words in the vocabulary.

To determine the strength of semantic affinity of two words from their representation in a semantic graph, we follow this intuition: the larger the number of paths and the shorter the paths connecting any two nodes the stronger is their affinity.

To make this intuition operative we resort to the following procedure, to be refined later on. First, the semantic graph G is represented as an adjacency matrix M such that iff two nodes of G with words w_i and w_j are related by an edge representing a direct semantic relation between them, the element M_{ij} is set to 1 (to 0 otherwise).

Second, to enrich M with scores that represent the strength of semantic affinity of nodes not directly connected with each other by an edge, the following cumulative iteration is resorted to

$$M_G^{(n)} = I + \alpha M + \alpha^2 M^2 + \ldots + \alpha^n M^n \quad (1)$$

where I is the identity matrix; the n-th power of

[1] For the vast number of applications of WordNet, see http://lit.csci.unt.edu/∼wordnet

the transition matrix, M^n, is the matrix where each M_{ij} counts the number of paths of lenght n between nodes i and j; and $\alpha < 1$ is a decay factor determining how longer paths are dominated by shorter ones.

Third, this iterative procedure is pursued until it converges into matrix M_G, which is analytically obtained by an inverse matrix operation given by[2]

$$M_G = \sum_{e=0}^{\infty} (\alpha M)^e = (I - \alpha M)^{-1} \qquad (2)$$

3 WordNet embeddings

In order to assess this procedure, we use it to convert a mainstream ontological graph into an embeddings matrix. We use Princeton WordNet (Fellbaum, 1998) as our working semantic network. This is a lexical ontology for English with over 120k concepts that are related by over 25 types of semantic relations and comprise over 155k words (lemmas), from the categories Noun (with 117k words), Verb, Adjective and Adverb.

The quality of the resulting semantic space (based on a semantic network) is assessed by resorting to the mainstream procedure to evaluate semantic spaces: (i) it is used to solve the task of determining the semantic similarity between words in a mainstream test data set used in the literature; (ii) its performance is compared to the performance of a mainstream semantic space (based on a text collection), namely word2vec (Mikolov et al., 2013b), which serves as our baseline.

The base data set was obtained by extracting a sub-graph from WordNet that supports a 60k word distributional matrix. All parts of speech in WordNet were considered.

The nodes in WordNet are related by different types of semantic relations (e.g. hypernymy, meronymy, etc.). Relations of different types were taken into account with identical weight for the sake of the conversion of the graph into a matrix.

Upon applying the conversion procedure by resolving equation (2),[3] its outcome M_G was subject to the Positive Point-wise Mutual Information transformation (PMI+) seeking to reduce the eventual bias introduced by the conversion towards words with more senses.

Model	Similarity
wnet2vec	0.50
word2vec	0.44

Table 1: Performance in semantic similarity task over SimLex-999 given by Spearman's coefficient (higher score is better).

For the sound application of the conversion, each line in M_G was normalized, using L2-norm, so that it corresponds to a vector whose scores sum to 1, corresponding to a transition matrix.

Finally, we used Principal Component Analysis (PCA) (Wold et al., 1987) to transform the matrix, reducing the size of the vectors and setting to 850 the dimension of the encoded semantic space.

To assess the quality of the resulting semantic space, we resorted to the test data set SimLex-999 (Hill et al., 2016), containing a list of 999 pairs of words. Each pair is associated with a score, on a 0-10 scale, that indicates the strength of the semantic similarity between the words in that pair. For each pair, with the resulting embedding matrix, the cosine between the vectors of the words in that pair is calculated and mapped into the 0-10 scale. The outcome is compared to the gold standard scores in SimLex-999 resorting to Spearman's rank correlation coefficient.[4] The respective scores are displayed in Table 1.

4 Discussion

These results indicate a clear advantage of around 15% of the WordNet embeddings, scoring 0.50, over the word2vec embeddings, scoring 0.44. This indicates that the proposed conversion procedure is very effective.

