International Journal of Fuzzy Logic and Intelligent Systems, vol. 11, no. 4, December 2011, pp. 238-246
http://dx.doi.org/10.5391/IJFIS.2011.11.4.238

A Semantic Representation Based-on Term Co-occurrence Network and Graph Kernel

Tae-Gil Noh, Seong-Bae Park[a], Sang-Jo Lee

School of Computer Science and Engineering, Kyungpook National University, Daegu, 702-701, Republic of Korea

Abstract

This paper proposes a new semantic representation and its associated similarity measure. The representation expresses textual context observed in a context of a certain term as a network where nodes are terms and edges are the number of co-occurrences between connected terms. To compare terms represented in networks, a graph kernel is adopted as a similarity measure. The proposed representation has two notable merits compared with previous semantic representations. First, it can process polysemous words in a better way than a vector representation. A network of a polysemous term is regarded as a combination of sub-networks that represent senses and the appropriate sub-network is identified by context before compared by the kernel. Second, the representation permits not only words but also senses or contexts to be represented directly from corresponding set of terms. The validity of the representation and its similarity measure is evaluated with two tasks: synonym test and unsupervised word sense disambiguation. The method performed well and could compete with the state-of-the-art unsupervised methods.

Key Words: Semantic Representation, Semantic Relatedness Measure, Graph Kernel, Word Sense Disambiguation

1. Introduction

Semantic representation is important for computers to handle various natural language processing tasks. For example, Information Retrieval (IR) is a task that retrieves documents that are semantically related to queries. Machine Translation is also a task that needs to find semantically equivalent translation. Other tasks like document classifications, or summarizations all uses some sort of semantic representation directly or indirectly. Thus, various work has been done on the topic of semantic representations. Existing semantic representations can be generally classified into one of three groups [1]: semantic network, spatial representation, and topic model.

Semantic networks represent the meaning of words by networks where nodes are words and edges are relationships among them. While sophisticated semantic networks like WordNet are valuable in many tasks, the cost of building such resources is high. In addition, they are often not available for minor languages.

Spatial representations, including vector-space model of classical IR and LSA-like rank reduced representations, are very popular and widely used. In spatial representations, documents or terms are expressed as vectors: vectors in a term-document space, or in a latent semantic space. One problem of the spatial representations is polysemous words. Spatial representations generally cannot capture polysemy directly, because they represent each word as a single point in the space.

The third group of semantic representation is topic model. In topic models [2, 3], the meaning of a term (or a document) is represented as a probability mixture of latent topics. Since a word is represented as a mixture of topics instead of a superposition of vectors, a polysemous word can be resolved more profoundly within the model. However, the fact that topic models can handle some polysemous words does not mean that the topic models can capture word *senses*. They can resolve ambiguous words only that have been captured at the right resolution of topics. The number of topics is generally several hundreds. Meanwhile, a typical sense inventory has several thousands or more senses. If a given sense is finer than the topic resolution, the sense cannot be captured by the topic model.

The representation of this paper is an attempt to cope with these issues of existing semantic representations. The proposed representation is a type of spatial representation. However, it does not directly represent a term as a point in space. Instead, it regards a structure obtained from co-occurrence data as a semantic representation of a term. Since co-occurrence data naturally form a network, the co-occurrence network is used as the structure to represent a term. An R-convolution kernel is then introduced to com-

[a]Corresponding Author.
Manuscript received Aug. 23, 2011; revised Nov. 3, 2011; accepted Nov. 4, 2011.

pare two structures to calculate the similarity between representations in an infinitely high dimensional space.

The representation and its associated similarity measure can process polysemous words in a profound way. Unlike vectors, a network is not a single point. In the proposed representation, a polysemous term is regarded as a combination of sub-networks where each sub-network represents different senses. Therefore, it is possible to identify an appropriate sub-network that is activated by context, and this sub-network can be used to resolve the polysemous term. Additionally, the representation permits not only words but also senses to be represented directly as networks originated from corresponding synsets. A specific sense can be generated by combining common sub-structures among words in a synonym set (*synset*) that represents the sense. Since synsets are a widely accepted way to represent word senses, this can be a practical way to represent word senses from statistical co-occurrence data. Another advantage of the approach is that it does not need sophisticated resources like WordNet relations. It only needs a large unlabeled text collection as LSA or topic models do.

