**IJCLA VOL. 5, NO. 1, JAN-JUN 2014, PP. 11–25**
**RECEIVED 06/01/14 ACCEPTED 06/02/14 FINAL 19/06/14**

# Semantic Similarity Measure Using Relational and Latent Topic Features

DAT HUYNH, DAT TRAN, WANLI MA, AND DHARMENDRA SHARMA

*University of Canberra, Australia*

ABSTRACT

*Computing the semantic similarity between words is one of the key challenges in many language-based applications. Previous work tends to use the contextual information of words to disclose the degree of their similarity. In this paper, we consider the relationships between words in local contexts as well as latent topic information of words to propose a new distributed representation of words for semantic similarity measure. The method models meanings of a word as high dimensional Vector Space Models (VSMs) which combine relational features in word local contexts and its latent topic features in the global sense. Our experimental results on popular semantic similarity datasets show significant improvement of correlation scores with human judgements in comparison with other methods using purely plain texts.*

## 1 INTRODUCTION

In many language-based applications, it is crucial to be able to measure precisely the semantic similarity between words. VSMs have been used to represent word meanings in a vector that captures semantic and syntactic information of the word. Generated latent topics from a large plain text corpus have been used as vector features for semantic similarity measure [1, 2]. Syntactic/lexical patterns of word (word pairs) in local contexts have also been used as vector features for the similarity measure [3–5].

In this work, we utilize a large plain text corpus as the knowledge-domain to propose a new set of features for semantic similarity task. The

feature set is extracted by considering relational participants (features) surrounding a focus word and its latent topic features over a large plain text corpus. Therefore, a VSM representation of a word is modelled as a high dimensional vector of the combination of relational features and latent topic features. We have developed parameters of the combined features on the MTurk dataset [6] and tested on the popular semantic similarity datasets such as WS-353 [7] and RG-65 [8]. The experimental results confirm the significant improvement of our proposed method on semantic similarity measure in comparison to other corpus-based methods tested on the same datasets.

The paper is organized as follows: We first present the construction of distributed representation in Section 2. In Section 3, the task of word similarity measure is described. In Section 4, our experimental setups and results are discussed. Finally, the related work on semantic similarity measure is presented in Section 5.

## 2 Semantic Distributed Representation

Meanings of a word can be inferred from its surround contexts. Consider the following example describing the contexts of an unknown word "*tezgüino*" (the modified example from [9, 5]).

- 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*", gets people "*drunk*" and normally contains in "*bottle*". In other words, the meanings of a given word could be disclosed by considering its relational participants in local contexts such as "*bottle*", "*strong*", "*drunk*", and "*corn*". This intuitive idea is also the motivation for building the relation-based distributed representation.

### 2.1 *Meaning Representation Using Relational Word Features*

It has been confirmed that meanings of a word is determined by its surrounding contexts [3]. The surrounding contexts include relational associations between the word and others in contexts. While some relational associations hold the meanings over long distance, such as in the

pairs ("*tezgüino*", "*drunk*") and ("*tezgüino*", "*corn*"), others maintain the meanings when the word interacts with its adjacent neighbours, such as in the pairs ("*tezgüino*", "*strong*") and ("*tezgüino*", "*bottle*"). Given a word $w_i$, its semantic representation $v(w_i)$ is described as a sparse vector as follows:

$$v(w_i) = \langle w_i^1, w_i^2, \ldots, w_i^n \rangle \tag{1}$$

where $w_i^k$ is the information value that reflects the degree of semantic association between the word $w_i$ and its relational participant $w_k$. The parameter $n$ is the size of word dictionary in the given text corpus. Futhermore, different corpus-based approaches come up with different information value measures. We used the point-wise mutual information (PMI) [10] to measure the degree of information value (association) between two different words in a relation. The information value $w_i^k$ of the pair of words $(w_i, w_k)$ is measured as follows:

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

$$p(w_i, w_k) = \frac{d(w_i, w_k)}{\sum\limits_{i,k=1\ldots n} d(w_i, w_k)} \tag{3}$$

$$p(w_i) = \frac{\sum\limits_{k=1\ldots n} d(w_i, w_k)}{\sum\limits_{i,k=1\ldots n} d(w_i, w_k)} \tag{4}$$

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

### 2.2 *Representation Using Latent Topic Features*

Word meanings have been successfully described using explicit topics such as Wikipedia concepts [11]. 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), and [1],

Latent Dirichlet Allocation (LDA) [2]. 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. On the task of semantic similarity measure, LDA has been confirmed for the better results than LSA [12]. 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 $t_j$ ($j = 1 \dots k$).