WordNet embeddings is a semantic space empirically based on an internal language resource: on a systematic elicitation and recording of the semantic relations between words, thus being closely aligned with the lexical knowledge in the minds of speakers. Word2vec, in turn, is a semantic space empirically based on an external language resource: on records of contingent language usage, namely some texts that were produced by a population of language users and happened to be

[2] This is equation (7.63) in (Newman, 2010) where it is presented as a regular equivalence measure termed Katz similarity.

[3] We used `linalg.inv` from the `numpy` package for the inverse matrix calculation.

[4] We used the `evaluate_word_pairs` function from `Gensim` package (Řehůřek and Sojka, 2010) to determine the performance of both semantic spaces, the wnet2vec and the word2vec embeddings.

collected together. Hence, while words related by some semantic relation are likely to be linked in WordNet, they may happen to rarely or never occur in relevant context windows, as practical constraints on the production and usage of language may not favor that. This may help to explain the advantage of wnet2vec over word2vec.[5]

The conversion procedure is composed by a number of steps where each may receive a range of configurations. This opens a large experimental space of which the experiment in Section 3 instantiates one set of coordinates. In the remainder of the present section we justify the eventual empirical settings used and discuss the lessons learned by exploring this experimental space. The conversion procedure will be revisited in a backwards fashion, from its final to its initial steps, with the experiments being performed over the 60k subset identified in Subsection 4.3.

4.1 Matrix manipulation

Vector dimension: There have been studies indicating the positive effect of the reduction of the dimensionality of the semantic space (e.g. (Underhill et al., 2007; Grünauer and Vincze, 2015)). We experimented with a range of final vector dimensions, namely sizes 100, 300, 850, 1000 and 3000, also over evaluation data sets other than just SimLex-999.[6] Results obtained consistently indicated that size 850 leads to better performance.[7]

Dimensionality reduction: We compared two different techniques for dimensionality reduction, PCA (Wold et al., 1987) and a neural network approach. For the neural solution, the encoder-decoder architecture with a Sigmoid activation function was employed. The model was trained using a Nadam optimizer with binary cross entropy as loss metric. Experimentation consistently indicated that PCA is substantially more successful.

Normalization and bias: We contrasted the performance of the WordNet embeddings obtained with and without normalization of the distributional vectors. Results consistently indicated the advantage of doing normalization, even if for a small margin, with a delta of around 0.08.

Ablation tests were done also with respect to PMI+, which indicated a clear advantage of applying it.

4.2 Graph manipulation

Decay factor: The best results were achieved with $\alpha = 0.75$, after experimenting with values in the range 0.65 to 0.85.

Picking semantic relations: Concepts in WordNet are connected via semantic relations of different types. The relations of Hypernymy/Hyponymy, Synonymy and Antonymy play an essential role in structuring a semantic network, as without them the network could not exist. We undertook experiments where all semantic relations or only these kernel relations were taken into account for the conversion procedure, with results indicating a clear advantage for using all relations.

Weighting semantic relations: In the definition of a semantic network, some types of relations appear as necessary (e.g. Hypernymy), while other appears as more secondary (e.g. Meronymy). It might thus happen that the conversion of a semantic network into a semantic space might be optimized if different weights were assigned to different relations accordingly. We ran an experiment where different weights were assigned to different relations, namely hypernymy, hyponymy, antonymy and synonymy got 1, meronymy and holonymy 0.8 and other relations 0.5; and another experiment where all types of semantic relation were assigned the same weight. Better results were obtained with the latter.

4.3 Base data sets

Subgraphs: The conversion procedure relies on equation (2), whose complexity is dominated by the calculation of the inverse matrix, which is of exponential order. For the Princeton WordNet graph, with over 120k concepts, given the size of the adjacency matrix M^1 is over 120k × 120k, its calculation and the overall conversion of the ontological graph into the final embeddings matrix faces substantial challenges in terms of the memory footprint. To cope with this issue, we resorted to initial subgraphs of manageable size.[8]

[5] Naturally, the comparative advantage between a semantic space based on a semantic network and another based on a collection of texts depends also on the sizes of the network and of the collection. The training corpus of word2vec-GoogleNews-vectors we used is one of the largest, with an impressive amount of 100 billion tokens, and a vocabulary of 3 million types, which differently from the vocabulary units in WordNet, are wordforms, not lemmas (Mikolov et al., 2013a).

