Texts Semantic Similarity Detection Based Graph Approach

Majid Mohebbi and Alireza Talebpour
Department of Computer Engineering, Shahid Beheshti University, Iran

Abstract: *Similarity of text documents is important to analyze and extract useful information from text documents and generation of the appropriate data. Several cases of lexical matching techniques offered to determine the similarity between documents that have been successful to a certain limit and these methods are failing to find the semantic similarity between two texts. Therefore, the semantic similarity approaches were suggested, such as corpus-based methods and knowledge based methods e.g., WordNet based methods. This paper, offers a new method for Paraphrase Identification (PI) in order to, measuring the semantic similarity of texts using an idea of a graph. We intend to contribute to the order of the words in sentence. We offer a graph based algorithm with specific implementation for similarity identification that makes extensive use of word similarity information extracted from WordNet. Experiments performed on the Microsoft research paraphrase corpus and we show our approach achieves appropriate performance.*

Keywords: *WordNet, semantic similarity, similarity metric, graph theory.*

Received November 17, 2013; accepted June 23, 2014; published online April 1, 2015

1. Introduction

Natural Language Processing (NLP) is the use of machinery approaches for analysis, understanding and generating human languages. Two main branches of NLP are Natural Language Analysis (NLA) and Natural Language Generation (NLG). Lexical, syntactic, semantic, pragmatic and morphological analysis of text is studied in NLA. Generation of eloquent multi-sentential or multi-paragraph responses are studied in NLG [6].

Two approaches in semantic similarity problem are paraphrase and bidirectional entailment. A paraphrase is a restatement of the meaning of a passage using other words. In NLG, paraphrase is an approach to increase the variety of generated text [19]. Paraphrases take place at the word level, phrase level, sentence level or discourse level. Paraphrasing has at least three categories, Paraphrase Generation (PG), paraphrase acquisition and Paraphrase Identification (PI). PG is enumerated as a NLG problem is the task of generating alternative paraphrase text [25]. Paraphrase acquisition or paraphrase extraction involves nominee paraphrases or extracting paraphrases from a large corpus [1]. PI or Paraphrase Recognition (PR) or Paraphrase Detection (PD) is the task of recognizing paraphrase relationships at input texts. Textual entailment is the task of identifying, given two text fragments, whether the meaning of one text is entailed (can be inferred) from another text [2]. A paraphrase can be considered as a bidirectional entailment relation namely text A is a paraphrase of text B if and only if A entails B and B entails A [19].

There are two main branches of PI, unsupervised and supervised learning. Unsupervised learning refers to the problem of trying to find hidden structure in unlabeled data. Supervised learning is the machine learning task of inferring a function from labelled training data [23].

For semantic similarity problem, in this article, we focus on sentential paraphrases by an unsupervised approach. The following is an introduction to the problem of similarity of texts.

The similarity between two candidate texts has typically been measured by using a simple lexical matching approach and producing a similarity score based on the number of lexical units that take place in both input segments. Stemming, stop-word removal, part-of-speech tagging, longest subsequence matching, as well as various weighting and normalization factors have been considered for improvement to this simple method [4, 20]. These methods although, successful to a particular degree, will fail to recognize the similarity between sentences which use different, but synonymous, words to carry the same meaning. For text semantic similarity, perhaps the most widely used approaches are the Latent Semantic Analysis (LSA) method [8]. However, due to the complexity and computational cost, LSA has not been used in a large scale.

A related work consists of unsupervised methods for PI, such as methods that Mihalcea *et al.* [14] described for PR and Semantic similarity matrix is described by Fernando and Stevenson [5] which made use of WordNet based methods. While these approaches had the potential of high precision on many examples, improper selection of a specific similarity weight was often insurmountable. Ramage *et al.* [17] presented an algorithm for text semantic similarity, coining the name "Random Walks for Text Semantic Similarity" for his work. This paper presents a new method, the graph based approach. This approach uses a specific implementation of graph theory to find the similarity of two text segments, but a key difference is that special word to word similarities are taken into account, not just the maximal similarities or not all similarities between the sentences as in the methods proposed in [5, 14]. We show the performance of our approach evaluating it on a PR task.

The rest of this paper is organized as follows: Section 2 reviews existing similarity measures. In section 3 we offer based on the graph-based measure, a new similarity measure. Experiments and results are described in section 4. Section 5 gives our conclusions.

