Contextual Analysis for the Representation of Words

Dat Huynh, Dat Tran, Wanli Ma

***Abstract*—Building the representation of word meanings is one of the key challenges in many language-based applications from document understanding, text summarization to sentiment analysis. One of the reasons to make this task harder is that word meanings involve not only the word surfaces in contexts, but the human experiences in specific domains. Previous work in the field considers these issues separately by analysing text contents in one hand and dealing with knowledge-based information on the other hand. In this work, we address this issue by accumulating contextual information of words and knowledge-based contents to construct the representation of words. We evaluate the effectiveness of the representation via the task of semantic similarity on standard benchmarks. The experimental results show the strong correlation between the proposed word representation to the perception of human in the task of semantic similarity measure.**

I. Introduction

In many language-based applications, it is crucial to be able to measure precisely the semantic similarity between words. While many work previously has been attempted to address the task, distributional representation models has recently drawn much attention. It is based on an straightforward intuition that meanings of a word are disclosed by its surrounding contexts [1]. The model represents word meanings in a vector space that captures contextual information of the word. And therefore, the problem of measuring the semantic similarity between two words becomes the problem of measuring the distance between two vector representations.

Vector space models (VSMs) have been used to capture the contextual information of a word in different ways. Using universal knowledge-based repositories such as Wikipedia and WordNet, the representation of a word is modelled as a high dimensional vector of Wikipedia concepts [2], [3], silent concepts [4], and WordNet synsets [5]. On the other hand, some other work has utilised a large amount of plain text data available to construct the representation of the word. Lexical-syntactic dependency patterns of a word appeared in contexts are captured as features for word representation [6]. Window-based lexical patterns of a word are also used to describe the representation of the word in VSMs [7]. Latent topics that a word is likely belonged have also been used are vector features for the representation [8], [9]. The representation of a work is also learned automatically using the nature distribution of the word over a large amount of text data [10]. Recently, the preliminary work from [11] combines contextual information and latent topic information for word representation.

Dat Huynh is a PhD candidate at the University of Canberra, Australia. Email: Dat.Huynh@canberra.edu.au. Dat Tran is an Associate Professor at the University of Canberra, Australia. Email: Dat.Tran@canberra.edu.au. Wanli Ma is an Assistant Professor and an Academic Program Leader at the University of Canberra, Australia. Email: Wanli.Ma@canberra.edu.au

In this work, we utilize a large plain text corpus to present a new set of features for word semantic representation. The feature set is extracted via a task of analysing contexts to withdraw features, which are then combined with global topic of the word to form a complete the representation using VSMs. To evaluate the effectiveness of the proposed word representation, we undertake the task of semantic similarity on standard testing benchmarks. We have developed parameters on the MTurk dataset [3] and tested on the popular semantic similarity datasets such as WS-353 [9] and RG-65 [12]. The experimental results confirm the strong correlation between the generated semantic similarity scores and the human judged results on standard testing benchmarks.

As the followings, we first present the semantic analysis of word meanings in local contexts in Section II and III. Section IV discusses about word representation using global topics information. In section VI, the task of word similarity measure is described. Section VII, our experimental setups and results are discussed. Finally, the related work on semantic similarity measure is presented in Section VIII.

II. Semantic Analysis of Word Meanings in Local Contexts

Meanings of a word can be inferred from surround contexts that the word appears. Consider the following example describing the contexts of an unknown word "*tezgüino*" (the modified example from [13], [6]).

> A bottle of *tezgüino* is on the table.
> Mexican likes *tezgüino*.
> Strong *tezgüino* makes you drunk.
> We make *tezgüino* out of corn.