[6] More on evaluation data sets in Section 4.4

[7] The vector size in word2vec embeddings is 300.

[8] To invert a 60k matrix, `numpy` used all memory available in a machine with 32 CPUs/2.50GHz and 430Gb RAM.

We reduced the size of M^1 by eliminating more sparse rows (rows with more zero elements), corresponding to eliminating words in concepts with lower number of outgoing edges in WordNet. Rows were ordered by decreasing sparsity, with rows with identical level of sparsity (identical number of zero elements) randomly ordered among themselves. The first 25k, 30k, 45k and 60k rows were extracted and used in the conversion process. To maximize overlap wth test set SimLex-999, its words in WordNet were retained. The performance scores of the resulting models are displayed in Table 2.

Random subgraphs	**25k**	**30k**	**45k**	**60k**
Semantic similarity	0.45	0.47	0.49	0.50

Table 2: Performance of wnet2vec in similarity task over SimLex-999 (Spearman's coefficient).

The larger the size of the WordNet subgraph the better is the performance of the resulting embeddings. As they contain more concepts, which on average are closer to each other, larger subgraphs tend to be denser and generate less sparse adjacency matrices. This supports semantic spaces with distributional vectors with more discriminative information on the semantic affinity of a word with respect to others.

The progression of scores in Table 2, for subgraphs with matrices in the range 25k-60k, supports the conjecture that when enough computational means are available and the full 155k word WordNet be used, the performance of the resulting embeddings may still improve by a substantial margin over the result now observed for the 60k matrix, with less than half of the words.

Additionally, we experimented with two specific subgraphs that were **not** randomly extracted from WordNet, namely: the subgraph supporting the matrix with the 13k most frequent words of English;[9] and the subgraph supporting the matrix with the 13k words used in (De Deyne et al., 2016),[10] which have been selected to act as cue words in psycholinguistic experiments for eliciting associated words from subjects. The performance results of the resulting models are displayed in Table 3.

Specific subgraphs	**13k most frequent**	**13k cue words**
Similarity	0.47	0.50

Table 3: Performance of wnet2vec in similarity task over SimLex-999 given by Spearman's coefficient. First row indicates the sizes of the matrices supported by specific subgraphs.

These matrices have less than $1/4$ of the size of the 60k matrix, and yet they show a better than expected approximation to its performance, taking into account the progression registered in Table 2. These results indicate that larger size is not the only factor improving the performance of WordNet embeddings. Very interestingly, they seem to indicate that words more commonly used may support semantic spaces that are more accurate to discriminate semantic similarity.

Frequency of occurence in texts plays no direct role in the conversion of semantic networks into semantic spaces by equation (2). Hence this effect likely results from the fact captured by one of the Zipf word distributions, that on average more frequent words are more ambiguous than less frequent ones: On average more frequent words express more concepts — that is, they occur in more WordNet synsets — and thus enter in more outgoing edges in the semantic network, and this should support less sparse vectors in the semantic space.

This explanation is empirically supported by the fact that the word ambiguity rates are 2.7 and 2.8, in the sugraphs with 13k cue words and with 13k most frequent words, respectively, while there is a lower word ambiguity rate of around 1.5 for the random graph with 60k words.[11]

Parts of Speech: Princeton WordNet covers nouns, adjectives, verbs and adverbs. Nouns (117k) are the largest portion of all words (155k) in the graph and, among the different POS, they support the most dense subgraph of semantic relations. We run experiments with words from all POS categories, and where only Nouns where considered. While results obtained with Nouns only (0.44) are not that distant from the results obtained with all POS (0.50), the latter setting consistently showed better performance.