2. Previous Approaches

Madnani *et al*. [12] re-examined the idea of automatic metrics used for evaluating translation quality for the task of PR. They employed 8 different machine translation metrics for identifying Paraphrases. Zia and Wasif [26] offered approach of PI using semantic heuristic features. In this approach the POS tagger is performed and closed-class words are removed, after pre-processing step, the feature set was defined. Features were extracted for each sentence pair; afterwards machine learning phase was done. Rajkumar and Chitra [16] offered a neural network classifier for recognizing paraphrases. A combination of lexical, syntactic and semantic features has been used to construct feature vector to train a back propagation network. For feature extraction, approaches such as: Modified string edit distance, the Jiang and Conrath [7] measure, skip-grams with skip distance k as 4 and adapted BLEU metric were used. Rus *et al*. [19] offered a graph subsumption approach for PR. The input sentences were mapped to graph structures and subsumption was detected by evaluating graph isomorphism. The entailment score for text A with respect to text B and B with respect to A have been averaged to determine whether A and B are paraphrases.

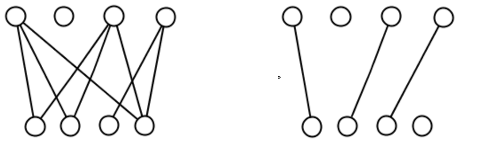
The approach was developed by Mihalcea *et al*. [14] surpassed simple lexical matching. To estimate the semantic similarity of the sentence pairs, Word-to-word similarity measures (such as Jiang *et al*. [7, 9, 11, 18, 24] and Inverse Document Frequency (IDF) of the word as a word specificity measure were used. The main idea of the approach proposed by Fernando and Stevenson [5] was to use the matrix similarity approach to find the similarity of two text segments, but a key difference was that all word to word similarities were taken into account, not just the maximal similarities between the sentences as in the method proposed in Mihalcea *et al*. [14]. The approach was developed by Ramage *et al*. [17] compared the distribution each text induced when used as the seed of a random walk over a graph constructed from WordNet and corpus statistics. Their algorithm aggregated local relatedness information via a random walk over a graph constructed from an underlying lexical resource. The stationary distribution of the graph walk forms a "semantic signature" that can be compared to another such distribution to get a relatedness score for texts [17].

3. Graph based Approach

Number of previous unsupervised works have shown that similarity measures is still limited by the fact that indicates only the most similar or all similar words in the other sentence is taken into account.

We propose a new similarity measure by using the idea of Maximum Matching (MM) of graph theory to better find the similarity between texts. We explore an unsupervised knowledge-based method for measuring the semantic similarity of texts that specific word to word similarities are taken into account, not just the maximal similarities or all similarities between the sentences. In the following, we present our algorithm.

First, we introduce MM algorithm of graph theory. Consider an undirected, unweighted bipartite graph $G=\{X, Y, E\}$, where $X=\{x_1, ..., x_m\}$ and $Y=\{y_1, ..., y_m\}$ are the partitions, $V=X\cup Y$ is the vertex set and $E=e_{ij}$ is the edge set. A matching M of G is a subset of the edges E, such that no vertex in V is incident to more than one edge in M. Intuitively, no two edges in M have a common vertex. A matching M is said to be Maximum if for any other matching $|M|\geq|M'|$. $|M|$ is the maximum sized matching [13]. We see using of MM for given the bipartite graph G Figure 1-a, as demonstrated in Figure 1-b.



a) Bipartite graph G. b) After applying MM algorithm.

Figure 1. Using of MM algorithm.

For a given pair of text segments, we begin by producing sets of open-class words, with a distinct set created for nouns, verbs and adjectives-adverbs-cardinals. Next, we try to determine similarity of pairs of words across the sets corresponding to the same open-class in the two text segments. We enforce the "same word-class" restriction to all the word-to-word similarity measures. For nouns and verbs, we use a measure of semantic similarity based on WordNet, while for the other word classes we use lexical matching. To quantify the degree of semantic relation of two words (nouns and verbs), we use six measures including [7, 9, 10, 11, 18, 24]. We use the WordNet-based implementation of these metrics available in the WordNet::Similarity package [15].

Only the score of Lin *et al*. [11, 24] measure is between 0 and 1. The remaining measures are normalized in a range of 0-1 by dividing the similarity score provided by a given measure with the possible maximum score for that measure.