The contexts in which the word "*tezgüino*" is appeared suggest that the meanings of "*tezgüino*" may be a kind of alcoholic beverage that makes from "*corn*", get people "*drunk*" and normally contains in "*bottle*". In other words, the meanings of a given word could be disclosed by considering the relationship with other surrounding words in local contexts. Moreover, from the linguistics perspective, meanings of a word could be found in its local contexts where the syntagmatic relations and paradigmatic relations play an important role. They jointly describe the word meanings in different aspects [14]. While paradigmatic relations hold the meanings over long distant relations, the syntagmatic relations contain the meanings when the word interacts with its adjacent neighbours. Words are shared in a paradigmatic relation as long as they are exchangeable in their contexts but still maintain the similar meanings in the contexts. For instance, the word *tezgüino* in the contexts above could be exchanged by any words holding the meaning of "alcoholic drink" as they are sharing the same lexical patterns such as "*strong * makes you drunk*" appeared the local contexts.

Syntagmatic relation is in different way as they are described the properties/attributes features. The words such as "*bottle*" and "*corn*" are considered as attributes of "*tezgüino*".

III. Word Meaning Representation Using Local Contexts

Aforesaid, paradigmatic and syntagmatic relations play an important role in inferring word meanings in contexts. To use VSMs for word representation, lexical patterns of these relations needs to be converted into word representation features. Formally, given w_i as a focus word, the contextual vector representation $v(w_i)$ of the word w_i is considered as follows:

$$v(w_i) = \langle w_i^1, w_i^2, \ldots, w_i^n \rangle \quad (1)$$

where w_i^1 is an association degree between the focus word w_i and its word feature w_j in the condition that w_i and w_j co-occurence in a lexical pattern described either paradigmatic or syntagmatic relations of w_i. The parameter n is the size of word dictionary in the given text corpus. To extract the pair of (w_i,w_j) from the paradigmatic/syntagmatic lexical patterns, we design two different rule-based approaches.

The first extraction approach aims to retain word features that are highly associated with the focus word under particular syntactical relations. The designed pattern single-passes through the plain text and returns pairs of the focus word and its associated word features. Each pair has to match the following conditions:

1) The word feature has to be a single noun, compound noun, or a name entity
2) If existed, the sequence in between the pair from the text has to match the following patterns:

   V+ | V+W*P | P

   V= (relative word | verb | particle | adverb)
   W = (noun | adjactive | adverb | pronoun | determiner)
   P = (preposition | particle | appositional modifier)

The second extraction approach applies a window size (WS) pattern on the local contexts of the focus word w_i to extract its word feature w_j. Any word co-occurred with the focus word w_i in a window size WS will be extracted as a pair (w_i,w_j). The extracted pairs then are filtered on its frequency and the degree of association to retain those with high information values.

Different approaches come up with different information value measures. In this work, the point-wise mutual information (PMI) [15] has been used to measure the degree of information value (association) between two different words. The information value w_i^k of a pair of words (w_i, w_k) is measured as follows:

$$w_i^k = \log \frac{p(w_i, w_k)}{p(w_i)p(w_k)} \quad (2)$$

$$p(w_i, w_k) = \frac{d(w_i, w_k)}{\sum_{i,k=1\ldots n} d(w_i, w_k)} \quad (3)$$

$$p(w_i) = \frac{\sum_{k=1\ldots n} d(w_i, w_k)}{\sum_{i,k=1\ldots n} d(w_i, w_k)} \quad (4)$$

where $d(w_i, w_k)$ is the number of times that w_i and w_k co-occur.

IV. Word Meaning Representation Using Global Topic Features

In the previous section, meanings of a word are constructed using VSMs on its local context. In this part, we utilise large amount of plain text to infer topics that the word likely belongs to. The topics of a word are disclosed using entire distribution of the word in the given text corpus. We consider these features as global topic features.

Word meanings have been successfully described using explicit topics such as Wikipedia concepts [2]. However, the method relies on the network structure of Wikipedia links, which hardily adapts to different domains as well as languages. In this work, we used the latent topics instead, which could be inferred using typical a generative topic model operated on a large plain text corpus. Several variants of topic model have been proposed such as Latent Semantic Analysis (LSA) [16], Latent Dirichlet Allocation (LDA) [17]. They are all based on the same fundamental idea that documents are mixtures of topics where a topic is a probability distribution over words, and the content of a topic is expressed by the probabilities of the words within that topic. In this work, we used LDA as the background topic model in building features for word representation. LDA performs the latent semantic analysis to find the latent structure of "topics" or "concepts" in a plain text corpus.