[9] To reach 13k, we used the 10k most common English words, as determined by n-gram frequency analysis of the Google's Trillion Word Corpus, from (Kaufman, 2017), supplemented with non repeating words from Wiktionary frequency lists (Wiktionary, 2017).

[10] Available from https://smallworldofwords.org/en

[11] This is obtained by counting n lemmas for a word that enters WordNet under n POS categories. Word ambiguity rate of the whole WordNet is 1.3.

4.4 Testing data and metrics

To assess the robustness of the results obtained, experiments were undertaken with: (i) yet another evaluation metric, namely Pearson's correlation coefficient; (ii) further evaluation data sets for semantic similarity, namely RG1965 (Rubenstein and Goodenough, 1965) and Wordsim-353-Similarity (Agirre et al., 2009); (iii) and testing over another task, namely semantic relatedness, with the evaluation data sets Wordsim-353-Relatedness (Agirre et al., 2009), MEN (Bruni et al., 2012) and MTurk-771 (Halawi et al., 2012). In these experiments we used our best settings, with a random 60k subgraph, and our second best settings, with the best model with a specific 13k subgraph, cf. Subsection 4.3.

Additional metric: The evaluation scores obtained over SimLex-999 with the Pearson's coefficient are basically aligned with the scores already obtained with Spearman's coefficient, confirming the superiority of the WordNet embeddings.

Additional data sets: Even with a number of test pairs much lower than the pairs in SimLex-999 and built under less standard procedure, and thus supporting less reliable results, we evaluated our models over the Wordsmith353-S and RG1965 data sets. Wnet2vec showed competitive performance when put side by side with word2vec even though their scores were not superior. With these smaller alternative data sets, the results for the specific 13k model were slightly superior to the results for the random 60k model.

Additional task: The relation "semantic relatedness" is broader and less well defined than the relation "semantic similarity". Experiments with a second task of determining semantic relatedness showed that word2vec performs clearly better on this task than on the task of semantic similarity, while wnet2vec in general performs worst on it. Wnet2vec is thus less prone than word2vec to get fooled by words that are just semantically related by not necessarily similar. This indicates that the superiority of wnet2vec in the similarity task results from an enhanced discriminative capacity, with it being better both at judging as similar, words that are actually similar, and at judging as non similar, not only words that may be clearly non similar but also words that are semantically related, and thus may be close to be similar.

The results obtained with these experiments are displayed in Table 4.[12]

5 Related work

From semantic spaces to semantic networks: There has been a long research tradition on semantic networks enhanced with information extracted from text, including distributional vectors, which in the limit may encompass semantic networks obtained from semantic spaces. As a way of illustration, among many others, this includes the work on semantic relations determined from patterns based on regular expressions, either hand crafted (Hearst, 1992), or learned from corpora (Snow et al., 2005); work on semantic relations predicted by classifiers running over distributional vectors (Baroni et al., 2012; Roller et al., 2014; Weeds et al., 2014); work on semantic relations obtained with deep learning that integrates distributional information and patterns of grammatical dependency relations (Shwartz et al., 2016), including the hard task of distinguishing synonymy from antonymy (Nguyen et al., 2017); etc. While being highly relevant for a unified account of lexical semantics, this line of research addresses the conversion direction, from semantic spaces to semantic networks, that is not the major focus of this paper.

From semantic networks to semantic spaces: Work towards the conversion direction that is of interest here is more recent. As a way of illustration, among others, one can mention (Faruqui et al., 2015), which explored retrofitting to refine distributional representations using relational information, and (Yu and Dredze, 2014), which focused also on refining word embeddings with lexical knowledge, but which are not addressing the goal of obtaining semantic spaces solely on the basis of semantic networks as we do here.

That is the aim also of recent work like (Camacho-Collados et al., 2015) who improve the embeddings built from data sets made of selected Wikipedia pages by resorting to the local, one-edge relations of each relevant word in the WordNet graph.