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

$$u(w_i) = \langle m_i^1, m_i^2, \dots, m_i^k \rangle \tag{5}$$

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.

### 2.3 *Word Representation Using Combination of Relational Features and Latent 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 $w_1 \dots w_n$ (see formula 1). Meanwhile, the focus word $w_i$ is also represented as a $k$-dimensional vector $u(w_i)$ of latent topics denoted $t_1 \dots t_k$ (see formula 5). Therefore, the composition vector representation $c(w_i)$ of the word $w_i$ is the linear concatenation of the relational feature vector $v(w_i)$ and the latent topic feature vector $u(w_i)$ as:

$$c(w_i) = \langle \alpha w_i^1, \dots, \alpha w_i^n, \beta m_i^1, \dots, \beta m_i^k \rangle \tag{6}$$

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

## 3 SEMANTIC WORD SIMILARITY

Our proposed content-based method of measuring semantic similarity was constructed using two different groups of features: relational words

in context and latent topics. These groups of features were tested separately and collectively. The following pre-processing steps were undertaken:

1. *Relation Extraction*: Relations surrounding words in contexts need to be extracted from a plain text repository. We designed a pattern-based extractor which single-passes through the plain texts and returns the extractions. Each extraction is a pair of a focus word and its relational participant, which have to match the following conditions:
   (a) The relational participant has to be a single noun, compound noun, or a name entity
   (b) If existed, the sequence in between the pair from the text has to match the following pattern:

   **V+ | V+W*P | P**

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

   These rules are expected to cover most of the local and long distance association between words in contexts.
2. *Word Representation*: Each focus word is represented by a set of relational participants. To reduce the number of relational associations, we retained those having considerable information value. Therefore, we applied a first filter on the relation frequency and a second filter on information value for each relation. There are three ways to construct the VSM of a word: relational feature VSM, latent topic feature VSM and combination feature VSM.
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:

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

## 4 Implementation Details

In this section we describe the implementation of our system.

**Table 1.** Experiment on MTruk for tuning parameters. The best Spearman's correlation score was obtained with $FF = 2$, $IVF = 1$, and $\frac{\alpha}{\beta} = \frac{1}{600}$ on both relational features and combination features. 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 [6]* | 59 |
| *Temporal Semantic Analysis [6]* | **63** |
| Relational feature | **63** |
| Topic features | 46 |
| Combination Feature | **63** |

### 4.1 *Benchmarks*

WordSimilarity-353 (WS-353) [7] 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 $r = 0.88 - 0.95$ [13, 7]. Therefore, the outputs of computer-generated judgments on semantic similarity are expected to be as close as possible to the human judgement correlations.

Rubenstein and Goodenough dataset (RG-65) [8] consists of 65 word pairs ranging from synonym 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 synonym).

MTurk[1] dataset contains 287 pairs of words [6]. 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.

[1] http://www.technion.ac.il/∼kirar/Datasets.html

**Table 2.** The correlation results with different information value filter (IVF) tested on WS-353 dataset using Spearman's rank correlation ($\rho$). The best results were bolded. The results with underline were using parameters selected from the development dataset

| IVF | Word features ($\rho \times 100$) | Combination features ($\rho \times 100$) | Topic features ($\rho \times 100$) |
|---|---|---|---|
| $-3.0$ | 60.58 | 62.97 | 66.78 |
| $-2.5$ | 60.76 | 63.05 | |
| $-2.0$ | 61.05 | 63.36 | |
| $-1.5$ | 62.06 | 64.31 | |
| $-1.0$ | 63.49 | 65.32 | |
| $-0.5$ | 64.34 | 65.82 | |
| 0.0 | 63.73 | 65.07 | |
| 0.5 | 66.48 | 67.29 | |
| 1.0 | **69.42** | 70.19 | |
| 1.5 | 68.30 | **70.79** | |
| 2.0 | 64.60 | 70.12 | |
| 2.5 | 49.19 | 66.39 | |
| 3.0 | 26.93 | 55.94 | |

## 4.2 *Text Repository*

We used Wikipedia English XML dump of October 01, 2012. After parsing the XML dump[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 relational feature representation for each word, we applied the pattern-based extractor to extract pairs of the focus word and its relational participant. After the extraction, we obtained $53,653,882$ raw unique pairs which then were normalized by applying the stemming technique [14]. Finally, we obtained $47,143,381$ unique relations between words and their relational paritipants.