Given a focus word w_i and a latent topic t_j, the topic model produces the probability m_i^j that w_i belongs to the particular topic t_j. As the result, the topic representation of the word w_i is considered as a vector of latent topics, where each value of the vector is represented for the probability that w_i belongs to a particular topic t_j ($j = 1 \ldots k$).

The topic representation of the word w_i is described as follows:

$$u(w_i) = \langle m_i^1, m_i^2, \ldots, m_i^k \rangle \quad (5)$$

where k is the number of latent topics. The vector $u(w_i)$ is used to describe the meanings of the word w_i using latent topic information.

V. Word Representation Using Combination of Word Features and Topic Features

Given w_i as a focus word, meanings of the word w_i is represented as a n-dimensional vector $v(w_i)$ of relational words denoted $w_1 \ldots w_n$ (see Formula 1). Meanwhile, the focus word w_i is also represented as a k-dimensional vector $u(w_i)$ of latent topics denoted $t_1 \ldots t_k$ (see Formula 5). Therefore, the composition vector representation $c(w_i)$ of the word w_i is the linear concatenation of the word feature vector $v(w_i)$ and the latent topic feature vector $u(w_i)$ as:

$$c(w_i) = \langle \alpha w_i^1, \ldots, \alpha w_i^n, \beta m_i^1, \ldots, \beta m_i^k \rangle \quad (6)$$

where n is the number of word features and k is the number of latent topics.

VI. Word Semantic Similarity

To evaluate the effects of the proposed word meaning representation, we implement the task of word semantic similarity measure. We also evaluated the word representation using different sets of features: word features using rule-based patterns, word features using window-size, topic

features, and the combination of either the word features and the topic features. The following pre-processing steps were undertaken:

1) *Word Feature Extraction*: Given a focus word w_i, all of the word features w_j were extracted using either the rule-based patterns or window size methods. Each feature was selected by applying the pair frequency filter and the information value filter on its weighting. As the result, the representation of a word using local contexts is described as Formula 1.
2) *Topic Feature Extraction*: Using a topic model as a background model for extracting topics for each word. The topic representation of a word is modelled as Formula 5.
3) *Distance Measure*: To measure the semantic similarity between two words, we directly used the standard *Cosine* distance measure on the representation vectors. Given two words w_i and w_j, the semantic similarity between them is computed as:

$$sim(w_i, w_j) = \frac{v(w_i) \times v(w_j)}{\|v(w_i)\| \times \|v(w_j)\|} \quad (7)$$

VII. Implementation Details

A. Benchmarks

WordSimilarity-353 (WS-353) [9] dataset has been one of the largest publicly available collections for semantic similarity tests. This dataset consists of 353 word pairs annotated by 13 human experts. Their judgement scores were scaled from 0 (unrelated) to 10 (very closely related or identical). The judgements collected for each word pair were averaged to produce a single similarity score. Several studies measured inter-judge correlations and found that human judgement correlations are consistently high $r = 0.88 - 0.95$ [18], [9]. Therefore, the outputs of computer-generated judgments on semantic similarity are expected to be as close as possible the human judgement correlations.

Rubenstein and Goodenough dataset (RG-65) [12] consists of 65 word pairs ranging from synonymy pairs to completely unrelated terms. The 65 noun pairs were annotated by 51 human subjects. All the noun pairs are non-technical words using scale from 0 (not-related) to 4 (perfect synonymy).

MTurk dataset contains 287 pairs of words [3]. Opposite to WS-353, a computer automatically draws the word pairs from words whose frequently occur together in large text domains. The relatedness of these pairs of words was then evaluated using human annotators, as done in the WS-353 dataset. We considered MTurk as a development dataset which was then used to find the range of optimal parameters. The selected parameters were tested on WS-353 and RG-65 datasets.