Further recent works worth mentioning include (Vendrov et al., 2015) that resorted to order embeddings, which however do not preserve distance and/or do not preserve directionality under

[12] Pairs in the evaluation data set but not in the semantic space do not count to compute the evaluation score: proportion of vocabulary overlap does not affect the scoring.

data set	task	size	over- lap %	w2vec	n2vec 13k s	n2vec 60k r	w2vec	n2vec 13k s	n2vec 60k r
				Spearman coef			Pearson coef		
SimLex-999	simil	999	99.8	0.44	0.50	0.50	0.45	0.52	0.51
RG1965	simil	65	100.0	0.75	0.65	0.56	0.75	0.75	0.72
Wordsim353-S	simil	203	98.0	0.74	0.65	0.51	0.73	0.67	0.58
Wordsim353-R	relat	252	97.6	0.61	0.32	0.31	0.58	0.33	0.30
MEN	relat	3000	44.9	0.70	0.46	0.45	0.68	0.48	0.45
MTURK-771	relat	771	99.7	0.66	0.54	0.53	0.63	0.54	0.52

Table 4: Performance of different models in the semantic similarity (simil) and relatedness (relat) tasks over different data sets measured by Spearman's and Pearson's coefficients. Models used: word2vec (w2vec); wnet2vec with the random 60k subgraph (n2vec 60k r); and wnet2vec with the best specific 13k subgraph (n2vec 13k s), cf. Subsection 4.3. Overlap with the vocabulary of wnet2vec 60k random appears in the fourth column.

the relevant semantic relations; (Nickel and Kiela, 2017) that experimented with computing embeddings not in Euclidean but in hyperbolic space, namely the Poincaré ball model. A shortcoming with these proposals is that their outcome is not easily plugged into neural models. Also they are not fit to evaluation on external tasks, like the semantic similarity task, with their evaluation being rather based on their ability to complete missing edges from ontological graphs. In contrats, an example of the sutability of wnet2vec to be plugged into neural models and of its application in a downstream task is reported in (Rodrigues et al., 2018), where these embeddings support the predicition of brain activation based on neural networks.

There has been also a long tradition of research on learning vector embeddings from multi-relational data of which, among many others, one can refer (Bordes et al., 2013), (Lin et al., 2015), and (Nickel et al., 2016). Though to a large extent these are generic approaches for graph to vectors conversion, also here the major focus has been on exploring these models on their ability to complete missing relations in knowledge bases rather than to experiment them on natural language processing and lexical semantics.

Other related approaches worth of note are (De Deyne et al., 2016) and (Goikoetxea et al., 2015). While being based also on the iterative conversion procedure used here, the first concentrates however on converting, not a semantic network, but a fragment of the lexicon represented under a feature-based approach into a semantic space.

While seeking to obtain WordNet embeddings, the second resorts, however, not to a genuine conversion procedure, but to a lossy intermediate "textual" representation: it generates sequences of words by concatenating words visited by random walks over the WordNet; this "artificial text" is a partial and contingent reflection of the semantic network and is used to obtain distributional vectors by resorting to typical word embeddings techniques based on text.

Distances in a semantic graph:

The task of determining the semantic similarity between two words can be performed not only on the basis of the distance of their respective vectors in a semantic space, but also on the basis of the distance of the respective concepts in a lexical semantic network, like WordNet. There has been a long research tradition on this issue whose major proposals include (Jiang and Conrath, 1997), (Lin, 1998), (Leacock and Chodorow, 1998), (Hirst and St-Onge, 1998),(Resnik, 1999), among others, which received nice comparative assessments in (Ferlez and Gams, 2004) and (Budanitsky and Hirst, 2006), including their correlation with human judgments.

In this context, it is worth of note the work by (Hughes and Ramage, 2007), which resorts to random graph walks over WordNet edges. Differently from our approach, its goal is to obtain word-specific stationary probability distributions — such that the semantic affinity of two words is based on the similarity of their probability distributions —, rather than to obtain vectorial representations for words in a shared distributional semantic space.