2. Related Works

Good semantic representations can greatly enhance performance of natural language processing tasks. Landauer et al. [4] showed that term-document vectors often fail to solve tasks related with semantics like synonym test, and proposed the well-known Latent Semantic Analysis (LSA). LSA reduces the dimension of vectors, and compares terms and documents in the reduced latent space. In effect, the reduced space can reflect higher order co-occurrences.

Reducing dimensionality is not the only way to reflect higher order co-occurrences. Second-order or higher order co-occurrences can be directly calculated [5] or can be obtained by random walks on WordNet-like ontology networks [6] or by replacing latent space of LSA to explicit topics of Wikipedia entries [7]. Those variations have reported equal or better result than the original LSA. However, they can be still regarded as spatial models where a term is represented as a point.

Topic models [3, 2] are statistical models that assumes a text is a probability mixture of hidden topics. While topic models capture the polysemous use of words, they do not carry the explicit notion of senses. To capture senses, topic models need some additional resources or models. For example, Boyd-Graber et al. [8] proposed a generative model that combines a topic model with a WordNet walk model.

Recently, several graph-based approaches have been applied to word sense induction (WSI) tasks [9, 10]. Typically, network-based WSI methods cluster nodes and edges of a co-occurrence network, where each cluster is then corresponding to one induced sense. To distinguish the sense of a polysemous word, terms observed in the context of the polysemous word are compared with the terms observed in each cluster.

One important difference between previous network-based WSI methods and the proposed approach is that how the co-occurrence network is used. In network-based WSI methods, resulting clusters are converted to some other representation (weighted tree, weighted vector, etc) to be compared with the terms of context. In general, they do not treat a network itself as a representation, and their networks cannot be directly compared, and they cannot be used to solve semantic relatedness task like synonym tests. Also, since they are a sense induction method, they need some additional mapping step to map the induced senses (clusters) to predefined senses if they are to be employed in WSD setup. This is a common problem of sense disambiguation methods based-on clustering methods [11].

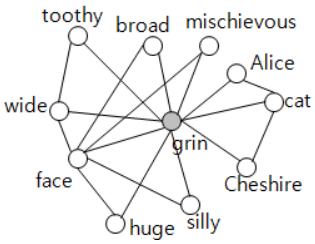
Another related previous work are methods that use random walks on WordNet to calculate semantic relatedness [12]. They calculate the semantic relatedness between two terms by calculating probability of ending up in the same node in the WordNet network by long random walks. This calculation method is similar to the famous PageRank algorithm, but the goal of this calculation is measuring relatedness, not ranking.

While such methods are related to the approach of this paper, there are some deep differences: First, the proposed representation of this paper is based on networks from an unlabeled corpus, not from sophisticated lexical database like WordNet. Moreover, in this work, a network corresponding to a specific term is regarded as a semantic representation which can possibly replace a vector representation. Thus, not only a similarity measure, but also network operations like finding sub-structures or combining two networks are considered and implemented in this work.

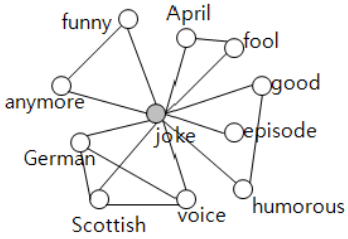
In our own previous work, a graph kernel space has been used to enhance disambiguation problems like lexical translations [13]. Compared to our previous work, the proposed method of this paper is more general, and not limited to lexical translations. Here, the co-occurrence network is regarded as a basic representation, which can replace general vector based representations. This paper shows that network representations can be used in more general NLP tasks like synonym detection. The goal of this paper is not to compete with a specific state-of-the-art (although the experiments shows a result comparable to the state-of-the-arts), but to compare the performance of the two representations: namely, the vector representation and the network representation.