We execute a part-of-speech tagging on a sentence using Stanford tagger [22]. We construct a bipartite graph $G=\{X, Y, E\}$ that X shows words associated with a one class of first sentence and Y shows words associated with a same class of second sentence and edges E extracted from WordNet 3.0 and an edge is placed between every two congener classes. No edge is placed between two incongruous classes.

Note, in building the graph, the arrangement of the nodes in the specific sets should be in accordance with the appearance of the words in the input texts.

Now, for building the graph, we need to implement an algorithm that uses feature of MM for weighted graph. For each set of built bipartite graph, initially we consider the group with minimum nodes (If the number of nodes in the two parts were equal, the maximum edge of each vertex is obtained. Then, the part is selected that its sum of maximum edge weight of vertices was minimal. If this condition became the same, The selected group has the greatest sum of IDF words), then for the first node of selected set, we choose the first edge with maximum weight, for the second node also we choose the first edge with maximum weight but with respect to property of MM (that no two edges share the same node) and so on.

In the proposed algorithm, we do not implement MM, rather, we use the features of this algorithm in the proposed approach. The features of our approach are affected by the order of appearance of the words and choosing special edge. We are coining the name Extended Maximum Matching (EMM) for this algorithm.

It should be noted that EMM will apply separately to each pair of nouns, verbs, adjectives-adverbs-cardinals. In other words, there would be no edge between incongruous classes, even with zero weight.

To apply the EMM algorithm to calculate the similarity between two sentences, in order to select values of similarity, we also, consider the edges with zero weight across the same class that they will be chosen by EMM. The cause is an impact of the words that don't have any resemblance to the corresponding class of other sentence. These words have increased the length of sentence, In other words, in general, the similarity has been reduced. Now, we present our algorithm with an example.

In the MSR paraphrase corpus [3] the paraphrase pair "408890-408992" is assessed at dissimilar.

The first sentence is: "Acer said its Veriton 7600G incorporates the Intel 865G chipset and is priced starting at $949" and the second sentence is: "The Intel 865G chipset is priced at $44 with integrated software RAID, $41 without RAID".

Figure 2 shows the constructed graph for two candidate sentences by wup measure values of WordNet: Similarity package [15] to determine the similarity of pairs of words across the same segment in the two texts. There is no edge between incongruous classes of two sentences. The implied edges are shown with a gray dash line. The grey edges have zero weight. There will be a chance to choose the implied edges By EMM. Now, edges are selected by EMM. Figure 3 shows the selected edges. Using the weights of selected edges and the number of nodes, the similarity between the two texts is determined by the following scoring function:

$$Sim(T_1, T_2) = \frac{\sum weight\ of\ Selection\ Edges}{\frac{1}{2}(Number\ of\ nods(T_1) + Number\ of\ nods(T_2))} \quad (1)$$

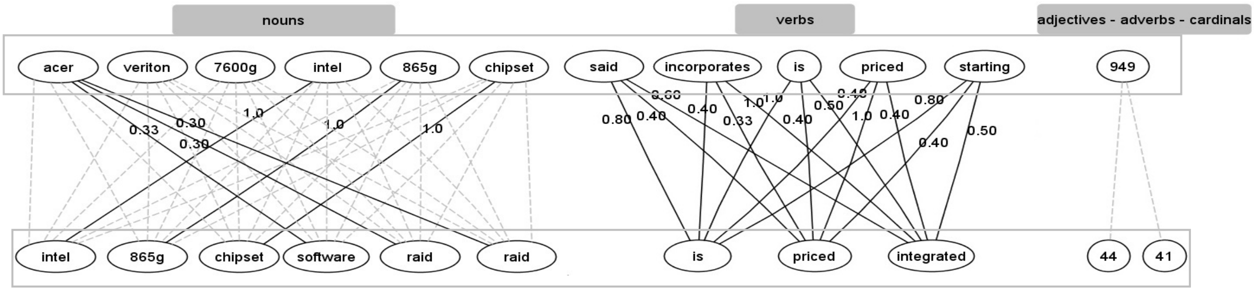


Figure 2. Constructed graph by wup measure values for pairs of words. The first row shows first sentence elements and the second row shows second sentence elements. There is no edge between incongruous classes of two sentences. The gray dash edges have zero weight.