B. Text Repository

We used Wikipedia English XML dump of October 01, 2012. After parsing the XML dump using Wikiprep [19], [2], we obtained about 13GB of text from $5,836,084$ articles. As we expect to have a large amount of text data to increase the coverage of the method, we used first $1,000,000$ articles for our experiments.

To build the feature representation for each word, we applied the pattern-based extractor to extract pairs of the focus word and its word feature. After the extraction, we obtained $53,653,882$ raw unique pairs which then were normalized by applying the stemming technique [20]. Finally, we obtained $47,143,381$ unique pairs. Similarly, to apply the window size extraction method, we used N-Gram model to extract pairs of words within a windows size of $W = 3$ words from the Wikipedia plain texts after removing stopwords. Then, we also applied the stemming technique [20] to all the extracted words. We finally obtained over $224M$ unique pairs overall.

However, there is the large number of rare pairs with very low frequency. We applied the first frequency filter (FF=2) to remove non-essential word association in pairs. Additionally, we applied the second information value filter (IVF) on each pair. We expect to monitor the influence of IVF on the performance of the similarity measure (see Table II). Only pairs have their information values equal or above the IVF will be retained to form the representation of words.

To extract latent topic features, we used plain texts from the first $100,000$ Wiki documents to feed to LDA training model. The reasons for us to choose this smaller amount of documents as LDA training phrase was time consuming with large amount of documents. We expected to reduce the number of input documents and kept the word dictionary was relatively large to cover most of the expected words. The plain text from these documents was removed stopwords and applied the stemming technique. Rare words was also removed by using document frequency threshold ($df = 5$). We obtained $190,132$ unique words from the given set of documents after pre-processing step. To build the LDA training model, we used GibbsLDA++ [21] with its standard configuration except $ntopic = 1,000$ as the number of expected latent topics.

Parameter Turning: The MTruk dataset was used for parameter turning. We evaluated our method using relational features, topic features, and combination features. After scanning the FF and IVF parameters as well as the $\frac{\alpha}{\beta}$ ratio on this dataset, we obtained the best Spearman's correlation score $\rho = 63$ on both relational features and combination features with $FF = 2$, $IVF = 1$, and $\frac{\alpha}{\beta} = \frac{1}{600}$. The Table I shows the results when the selected parameters were applied as well as the results of other related methods that have been tested on the same dataset. These turning values were used when testing on WS-353 and RG-65 datasets.

C. Evaluation

In this section, we firstly discuss about the effectiveness of word representations over different of semantic similarity

TABLE I
Experiment on MTruk for turning parameters. The best Spearman's correlation score was obtained with $FF = 2$, $IVF = 1$. The related work's results on the same dataset was also presented. The knowledge-based methods are *italic*

Algorithm	$\rho \times 100$
Explicit Semantic Analysis [3]	59
Temporal Semantic Analysis [3]	**63**
Topic features (1000 topics)	46
Word features (pattern-based)	61
Word features (window-based)	**63**
Word + Topic features (pattern-based)	61
Word + Topic features (window-based)	61

TABLE II
THE CORRELATION RESULTS WITH DIFFERENT INFORMATION VALUE FILTER (IVF) TESTED ON WS-353 DATASET USING SPEARMAN'S RANK CORRELATION (ρ). THE BEST RESULTS WERE BOLDED. THE RESULTS WITH UNDERLINE WERE USING PARAMETERS SELECTED FROM THE DEVELOPMENT DATASET

IVF	$\rho \times 100$			
	Word features (pattern)	Word-topic features (pattern)	Word features (window)	Word-topic features (window)
-3.0	60.58	62.97	58.95	62.72
-2.5	60.76	63.05	59.01	62.75
-2.0	61.05	63.36	59.32	63.09
-1.5	62.06	64.31	60.37	64.01
-1.0	63.49	65.32	62.39	66.19
-0.5	64.34	65.82	63.31	67.05
0.0	63.73	65.07	61.80	67.91
0.5	66.48	67.29	66.67	69.76
1.0	**69.42**	70.19	**71.09**	73.36
1.5	68.30	**70.79**	70.47	**73.67**
2.0	64.60	70.12	67.14	72.74
2.5	49.19	66.39	56.23	69.25
3.0	26.93	55.94	38.78	48.48