3. Basic Idea of the Network Representation

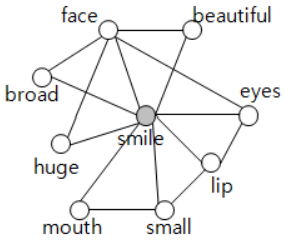
Co-occurrences of words can be understood as a net-



(a) grin



(b) joke



(c) smile

Figure 1: Three Networks of term “grin”, “joke” and “smile”.

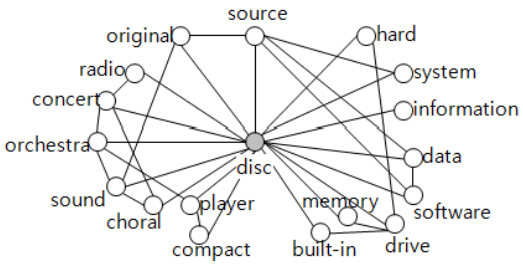
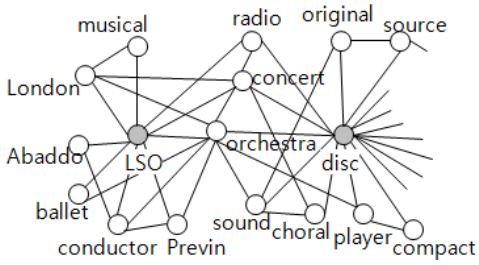
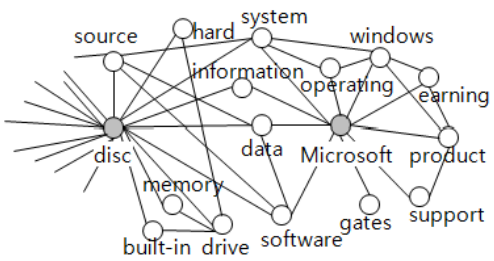


Figure 2: The network of term “disc”.



(a) disc-LSO



(b) disc-Microsoft

Figure 3: Two expanded networks of “disc”.

work. In the network, a node represents a term, and an edge between two nodes is made when the terms of the nodes co-occur. A term can be expressed as a fragment of the co-occurrence network where the term is the central node. For example, Figure 1 shows an actual co-occurrence network observed from BNC corpus. For clarity, only a very small number of nodes and edges are shown in the figure, and weight values on the edges are omitted.

The basic assumption of the proposed approach is that if two terms were semantically similar, their corresponding network parts would be similar. In the figure, the network similarity of “grin” and “smile” will be higher than that of “grin” and “joke”. Common edges and common cliques are found between two networks of “grin” and “smile”, but they are absent between those of “grin” and “joke”.

Polysemous words have more than one sense. In this work, a network corresponding to a polysemy is regarded as an aggregation of several sub-networks. Figure 2 shows a network whose center is the term “disc”. The left side of this network is a part in which the sense of “disc” is phonograph or music album. The right part of the network represents a sub-network as a magnetic disc or memory device used in computers.

The second assumption of this paper is that the correct sense of a polysemous term can be found by expanding the network with words occurred in the context of the term. For example, consider following two sentences;

- “Previn and the LSO on the front of any disc was . . . ”
- “Microsoft will replace your disc, if it’s within . . . ”

Figure 3 shows two expanded networks of the term “disc”. Figure 3-(a) is the network of “disc” expanded by “LSO” which is observed in the context of the first sentence. Figure 3-(b) is expanded by term “Microsoft”. The network of “disc” has no prevailing sense, but in the expanded graph, a particular sense is prevailing. The network of “phonograph” will be more similar to that of “disc, LSO” than that of “disc, Microsoft”. On the other hand, the network of “computer” will be more similar to that of “disc, Microsoft”.

Networks for senses can be built in a similar fashion. Figure 4 outlines the idea. In English WordNet, the sense of disc as sound/music recording is defined with a synset: {*disc*, *disk*, *phonograph*, *recording*, *record*}.

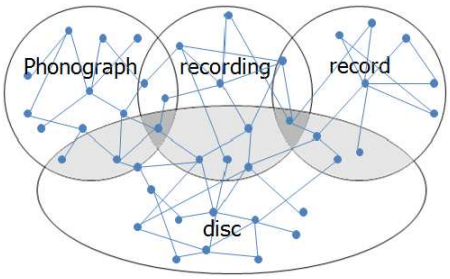


Figure 4: Finding common parts from a synset.