The focus of the present paper is on an effective method to convert a semantic network into a semantic space, with the graph-based affinity obtained by the chaining of "local" one-edge distances ensured by the iteration in (1)-(2) being central for that goal.

It will be interesting to understand whether it will be possible to consider, as an alternative, those graph-based metrics of semantic similarity for any two nodes anywhere in the graph — resorting to the "non-local" multi-edge distance between the two input words. It remains to be understood whether they can be resorted to as the basis of an "all vs. all" type of procedures for an exhaustive screening of the graph that are computationally tractable — thus aiming at keeping up with an effective method for graph to matrix conversion of an entire lexical semantic network that resists the eventual exponential explosion.

6 Conclusions

In this paper, we offer a contribution towards a unified account of lexical semantics. We propose a methodology to convert from semantic networks, that are encoded in ontological graphs and empirically based on systematic linguistic intuitions (in their higher quality incarnations), to semantic spaces, that are encoded in distributional vectors and empirically based on very large collections of texts (in their higher quality implementations). This conversion methodology relies on a straightforward yet powerful intuition — the larger the number of paths and the shorter the paths connecting two nodes in an ontological graph the stronger is their semantic affinity —, with iteration (1) making it operative in order to generate a distributional matrix from an ontological graph.

We report also on the results of assessing this conversion methodology with a case study, namely by applying it to a subgraph of WordNet with less than half of its words (60k), randomly selected from the ones whose senses have a larger number of outgoing edges. The resulting distributional vectors wnet2vec were evaluated under the mainstream task of determining the semantic similarity of words arranged in pairs, against the mainstream gold standard SimLex-999, with very good results. The performance of wnet2vec was around 15% superior to the performance of word2vec, trained on a 100 billion token collection of texts. This indicates that the proposed conversion procedure is very effective and that the WordNet embeddings are competitive when compared to text based embeddings.

It is nevertheless worth underlying that the research goal of this paper was not to search for word embeddings that outperform all previous proposals known in the literature in terms of intrinsic evaluation tasks, like semantic similarity, etc., or when they are embedded in larger systems. Its research goal was rather to demonstrate that it is feasible to create very effective word embeddings from semantic networks with a straightforward and yet powerful method of conversion from semantic networks to semantic spaces that, given its simplicity, offer the promise to generalize very well for more types of lexical networks and ontologies other than just WordNet, which was the case study used here.

The fact that less than half of the words in WordNet were used in the reported experiment reinforces this positive expectation with respect to the strength of the proposed approach, and point towards future work that will seek to use larger portions of WordNet, as computational limitation can be overcome.

The results reported in this paper thus hint at very promising research avenues, including, among others, experiments with further ontologies of different domains, empirical origins, etc.; with cross-lingual triangulation with aligned WordNets and aligned embeddings; with reciprocal reinforcement of ontological graphs and distributional vectors; with other metrics of semantic affinity in a graph, etc.

The wnet2vec data and software and their future updates are distributed at https://github.com/nlx-group/WordNetEmbeddings

Acknowledgments

The research results presented here were partly supported by PORTULAN/CLARIN Infrastructure for the Science and Technology of Language, by the National Infrastructure for Distributed Computing (INCD) of Portugal, and by the ANI/3279/2016 grant.

References

Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Paşca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches. In *NAACL-HLT2009*, pages 19–27.

John Robert Anderson. 1974. Retrieval of propositional information from long-term memory. *Cognitive Psychology*, 6(4):451–474.

Marco Baroni, Raffaella Bernardi, Ngoc-Quynh Do, and Chung-chieh Shan. 2012. Entailment above the word level in distributional semantics. In *EACL2012*, pages 23–32.

Daniel G. Bobrow and Donald Arthur Norman. 1975. Some principles of memory schemata. In *Representation and Understanding: Studies in Cognitive Science*, page 131–149. Elsevier.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26*, pages 2787–2795.