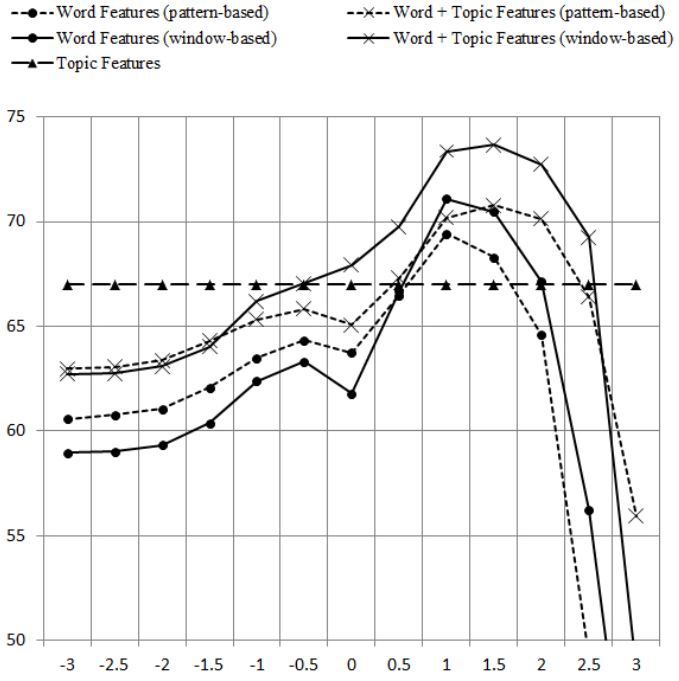

Fig. 1. The visualisation of experiment results from WS-353 dataset (see Table II). The combination feature-based method outperformed the one using word features regardless IVF.

standard datasets. Firstly, Figure 1 shows the experimental results over four kinds of features outperform to latent topic features on the task of semantic similarity using WS-353 dataset. This also support the hypothesis that distributional representation using word in contexts produces better discrimination between words than one using latent topics inferred on the same text corpus. Moreover, information from Figure 1 and Table II has concluded that the combination between word features and topic features produces effective results on the task of semantic representation as well as semantic similarity measure.

It is notable to compare the performance of the proposed method to other related work on the same benchmarks (see Table III). On the standard WS-353 dataset, our method outperforms to most of the semantic similarity methods (corpus-based methods) using single VSM for word representation.

Additionally, the proposed method achieves the promising performance on RG-65 dataset on both word features and combination features. Interestingly, the topic feature-based method on Wikipedia outperforms to most of the other latent topic feature-based methods such as LSA and LDA on both WS-353 and RG-65 datasets. This also confirms the benefits of using a rich text repository when constructing representation of words.

Finally, in comparison to the work done by [6], one of the closest approaches to our work in term of feature engineering, the proposed method outperformed on both WS-353 and RG-65 datasets.

TABLE III
THE COMPARISON RESULTS WITH DIFFERENT CONTENT-BASED METHODS ON WS-353 AND RG-65 DATASETS USING SPEARMAN'S RANK CORRELATION (ρ).THE KNOWLEDGE-BASED METHODS ARE ITALIC. (†) NOTED USING PARAMETERS FROM THE DEVEOPMENT DATASET. ($\star$) NOTED THE BEST RESULTS IN OUR EXPERIMENTS