A network which corresponds to the synset can be built by finding shared network sub-structures. In the figure, sub-networks shared between *disc–phonograph*, *disc–recording* and *disc–record* have been unified (the grey area). This unified part is also a network, and it can be used as the network representation for the sense.

By representing senses in this way, senses and the polysemous term with its context are now both represented in the same form of networks. Thus, it is possible to compare them directly with a kernel function for networks.

4. Building Network Representation from Corpus

4.1 Building Co-occurrence Network

Let $T_x = \{x, t_1, \ldots, t_m\}$ be terms observed around term x from the corpus. Instead of converting the observation into a vector of $m+1$ elements, it is possible to build a network with $m+1$ nodes. Let a matrix S_x be an adjacency matrix of a network. The matrix S_x is a square matrix where both columns and rows represent nodes, and its element $[S]_{ij}$ represents the weight of an edge from node i to node j. That is,

$$[S_x]_{ij} = count(t_i, t_j),$$

where $count(t_i, t_j)$ returns the number of times that the terms t_i and t_j are observed together in contexts. All nodes have an edge connected to x, the center node, but they also have other edges. For example, if two terms t_i and t_j have co-occurred in the context of x, there is an edge between node t_i and node t_j. The network expressed by matrix S_x is then regarded as a unit representation for the term x. The size of the context (context window) is a parameter of the network building.

A unit network that represents a single term can be extended to express a set like a synset or a set of context terms. To do so, two additional operations are needed. One is a union operation for networks, and the other is an intersection operation.

Let A and B be adjacency matrices. $U_{AB} = terms(A) \cup terms(B)$ is a set of terms that appear either in A or B, where a function $terms(X)$ returns all terms in the adjacency matrix X. Similarly, $I_{AB} = terms(A) \cap terms(B)$ is a set of terms that appears both in A and B. Let A' be an adjacency matrix expanded from A with U_{AB}. In A', the rows and columns which do not appear in A but B are filled with 0. Then, the union of two networks A and B is defined as simple sum of two expanded matrices.

$$S_A \cup S_B = A' + B'.$$

Similarly, the intersection of two networks is defined as

$$S_A \cap S_B = A'' + B'',$$

where A'' is an adjacency matrix reduced from A with I_{AB}. For A'', the rows and columns which do not appear in both A and B are removed from A.

4.2 Network Operations for Context and Synset

A unit network of a term only holds terms that co-occurred directly with the term. Thus, every node in a unit network is just one walk away from the center term (distance-1 network). Network union can be used to expand a unit network with its co-occurring terms to reflect higher order co-occurrences. For example, in the network of term “data”, top five nodes in terms of node degree are: {“computer”, “available”, “system”, “information”, “collection”}. A second-order network (distance-2 network) with expansion parameter α (number of expanding nodes, which is 5 in this case) can be built by using union operations: $S_{computer} \cup S_{available} \cup S_{system} \cup S_{information} \cup S_{collection}$.

For tasks related to resolving polysemous words, it is important to find an appropriate sub-network. Network intersection operation can be applied to find a common-substructure that is activated by context terms. For example, if the term “disc” appears in the sentence like “A 550 watt motor provides the power behind the 11,000rpm disc.”, the goal is to find sub-networks of “disc” that is activated by terms “motor”, “power” and “watt”. To extract the activated sub-network, network intersections and unions should be applied as $(S_{disc} \cap S_{motor}) \cup (S_{disc} \cap S_{power}) \cup (S_{disc} \cap S_{watt})$. In effect, this operation returns the shared network that is spanning between the polysemous term and its context terms. Networks corresponding to senses are also built similarly from synsets.

5. Graph Kernel as Network Similarity Measure

Haussler first defined a principle way of making kernels on structured objects, so called R-convolution kernel [14]. Graph kernels are R-convolution kernels on graphs, and define the similarity between graphs by their common sub-structures. The most commonly used kernel for networks is the random-walk graph kernel which uses common random-walks between two graphs as sub-structures. Random walk kernel counts the number of matched random walks where matches between nodes and edges are determined by comparing their labeled values.