Elia Bruni, Gemma Boleda, Marco Baroni, and Nam-Khanh Tran. 2012. Distributional semantics in technicolor. In *ACL2012*, pages 136–145.

Alexander Budanitsky and Graeme Hirst. 2006. Evaluating wordnet-based measures of lexical semantic relatedness. *Computational Linguistics*, 32(1):13–47.

José Camacho-Collados, Mohammad Taher Pilehvar, and Roberto Navigli. 2015. Nasari: a novel approach to a semantically-aware representation of items. In *NAACL-HLT2015*, pages 567–577.

Simon De Deyne, Daniel J Navarro, and Gert Storms. 2013. Better explanations of lexical and semantic cognition using networks derived from continued rather than single-word associations. *Behavior Research Methods*, 45(2):480–498.

Simon De Deyne, Amy Perfors, and Daniel J Navarro. 2016. Predicting human similarity judgments with distributional models: The value of word associations. In *COLING2016*, pages 1861–1870.

William K Estes. 1994. *Classification and Cognition*. Oxford University Press.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. 2015. Retrofitting word vectors to semantic lexicons. In *AACL-HLT 2015*, pages 1606–1615.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Jure Ferlez and Matjaz Gams. 2004. Shortest-path semantic distance measure in wordnet v2.0. *Informatica*, 28:381–386.

Josu Goikoetxea, Aitor Soroa, and Eneko Agirre. 2015. Random walks and neural network language models on knowledge bases. In *NAACL-HLT25*, pages 1434–1439.

Andreas Grünauer and Markus Vincze. 2015. Using dimension reduction to improve the classification of high-dimensional data. *arXiv preprint arXiv:1505.06907*.

Guy Halawi, Gideon Dror, Evgeniy Gabrilovich, and Yehuda Koren. 2012. Large-scale learning of word relatedness with constraints. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1406–1414. ACM.

Zellig S Harris. 1954. Distributional structure. *Word*, 10(2-3):146–162.

Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *COLING1992*, pages 539–545.

Felix Hill, Roi Reichart, and Anna Korhonen. 2016. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*, 41:665–695.

G. Hirst and D. St-Onge. 1998. Lexical chains as representations of context for the detection and correction of malapropisms. In C. Fellbaum, editor, *WordNet: An Electronic Lexical Database*, pages 305–332. MIT Press.

Keith J Holyoak and Kyunghee Koh. 1987. Surface and structural similarity in analogical transfer. *Memory & Cognition*, 15(4):332–340.

Thad Hughes and Daniel Ramage. 2007. Lexical semantic relatedness with random graph walks. In *EMNLP-CONLL2007*, Prague, Czech Republic.

Yangfeng Ji and Jacob Eisenstein. 2014. Representation learning for text-level discourse parsing. In *ACL2014*, pages 13–24.

J. Jiang and D. Conrath. 1997. Semantic similarity based on corpus statistics and lexical taxonomy. In *Proceedings on International Conference on Research in Computational Linguistics*.

Josh Kaufman. 2017. 10,000 most common english words in google's trillion word corpus. `https://github.com/first20hours/google-10000-english`.

C. Leacock and M. Chodorow. 1998. Combining local context and wordnet similarity for word sense identification. In C. Fellbaum, editor, *WordNet: An Electronic Lexical Database*, pages 265–285. MIT Press.

Jiwei Li and Dan Jurafsky. 2015. Do multi-sense embeddings improve natural language understanding? *arXiv preprint arXiv:1506.01070*.

D. Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of 15th International Conference on Machine Learning*.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *AAAI'15*, pages 2181–2187.

David E Meyer and Roger W Schvaneveldt. 1971. Facilitation in recognizing pairs of words: Evidence of a dependence between retrieval operations. *Journal of Experimental Psychology*, 90(2):227.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Googlenews-vectors-negative300.bin.gz - efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*. `https://code.google.com/archive/p/word2vec/`.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26*, pages 3111–3119.