Algorithm	$\rho \times 100$	
	WS-353	RG-65
Syntactic Features [6]	34.80	78.8
Latent Topic Features (LSA) [9]	58.10	60.9
Latent Topic Features (LDA) [8]	53.39	–
Multi-Prototype [22]	**76.9**	–
Single-Prototype [22]	55.3	–
Multi-Prototype [23]	71.3	–
Learned Features [10]	49.86	–
Context Window Pattern (WS=1) [5]	69	89
Context Window Pattern (WS=4) [5]	66	**93**
Topic Features	67	63.93
Word Features (pattern-based)†	69.42	79.72
Word Features (window-based) †	71.09	79.56
Topic + Word Features (pattern-based)†	70.19	79.16
Topic + Word Features (window-based)†	73.52	78.82
Word Features*	71.09	84.43
Word + Topic Features*	73.67	84.59

VIII. RELATED WORK

Previous work in the field of semantic similarity is categorized as corpus-based and knowledge-based approaches. While the corpus-based methods utilize statistical techniques to measure the similarity between words using the pure text content of a given corpus, the knowledge-based approaches explore the embedded knowledge from a large repository such as WordNet, networks of concepts from Wikipedia.

VSMs are mostly used to model the meanings of words. In the knowledge-based approaches, Gabrilovich et. al. have proposed Explicit Semantic Analysis (ESA) [2], which represents word meanings as a vector of explicit Wikipedia concepts. The relatedness between words is measured by the distance between the respective vectors. Silent Semantic Analysis (SSA) was proposed by Hassan et. al [4]. SSA explores Wikipedia silent concepts which were then incorporated with the explicit Wikipedia concepts to model the word representation using VSMs. One of the main differences between these methods and our approach is the way of estimating the degree of association between words. In ESA and SSA, word-word relations are defined indirectly using their relationship with Wikipedia concepts. However, the relation between words in our approaches is defined directly using

the common relational participants within local contexts as well as their common latent topics.

Different from the knowledge-based methods, the content-based methods are purely relied on plain text. Latent Semantic Analysis (LSA) [16] was proposed to take into account word-document associations to present the semantic representation of words. LSA considers meanings of a word as a vector of latent topics and the similarity between words is measured by the distance of its represented vectors. Similarly, topic model Latent Dirichlet Allocation (LDA) [8] was used to to measure word semantic similarity. The fundamental idea that documents are mixtures of topics where a topic is a probability distribution over words. The similarity of words could be inferred by the associated of their common topics.

Agirre et. al used word patterns in context windows as the features. The method produced promising correlation results in RG-65 dataset and considerable results on WS-353 dataset with Windowsize (WS=1 and WS=4) [5]. Lin et. al. [6] measured the similarity between words using the distributional lexical and syntactic patterns of words over a parsed corpus. The similarity between a pair of words was measured by the common between their distributions. The idea of feature engineering in this work is quite similar to our approach that using the local contexts to extract relations between words. However, while these authors considered syntactic associations between a focus word and its adjacent words to produce the word's representation. We combined relational features and topic features to form a representation of words. Moreover, to reduce the influences of the noise in the semantic similarity measure, we applied different filters to retain information valuable relations. This has contributed to leverage the performance of our proposed method.

Recent work on feature learning has opened a new way of building word semantic representation automatically from the nature of language. Collobert et. al. [10] proposed a deep learning framework for automatically building word meaning representations (word embeddings). Huang et. al. [23] have successfully inherited the word embeddings to learn multiple word prototypes (multiple VSM represented for meanings of a word), which show the promising results on the task of semantic similarity. Similarly, Reisinger et. al. [22] have proposed multi-prototype VSM for word meaning representation using text clustering. The method presents significant improvement performance on semantic similarity measure. However, they also confirmed that single word prototype is still having issues in gaining the performance of content-based semantic similarity measure.

IX. Conclusion

We have presented our work on building the representation of words using contextual analysis. The method takes into account the relations between words in local contexts and latent topics information from global contexts. The experimental results have shown the positive contribution of word features as well as their combinations with topic features on the task of semantic representation and also the task of semantic similarity on standard datasets.

References

[1] P. D. Turney, "Mining the web for synonyms: Pmi-ir versus lsa on toefl," in *Proceedings of the 12th European Conference on Machine Learning*, 2001, pp. 491–502.