As there were a large number of rare words and pairs associated with each focus word, we applied two filters to leave out those we believed as noise. While the relation frequency filter (FF) is reponsible to remove rare relational pairs, the information value filter (IVF) is expected to leave out pairs with low information value. Any pair with their respective informa-

[2] We used Wikiprep as the main tool to convert Wikipedia format to XML plain text, http://sourceforge.net/projects/wikiprep/

**Table 3.** The correlation results with different information value filter (IVF) tested on 65 pairs of RG-65 dataset using Spearman's rank correlation ($\rho$). The best results were bolded. The results with underline were using parameters selected from the development dataset

| IVF | Word features ($\rho \times 100$) | Combination features ($\rho \times 100$) | Topic features ($\rho \times 100$) |
|---|---|---|---|
| $-3.0$ | 73.64 | 72.47 | 63.93 |
| $-2.5$ | 73.62 | 72.25 | |
| $-2.0$ | 73.74 | 72.58 | |
| $-1.5$ | 74.50 | 72.91 | |
| $-1.0$ | 75.25 | 73.96 | |
| $-0.5$ | 76.65 | 75.89 | |
| 0.0 | 77.12 | 76.53 | |
| 0.5 | 77.63 | 77.09 | |
| 1.0 | 79.72 | 79.16 | |
| 1.5 | 84.11 | 83.59 | |
| 2.0 | **84.43** | **84.59** | |
| 2.5 | 78.46 | 83.33 | |
| 3.0 | 59.64 | 79.88 | |

tion is equal or above the filter's threshold will be retained to contruct 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 relatively large to cover most of the expected words. The plain text from these documents was removed stop-words and applied the stemming technique. Rare words were 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++[3] [15] with its standard configuration except $ntopic = 1,000$ as the number of expected latent topics.

**Parameter Tuning:** The MTruk dataset was used for parameter tuning. 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 corre-

[3] http://gibbslda.sourceforge.net

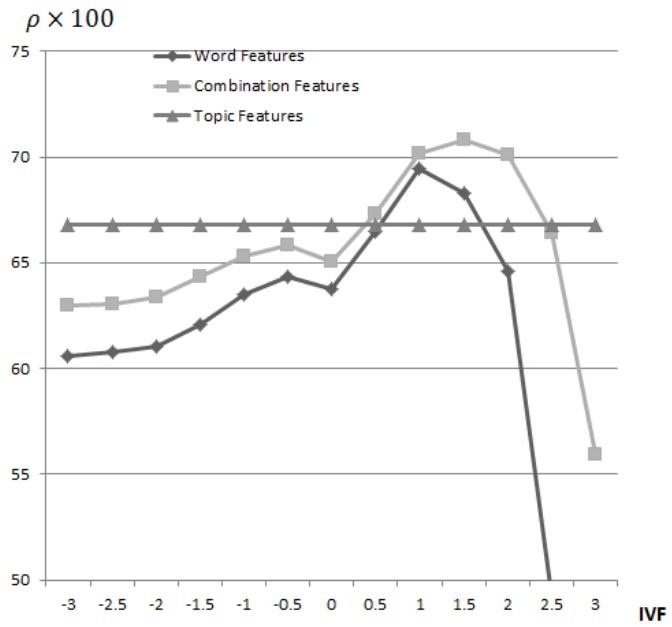


**Fig. 1.** The visualisation of experimental results from WS-353 dataset (see Table 2). The combination feature-based method outperformed the one using word features regardless IVF.

lation score $\rho = 63$ on both relational features and combination features with $FF = 2$, $IVF = 1$, and $\frac{\alpha}{\beta} = \frac{1}{600}$. Table 1 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 tuning values were used when testing on WS-353 and RG-65 datasets.