Marvin Minsky. 1975. A framework for representing knowledge. In *Psychology of Computer Vision*. McGraw-Hill.

Mark Newman. 2010. *Networks: An Introduction*. Oxford University Press.

Kim Anh Nguyen, Sabine Schulte im Walde, and Ngoc Thang Vu. 2017. Distinguishing antonyms and synonyms in a pattern-based neural network. *arXiv preprint arXiv:1701.02962*.

Maximilian Nickel, Lorenzo Rosasco, and Tomaso Poggio. 2016. Holographic embeddings of knowledge graphs. In *AAAI'16*, pages 1955–1961.

Maximillian Nickel and Douwe Kiela. 2017. Poincaré embeddings for learning hierarchical representations. In *Advances in Neural Information Processing Systems 30*, pages 6341–6350.

Charles E Osgood, George J Suci, and Percy H Tannenbaum. 1957. The measurement of meaning. *Urbana: University of Illinois Press*.

Francis Jeffrey Pelletier. 2016. Semantic compositionality. In *The Oxford Research Encyclopedia of Linguistics*. Oxford University Press.

M Ross Quillan. 1966. Semantic memory. Technical report, Bolt Beranek and Newman Inc., Cambridge MA.

Radim Řehůřek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pages 45–50. European Language Resources Association.

P. Resnik. 1999. Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11.

Lance J Rips. 1975. Inductive judgments about natural categories. *Journal of Verbal Learning and Verbal Behavior*, 14(6):665–681.

João António Rodrigues, Ruben Branco, João Ricardo Silva, Chakaveh Saedi, and António Branco. 2018. Predicting brain activation with wordnet embeddings. In *Proceedings of the 8th Workshop on Cognitive Aspects of Computational Language Learning and Processing (CogACLL2018), the 56th Annual Meeting of the Association for Computational Linguistics (ACL2018)*, Melbourne, Australia. Association for Computational Linguistics.

Stephen Roller, Katrin Erk, and Gemma Boleda. 2014. Inclusive yet selective: Supervised distributional hypernymy detection. In *COLING 2014*, pages 1025–1036.

Brian H Ross. 1984. Remindings and their effects in learning a cognitive skill. *Cognitive Psychology*, 16(3):371–416.

Herbert Rubenstein and John B Goodenough. 1965. Contextual correlates of synonymy. *Communications of the ACM*, 8(10):627–633.

Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016. Improving hypernymy detection with an integrated path-based and distributional method. In *ACL2016*, pages 2389–2398.

Rion Snow, Daniel Jurafsky, and Andrew Y Ng. 2005. Learning syntactic patterns for automatic hypernym discovery. In *Advances in Neural Information Processing systems 17*, pages 1297–1304.

Richard Socher, John Bauer, Christopher D Manning, and Andrew Y. Ng. 2013. Parsing with compositional vector grammars. In *ACL2013*, pages 455–465.

David G Underhill, Luke K McDowell, David J Marchette, and Jeffrey L Solka. 2007. Enhancing text analysis via dimensionality reduction. In *IEEE-IRI2007*, pages 348–353.

Ivan Vendrov, Ryan Kiros, Sanja Fidler, and Raquel Urtasun. 2015. Order-embeddings of images and language. *arXiv preprint arXiv:1511.06361*.

Julie Weeds, Daoud Clarke, Jeremy Reffin, David Weir, and Bill Keller. 2014. Learning to distinguish hypernyms and co-hyponyms. In *COLING 2014*, pages 2249–2259.

Wiktionary. 2017. Wiktionary: Frequency lists. `https://en.wiktionary.org/wiki/Wiktionary:Frequency_lists`.

Svante Wold, Kim Esbensen, and Paul Geladi. 1987. Principal component analysis. *Chemometrics and Intelligent Laboratory Systems*, 2(1-3):37–52.

Mo Yu and Mark Dredze. 2014. Improving lexical embeddings with semantic knowledge. In *ACL 2014*, pages 545–550.