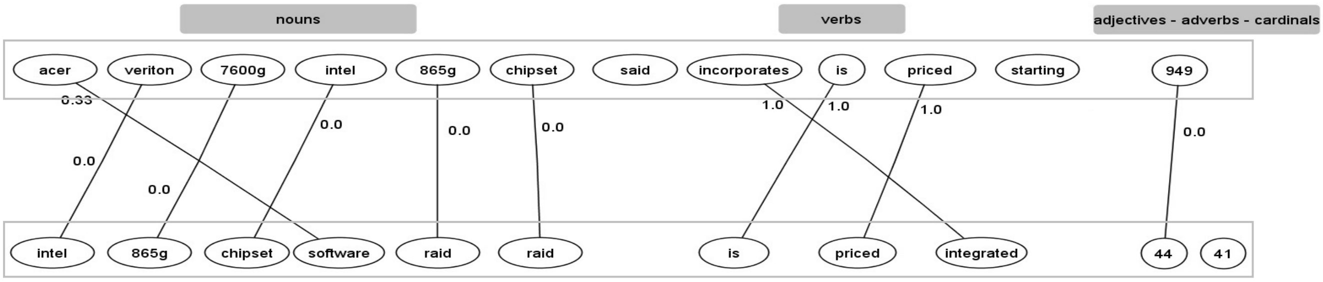


Figure 3. Result of our approach-selected edges by the EMM algorithm.

For example, for two candidate sentences from the dataset that Figure 3 has shown selected edges, by using the metric shown in Equations 1, the similarity between sentences is:

$$Sim(T_1,\ T_2) = \frac{0.30+0+0+0+0+0+1.0+1.0+1.0+0}{\frac{1}{2}(12+11)} = 0.286$$

We use a threshold of 0.59 for classification; a score below the threshold was classified as non-similar sentence otherwise as similar (paraphrase). Therefore, we get a correct diagnosis (not paraphrase).

In the following, we present another version of our algorithm, the second type. We take into account the specificity of words, in a way that we give a higher weight to the similarity measured between two specific words and give less importance to the similarity calculated between generic concepts.

For determining the specificity of a word, we use the IDF [21] defined as the total number of documents in the corpus divided by the total number of documents including that word. We use "BNC database and word frequency lists" by Adam Kilgarriff for document frequency counts for the experiments reported here.

In the second type algorithm, for each edge, we multiply the edge weights by the average IDF of two nodes of an edge, afterwards we run EMM algorithm. We are coining the name "EMM before" for this algorithm. The main feature of this feature of this algorithm is combining the word similarity and their specificity.

The similarity for EMM before is determined using the following scoring function:

$$Sim_{before}(T_1,T_2) = \frac{\sum idfweighted\ of\ Selection\ Edges}{\frac{1}{2}(\sum idf\ of\ nods(T_1) + \sum idf\ of\ nods(T_2))} \qquad (2)$$

Using Equation 2 we get the semantic similarity of the two candidate sentences as 0.226, i.e., correct diagnosis (not paraphrase).

The approach proposed by Mihalcea *et al*. [14] for the nouns 'intel', '865g' and 'chipsetin' of first sentence, find the same similar word from the second sentence. E.g., weighting by this approach for the mentioned sentence pairs, leads to the fact that these two sentences are detected as paraphrase, i.e., not correct diagnosis and also semantic similarity matrix [5] does not provide an adequate performance. In the semantic similarity matrix [5] it was considered all similarity values to complete the similarity matrix and in this approach, selecting additional weights that would affect the accuracy of system, would increases computing time.

4. Evaluation and Results

The Microsoft Research Paraphrase Corpus has been used throughout our experiments. It is the result of an effort to construct a large scale paraphrase corpus for generic purposes [3]. The data have been arbitrarily split into a training set containing 4076 examples and a test set containing 1725 examples.

Our algorithm can be used as unsupervised or supervised. At unsupervised experimental setting, we only use the test data in the experiments and for each pair in the test set, we evaluate our algorithm, and we use a threshold of 0.59.

In our evaluation, we show accuracy, precision, recall and F_measure of our system.

We compare the results of our system with unsupervised algorithms with other unsupervised approaches.

Table 1 shows the results obtained from our algorithms in the unsupervised setting using a threshold of 0.59.

Table 1. Experimental results of our algorithms on MSR paraphrase corpus by using a threshold of 0.59.