### 4.3 *Evaluation*

In this section [4], we firstly discuss about the effectiveness of our method over different of standard datasets. Table 2 and 3 show the results of our experiments over three kinds of features. Overall, the method based on relational features outperformed those using topic features on WS-353

[4] The experimental results can be found at http://137.92.33.34/CICLING2014/

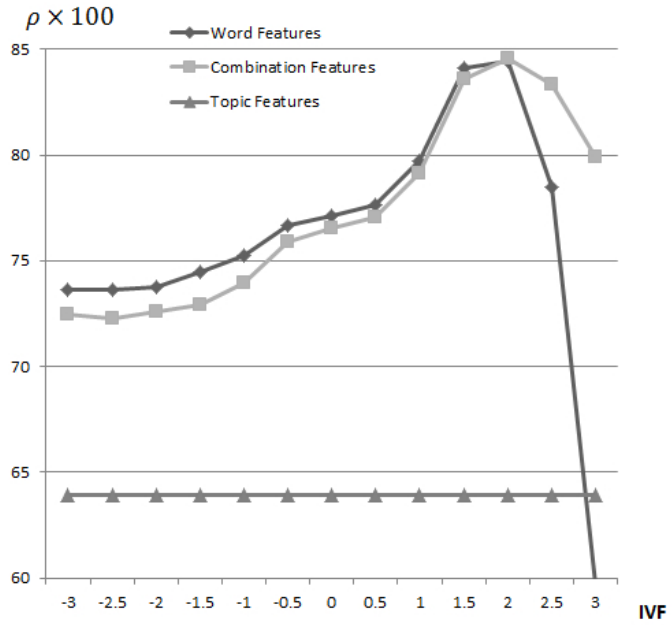


**Fig. 2.** The visualisation of experimental results on RG-65 dataset (see Table 3). The method using combination features is comparable the one based on word features.

dataset (69.42 vs. 66.78) and on RG-65 dataset (84.43 vs. 63.93). Particularly, when the relational features are combined with topic features in a single VSM, the performance of the combination method was improved in comparison with those using the type of features separately.

Moreover, Table 2 and 3 also confirms that the selected parameters from the development dataset potentially work really well on the WS-353 and RG-65 datasets. They produced significant improvement compared to those using the similar kinds of features (see Table 4).

It is notable to compare the performance of the proposed method to other related work on the same benchmarks. On the standard WS-353 dataset, our method outperforms to most of the semantic similarity methods using single VSM for word representation. Compare to other corpus-based methods in general, our approach also achieves the second high-

**Table 4.** The comparison results with different content-based methods on WS-353 and RG-65 datasets using Spearman's rank correlation ($\rho$). The knowledge-based methods are in italic. (†) denotes using parameters from the deveopment dataset. ($\star$) denotes the best results in our experiments

| Algorithm | WS-353 ($\rho \times 100$) | RG-65 ($\rho \times 100$) |
|---|---|---|
| Syntactic Features [5] | 34.80 | 78.8 |
| Latent Topic Features (LSA) [7] | 58.10 | 60.9 |
| Latent Topic Features (LDA) [12] | 53.39 | – |
| Multi-Prototype [16] | **76.9** | – |
| Single-Prototype [16] | 55.3 | – |
| Multi-Prototype [17] | 71.3 | – |
| Learned Features [18] | 49.86 | – |
| Context Window Pattern ($\mathrm{WS} = 1$) [4] | 69 | 89 |
| Context Window Pattern ($\mathrm{WS} = 4$) [4] | 66 | **93** |
| Topic Features | 66.78 | 63.93 |
| Relational Features$^{\dagger}$ | 69.42 | 79.72 |
| Combination Features$^{\dagger}$ | 70.19 | 79.16 |
| Relational Features$^{\star}$ | 69.42 | 84.43 |
| Combination Features$^{\star}$ | 70.79 | 84.59 |

est correlation score on this dataset after the multi-prototype VSM done by [16].

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.

Finally, in comparison to the work done by [5], 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.