The kernel value of two graphs G_1 and G_2 is the sum of the sub-kernels for all pairs of walks within these two graphs.

$$k_{graph}(G_1, G_2) = \sum_{walk_1 \in G_1} \sum_{walk_2 \in G_2} k_{walk}(walk_1, walk_2), \quad (1)$$

where k_{walk} is a sub-kernel to compute the similarity between two random walks. k_{walk} is also a product of all similarities between nodes and edges along the walks. Let $k_{node}(v, w)$ return the similarity between two nodes (terms) and an edge kernel k_{edge} return the similarity between two co-occurrence counts. Then, k_{walk} is written as

$$k_{walk}(walk_1, walk_2) = \prod_{i=1}^{n-1} \lambda k_{step}((v_i, v_{i+1}), (w_i, w_{i+1})), \quad (2)$$

where

$$k_{step}((v_i, v_{i+1}), (w_i, w_{i+1})) = k_{node}(v_i, w_i) \cdot k_{node}(v_{i+1}, w_{i+1}) \cdot k_{edge}((v_i, v_{i+1}), (w_i, w_{i+1}))$$

and λ is a decay parameter that is used to guarantee the convergence.

In this paper, node labels (terms) are compared by a delta kernel. That is,

$$k_{node}(x, x') = \begin{cases} 1 & \text{if } label(x) = label(x'), \\ 0 & \text{otherwise} \end{cases}$$

where $label(x)$ is the label of the node x. Edge values are compared via a Brownian bridge kernel. That is,

$$k_{edge}(y, y') = \max(0, 1 - |wgt(y) - wgt(y')|),$$

where $wgt(y)$ returns the labeled value of the edge y. The function returns normalized value of edge counts by Jaccard-coefficients.

Gartner et al. [15] proposed an elegant solution for calculating random walk graph kernel based on matrix operations. Random walks generally include cycles that visits same nodes more than once. Noh et al. [13] proposed an acyclic random walk kernel calculation for special cases where all node labels are unique. The co-occurrence network satisfies this property, and the kernel calculation is adopted in this paper.

Table 1: Experimental results of synonym test

Wnd. Size	Network		Vector	
	N,V,ADJ	N,ADJ	N,V,ADJ	N,ADJ
4	57.50%	65.00%	55.00%	62.50%
6	55.00%	68.75%	50.00%	63.75%
8	50.00%	63.75%	45.00%	58.75%
Sent.	46.25%	65.00%	45.00%	45.00%

6. Experiments

6.1 Setup

Co-occurrence data for experiments have been collected from BNC-XML corpus. To build a unit network for each term, all sentences that have the term are collected from the corpus. Each sentence is then part-of-speech (POS) tagged with a POS tagger [16], and stop words are removed from the resulting tagged sequence. Finally the network building process described in Section 4.1 is applied to preparing unit networks.

6.2 Synonym Test

Synonym test is a task of selecting synonyms from candidates without any context. For example, a question word *grin* is given with a set of candidates, {exercise, rest, joke, smile}. The task is to select one synonym from the candidates. TOEFL synonym test set of [4] is used for the experiment.

In the proposed representation, the synonym task is to select the most similar network. Let G_t be a network that represent the question term t and G_{c_i} be that of a synonym candidate c_i where $c_i \in C$, the set of candidates. Then, the most probable synonym c^* is determined by

$$c^* = \arg\max_{c_i \in C} \frac{k_{graph}(G_t, G_{c_i})}{\sqrt{k_{graph}(G_t, G_t) k_{graph}(G_{c_i}, G_{c_i})}}.$$

The synonym test was done in two steps. First the synonym task was performed only with unit networks. Then, the expanded networks are used to represent both question and candidate terms to see the difference.

Table 1 shows the result of the first step with baseline method of vectors with cosine similarity. The rows of the table show the window size for the context. Window size 4 means that only 4 words in the left and right to the center term are treated as context. The columns of the table show

Table 2: Synonym test with distance-2 networks.