[2] E. Gabrilovich and S. Markovitch, "Computing semantic relatedness using wikipedia-based explicit semantic analysis." in *IJCAI*, vol. 7, 2007, pp. 1606–1611.

[3] K. Radinsky, E. Agichtein, E. Gabrilovich, and S. Markovitch, "A word at a time: computing word relatedness using temporal semantic analysis," in *Proceedings of the 20th international conference on World wide web*. ACM, 2011, pp. 337–346.

[4] S. Hassan and R. Mihalcea, "Semantic relatedness using salient semantic analysis." in *AAAI*, 2011.

[5] E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Paşca, and A. Soroa, "A study on similarity and relatedness using distributional and wordnet-based approaches," in *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 2009, pp. 19–27.

[6] D. Lin, "An information-theoretic definition of similarity." in *ICML*, vol. 98, 1998, pp. 296–304.

[7] E. Agirre, M. Cuadros, G. Rigau, and A. Soroa, "Exploring knowledge bases for similarity." in *LREC*, 2010.

[8] G. Dinu and M. Lapata, "Measuring distributional similarity in context," in *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2010, pp. 1162–1172.

[9] L. Finkelstein, E. Gabrilovich, Y. Matias, E. Rivlin, Z. Solan, G. Wolfman, and E. Ruppin, "Placing search in context: The concept revisited," in *Proceedings of the 10th international conference on World Wide Web*. ACM, 2001, pp. 406–414.

[10] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural language processing (almost) from scratch," *The Journal of Machine Learning Research*, vol. 12, pp. 2493–2537, 2011.

[11] D. Huynh, D. Tran, and W. Ma, "Combination features for semantic similarity measure," in *Lecture Notes in Engineering and Computer Science: Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 1, 2014, pp. 324–327.

[12] H. Rubenstein and J. B. Goodenough, "Contextual correlates of synonymy," *Communications of the ACM*, vol. 8, no. 10, pp. 627–633, 1965.

[13] E. A. Nida, *Componential analysis of meaning*. Mouton The Hague, 1975.

[14] M. Sahlgren, "The word-space model: Using distributional analysis to represent syntagmatic and paradigmatic relations between words in high-dimensional vector spaces," Ph.D. dissertation, Stockholm, 2006.

[15] I. Dagan, S. Marcus, and S. Markovitch, "Contextual word similarity and estimation from sparse data," in *Proceedings of Association for Computational Linguistics*. Association for Computational Linguistics, 1993, pp. 164–171.

[16] S. T. Dumais, "Latent semantic analysis," *Annual review of information science and technology*, vol. 38, no. 1, pp. 188–230, 2004.

[17] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[18] A. Budanitsky and G. Hirst, "Evaluating wordnet-based measures of lexical semantic relatedness," *Computational Linguistics*, vol. 32, no. 1, pp. 13–47, 2006.

[19] E. Gabrilovich and S. Markovitch, "Overcoming the brittleness bottleneck using wikipedia: Enhancing text categorization with encyclopedic knowledge," in *AAAI*, vol. 6, 2006, pp. 1301–1306.

[20] C. J. Van Rijsbergen, S. E. Robertson, and M. F. Porter, *New models in probabilistic information retrieval*. Computer Laboratory, University of Cambridge, 1980.

[21] X.-H. Phan, L.-M. Nguyen, and S. Horiguchi, "Learning to classify short and sparse text & web with hidden topics from large-scale data collections," in *Proceedings of the 17th international conference on World Wide Web*. ACM, 2008, pp. 91–100.

[22] J. Reisinger and R. J. Mooney, "Multi-prototype vector-space models of word meaning," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 2010, pp. 109–117.

[23] E. H. Huang, R. Socher, C. D. Manning, and A. Y. Ng, "Improving word representations via global context and multiple word prototypes," in *Proceedings of Association for Computational Linguistics*. Association for Computational Linguistics, 2012, pp. 873–882.