## 5 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 meaning of words. In frame of knowledge-base approaches, Gabrilovich et al. have proposed Explicit Semantic Analysis (ESA) [11], 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. [19]. 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.

In contrast to the knowledge-based methods, the content-based methods rely only on plain text. Latent Semantic Analysis (LSA) [1] 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) [12] 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) [4]. Lin et al. [5] 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. [18] proposed a deep learning framework for automatically building word meaning representations (word embeddings). Huang et al. [17] 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. [16] 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.

## 6 CONCLUSION

We have presented an approach for semantic similarity measure using relational features and topic features. 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 relational features and topic features to the performance of corpus-based methods. Especially, their combination in modelling word representation yields the improvement results to most of the content-based methods on both tested datasets.

## REFERENCES

1. Dumais, S.T.: Latent semantic analysis. Annual review of information science and technology **38**(1) (2004) 188–230
2. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. The Journal of Machine Learning Research **3** (2003) 993–1022
3. Turney, P.D.: Mining the web for synonyms: PMI-IR versus LSA on TOEFL. In: Proceedings of the 12th European Conference on Machine Learning. (2001) 491–502
4. Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., Soroa, A.: 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) 19–27

5. Lin, D.: An information-theoretic definition of similarity. In: ICML. Volume 98. (1998) 296–304
6. Radinsky, K., Agichtein, E., Gabrilovich, E., Markovitch, S.: 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) 337–346
7. Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., Ruppin, E.: Placing search in context: The concept revisited. In: Proceedings of the 10th International Conference on World Wide Web, ACM (2001) 406–414
8. Rubenstein, H., Goodenough, J.B.: Contextual correlates of synonymy. Communications of the ACM **8**(10) (1965) 627–633
9. Nida, E.A.: Componential analysis of meaning. Mouton The Hague (1975)
10. Dagan, I., Marcus, S., Markovitch, S.: Contextual word similarity and estimation from sparse data. In: Proceedings of Association for Computational Linguistics, Association for Computational Linguistics (1993) 164–171
11. Gabrilovich, E., Markovitch, S.: Computing semantic relatedness using Wikipedia-based explicit semantic analysis. In: IJCAI. Volume 7. (2007) 1606–1611
12. Dinu, G., Lapata, M.: Measuring distributional similarity in context. In: Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics (2010) 1162–1172
13. Budanitsky, A., Hirst, G.: Evaluating WordNet-based measures of lexical semantic relatedness. Computational Linguistics **32**(1) (2006) 13–47
14. Van Rijsbergen, C.J., Robertson, S.E., Porter, M.F.: New models in probabilistic information retrieval. Computer Laboratory, University of Cambridge (1980)
15. Phan, X.H., Nguyen, L.M., Horiguchi, S.: 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) 91–100
16. Reisinger, J., Mooney, R.J.: 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) 109–117
17. Huang, E.H., Socher, R., Manning, C.D., Ng, A.Y.: Improving word representations via global context and multiple word prototypes. In: Proceedings of Association for Computational Linguistics, Association for Computational Linguistics (2012) 873–882
18. Collobert, R., Weston, J., Bottou, L., Karlen, M., Kavukcuoglu, K., Kuksa, P.: Natural language processing (almost) from scratch. The Journal of Machine Learning Research **12** (2011) 2493–2537
19. Hassan, S., Mihalcea, R.: Semantic relatedness using salient semantic analysis. In: AAAI. (2011)

**DAT HUYNH**
UNIVERSITY OF CANBERRA,
BRUCE, ACT 2617, AUSTRALIA
E-MAIL: <DAT.HUYNH@CANBERRA.EDU.AU>

**DAT TRAN**
UNIVERSITY OF CANBERRA,
BRUCE, ACT 2617, AUSTRALIA
E-MAIL: <DAT.TRAN@CANBERRA.EDU.AU>

**WANLI MA**
UNIVERSITY OF CANBERRA,
BRUCE, ACT 2617, AUSTRALIA
E-MAIL: <WANLI.MA@CANBERRA.EDU.AU>

**DHARMENDRA SHARMA**
UNIVERSITY OF CANBERRA,
BRUCE, ACT 2617, AUSTRALIA
E-MAIL: <DHARMENDRA.SHARMA@CANBERRA.EDU.AU>