α	Exp. all qst.	Exp. noun qst. only	Exp. noun & adj. qst.
2	56.25%	67.50%	72.50%
3	55.00%	70.00%	73.75%
4	52.50%	68.70%	73.75%

POS classes used for building networks and vectors. Values in the table are accuracy in percentage.

Two POS space is used in the experiment: A space of nouns (N), adjectives (A) and verbs (V), and a space of nouns and adjectives. The limited POS space of nouns and adjectives performs better for both vectors and networks. Another point to note is the effect of window size. For vectors, reduced window size significantly improves the result of synonym test, and this concurs with [17].

However, for networks, the effect of window size is not that significant. The network and its kernel consistently outperform vector and its cosine similarity under the same conditions. The best performance of the network was achieved with networks of nouns and adjective and window size 6. Networks of all experiments below have been built with this setup.

The second step of synonym test is designed to test the effect of higher-order co-occurrences. Both the question word and their candidate words are expanded to distance-2 networks as described in Section 4.2. The resulting networks now have second order co-occurrences.

Table 2 shows the accuracy of synonym finding according to α, the number of nodes to be expanded. The test set has nouns, adjectives, verbs and adverbs as the question words. Expanding all questions does not yield good results. When expanding all questions with $\alpha = 4$, the accuracy was dropped from 68.75% to 56.25%. Some previous work [5] suggests that including higher-order co-occurrences usually improves accuracy from 10% to 15%. However, for the proposed representation with test setup, accuracy of verb and adverb questions actually degrades with distance-2 network. On the contrary, noun questions and adjective questions yield improved results with higher-order co-occurrences. The best result for synonym test is 73.75% with $\alpha = 3$ where noun and adjective questions are compared with distance-2 networks and all other questions were compared with unit networks.

Original LSA paper [4] achieved 64% of accuracy in the test with a smaller corpus. ACL wiki lists results of various systems with the same test set. The best score known for the test set is as high as 97.5% [18] and many achieved better than the proposed representation's 73.75%. However, the systems with high accuracy are mostly supervised systems that need either a manually tagged corpus or a language resource like WordNet.

6.3 Word Sense Disambiguation

For Word Sense Disambiguation, Senseval-3 English lexical sample (S3LS) test set is used as a test set. Although there are more recent test sets for WSD, Senseval-3 is the most widely tested and reported. Senseval-3 English lexical sample test set has 3944 tests for 57 polysemous terms including nouns, adjectives and verbs. For this experiment, only noun disambiguation tests have been selected and tested. There are twenty nouns with total 1807 tests.

To resolve the sense of a polysemous term, both the context of the polysemous term and the candidate senses should be expressed as networks. The network for contexts are built by collecting five nouns and adjectives from the context of the polysemous word. Then, the unit network of the polysemous term and context terms are combined by intersections and unions as described in Section 4.2. Networks corresponding to senses are also built similarly from terms in the synsets.

One problem of making networks from synsets is the existence of empty synsets. Senseval-3 uses WordNet senses as its sense inventory, and many senses have empty synsets and only have glosses. For example, the sense for "paper" as "research paper" has an empty synset and a gloss: "a scholarly article describing the results of observations or stating hypotheses; 'he has written many scientific papers' ".

For such a sense, a pseudo-synset has been manually prepared to have at least four nouns and adjectives from the gloss or siblings. For example, a set for the sense is manually filled with {*paper, scholarly, article, hypotheses, scientific*}. This is the only manual intervention in the experiments. No training data or sense frequencies has been used.

After the senses and contexts have been prepared, their similarity is compared to select the most likely sense of the term in the context. The selection is done in the same fashion with the synonym test experiment. The only difference is that now the context network is used as the question, and the networks of senses are now treated as the candidates. Thus, the polysemous word with its context is classified into the most similar sense in terms of its network similarity.

Table 3 shows the result against a baseline method and other systems reported with the same data (S3LS nouns) in the literature. Values in the table is recall, but it is accuracy at the same time, since the coverage is 100% in all cases. "Proposed" shows the sense disambiguation result of the proposed representation. "MFS" is a baseline method that always selects the most frequent sense. "Vector-based" is another baseline method that uses the same setup and corpus with the proposed method, but it uses the vector representation and the cosine similarity measure. "S3LS best supervised" is the highest scoring system in the S3LS task [19]. Its performance is recall value filtered for nouns as

Table 3: Sense disambiguation result on nouns of Senseval-3 lexical samples.

Methods	Recall
Proposed	61.9
Vector-based	57.6
MFS	55.0
S3LS best supervised	72.9
kNN-all	70.6
kNN-BoW	63.5
HyperLex-m	59.9
HyperLex-m-opt.	64.6
Cymfony	57.9
Prob0	54.2

reported in [20]. "kNN-all" and "kNN-BoW" are the results of a supervised WSD method [21]. "kNN-all" uses all features including local and topical features while "kNN-BoW" uses only the lexical features.

HyperLex is an unsupervised WSI method based on co-occurrence network [9]. Its output is mapped to S3LS senses by S3LS training data [20]. "HyperLex-m"' implies the performance with default parameters, while "HyperLex-m-opt" is its performance with the parameters optimized by S2LS (Senseval-2 LS) data. "Cymfony" is an unsupervised WSD method that uses a Maximum Entropy model for clustering contexts [22]. It uses some percentage (10%) of S3LS training data to map clusters to senses. "Prob0" is another unsupervised WSD module based on POS and frequency information [23].

The proposed approach outperforms baseline method of MFS more than 6 points. Also, it outperforms an equivalent vector based method than 4 points, and marginally outperforms "HyperLex-m". Among listed unsupervised methods, only the optimized HyperLex outperforms the proposed approach. Network-based WSI methods are generally sensitive to various parameters. For example, the performance of HyperLex on S3LS varies nearly 5 points between default version and optimized version. In the above case, "Hyperlex-m-opt" needed S2LS test data and its answers for the optimization. On the contrary, the proposed approach performed without any training data, sense frequency nor mapping data. The result shows that the proposed approach is comparable to the state-of-the-art unsupervised sense disambiguation methods.

7. Conclusions

A semantic representation and an associated similarity measure have been proposed in this paper. In the proposed approach, term co-occurrences observed around a certain term are recorded as a network, and it is regarded as a semantic representation for the term. A graph kernel is adopted to calculate similarity between two networks.

The validity of the representation and its associated similarity measure was tested with two experiments: synonym test and unsupervised sense disambiguation. The proposed representation generally outperforms a vector representation in the synonym test. In the second experiment, it was shown that the proposed representation and its similarity measure can resolve senses comparable to state-of-the-art unsupervised methods.

The proposed representation is similar to network-based WSI methods in that both uses co-occurrence networks to resolve senses. However, the proposed representation is more flexible since it uses a network itself as a representation and computes similarity between two networks directly in a high dimensional space. Thus, it can be directly used in other tasks like calculation of semantic relatedness.

The proposed representation can be used as a replacement for vectors, since the representation is a spatial representation and its associated kernel is an inner-product function. In this work, two real-world unsupervised setups have been used to show that the network representation outperforms the equivalent vector method representation. However, the general usefulness of the proposed representation in other setups like a supervised setup is yet to be explored, and is a major future work.

Acknowledgements

This work was supported in part by the Industrial Strategic Technology Development Program (10035348, Development of a Cognitive Planning and Learning Model for Mobile Platforms) funded by the Ministry of Knowledge Economy (MKE, Korea).

References

[1] T. Griffiths, M. Steyvers, and J. Tenenbaum, "Topics in semantic representation," *Psychological Review*, vol. 114, no. 2, pp. 211–244, 2007.

[2] D. Blei, A. Ng, and M. Jordan, "Latent dirichlet allocation," *The Journal of Machine Learning Research*, vol. 3, pp. 993–1022, 2003.

[3] M. Rosen-Zvi, T. Griffiths, M. Steyvers, and P. Smyth, "The author-topic model for authors and documents," in *Proceedings of the 20th Conference on Uncertainty in Artificial Intelligence*, pp. 487–494, 2004.

[4] T. Landauer and S. Dumais, "A solution to plato's problem: The latent semantic analysis theory of the acquisition, induction, and representation of knowledge," *Psychological Review*, vol. 104, no. 2, pp. 211–240, 1997.

[5] G. Denhière and B. Lemaire, "Effects of high-order co-occurrences on word semantic similarity," *Current Psychology Letters: Behaviour, Brain & Cognition*, vol. 1, no. 18, 2006.

[6] T. Hughes and D. Ramage, "Lexical semantic relatedness with random graph walks," in *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 581–589, 2007.

[7] E. Gabrilovich and S. Markovitch, "Computing semantic relatedness using wikipedia-based explicit semantic analysis," in *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pp. 6–12, 2007.

[8] J. Boyd-Graber, D. Blei, and X. Zhu, "A topic model for word sense disambiguation," in *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 1024–1033, 2007.

[9] J. Veronis, "Hyperlex: lexical cartography for information retrieval," *Computer Speech & Language*, vol. 18, no. 3, pp. 223–252, 2004.

[10] I. Klapaftis and S. Manandhar, "Word sense induction using graphs of collocations," in *Proceedings of the 18th European Conference on Artificial Intelligence*, pp. 298–302, 2008.

[11] W. Duan, M. Song, and A. Yates, "Fast max-margin clustering for unsupervised word sense disambiguation in biomedical texts," *BMC Bioinformatics*, vol. 10, no. 3, 2009.

[12] 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 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL '09, pp. 19–27, 2009.

[13] T.-G. Noh, S.-B. Park, H.-G. Yoon, S.-J. Lee, and S.-Y. Park, "An automatic translation of tags for multimedia contents using folksonomy networks," in *Proceedings of the 32nd ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 492–499, 2009.

[14] D. Haussler, "Convolution kernels on discrete structures, UCSC-CRL-99-10," tech. rep., University of Santa Cruz, 1999.

[15] T. Gärtner, P. Flach, and S. Wrobel, "On graph kernels: Hardness results and efficient alternatives," in *Proceedings of the 16th Computational Learning Theory and Kernel Machines*, pp. 129–143, 2003.

[16] Y. Tsuruoka, "Bidirectional inference with the easiest-first strategy for tagging sequence data," in *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pp. 467–474, 2005.

[17] K. Rothenhäusler and H. Schütze, "Part of speech filtered word spaces," in *Proceedings of the Workshop on Contextual Information in Semantic Space Models*, pp. 25–33, 2007.

[18] J. Bigham, M. Littman, V. Shnayder, and P. Turney, "Combining independent modules to solve multiple-choice synonym and analogy problems," in *Proceedings of the International Conference on Recent Advances in Natural Language Processing*, pp. 482–489, 2003.

[19] R. Mihalcea, T. Chklovski, and A. Kilgarriff, "The Senseval-3 english lexical sample task," in *Proceedings of the 3rd International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pp. 25–28, 2004.

[20] E. Agirre, D. Martínez, O. Lacalle, and A. Soroa, "Two graph-based algorithms for state-of-the-art wsd," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 585–593, 2006.

[21] E. Agirre, O. de Lacalle, and D. Martnez, "Exploring feature spaces with SVD and unlabeled data for word sense disambiguation," in *Proceedings of the Conference on Recent Advances on Natural Language Processing*, 2005.

[22] C. Niu, W. Li, R. Srihari, and H. Li, "Word independent context pair classification model for word sense disambiguation," in *Proceedings of the 9th Conference on Computational Natural Language Learning*, pp. 33–39, 2005.

[23] J. Preiss, "Probabilistic WSD in Senseval-3," in *Proceedings of the 3rd International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pp. 213–216, 2004.

Tae-Gil Noh
Research Associate at Kyungpook National University.
Research Area: Natural Language Processing, Information Retrieval.
E-mail : tgnoh@sejong.knu.ac.kr

Seong-Bae Park

Professor of Kyungpook National University.
Research Area: Machine Learning, Natural Language Processing.
E-mail : seongbae@knu.ac.kr

Sang-Jo Lee
Professor of Kyungpook National University.
Research Area: Natural Language Processing.
E-mail : sjlee@knu.ac.kr