	Metric	Acc.	Prec.	Rec.	F
	Semantic Similarity (Knowledge-Based)				
Our Approach (EMM)	J and C	69.57	78.43	74.80	76.57
	L and C	72.00	72.96	91.98	81.37
	Lesk	67.01	78.78	68.96	73.55
	Lin	70.96	75.47	83.44	79.25
	W and P	71.94	72.26	93.81	**81.64**
	Resnik	**72.70**	76.70	84.66	80.48
EMM Before	J and C	61.04	79.28	56.06	65.68
	L and C	67.01	75.71	74.19	74.94
	Lesk	59.25	79.06	52.66	63.21
	Lin	65.45	76.02	70.18	72.98
	W and P	67.71	73.38	80.73	76.88
	Resnik	64.46	77.24	66.00	71.18

As we showed the experiment results of our two approaches in Table 1, it indicates that "EMM" offers better results than "EMM before" approach. The reason is that only open-class words have been evaluated by our algorithm and closed-class words were removed. Because the use of the valence of words, does not have the desired effect. Hence, we compared the results of the EMM approach to the other approaches. For having a fair judgment result, we generate results of Mihalcea *et al.* [14] measure by using WordNet3.0. Hence, we implement Mihalcea *et al.* [14] measure then, evaluate it. By comparing the results in Table 2 and the results reported in Mihalcea *et al.* [14] we observed an increase in the accuracy by applying WordNet3.0. Also in Table 2, the results reported in Mihalcea *et al.* [14] associated with six metrics are shown.

A comparison between the high value of achieved accuracy in the results of our system in Table 1 and Mihalcea *et al.* [14] measure together with corpus-based measure in Table 2, show our approach outperforms these approaches.

Table 3 shows the subset results reported in Ramage *et al.* [17] that was used in version 3.0 of WordNet. We observed our approach outperforms random graph walk approach.

Table 2. Experimental results of Mihlcea *et al.* [14] approach.

	Metric	Acc.	Prec.	Rec.	F
	Semantic Similarity (Knowledge-Based)				
Mihlcea *et al.* [14] Approach by using WordNet3.0	J and C	**70.38**	71.46	92.33	80.56
	L and C	69.10	68.81	97.91	80.82
	Lesk	69.91	**71.72**	90.41	79.98
	Lin	70.20	70.17	95.99	**81.08**
	W and P	69.28	68.80	**98.43**	80.99
	Resnik	69.80	69.81	96.16	80.89
Mihlcea *et al.* [14] Approach	J and C [14]	69.3	72.2	87.1	79.0
	L and C [14]	69.5	72.4	87.0	79.0
	Lesk [14]	69.3	72.4	86.6	78.9
	Lin [14]	69.3	71.6	88.7	79.2
	W and P [14]	69.0	70.2	92.1	80.0
	Resnik [14]	69.0	69.0	96.4	80.4
	Combined [14]	**70.3**	69.6	97.7	81.3
Mihlcea *et al.* [14] Measure	**Semantic Similarity (Corpus-Based)**				
	PMI-IR [14]	69.9	70.2	95.2	81.0
	LSA [14]	68.4	69.7	95.2	80.5
	Baselines				
	Vector-based [14]	65.4	71.6	79.5	75.3
	Random [14]	51.3	68.3	50.0	57.8

Table 3. Experimental results of random graphwalk approach.

	Metric	Acc.	F
Random GraphWalk [17]	Walk (Cosine) [17]	68.7	78.7
	Walk (Dice) [17]	**70.8**	80.1
	Walk (JS) [17]	68.8	**80.5**

5. Conclusions

We offered a new approach using graph theory for computing text semantic similarity and using WordNet as a knowledge base.

In our algorithm, we use the features of the MM algorithm in the proposed approach. By selecting specific edges, only the specific weight of similarity is selected for pair of words. Our proposed algorithm does not attempt to find the max similarity for each word and do not use all similarity values; rather it selects the certain weights (edges), according to previous selections. The features of our approach are affected by the order of appearance of the words and by choosing a special edge. Using our algorithm, we obtained appropriate results.

By using the specificity of words, we present another version of the algorithm, first proposed. Results indicated that the first algorithm outperforms the second and other algorithms.

We evaluated our system on the Microsoft research paraphrase corpus and achieved an appropriate performance.

References

[1] Bhagat R., Hovy E., and Patwardhan S., "Acquiring Paraphrases From Text Corpora," *in Proceedings of the 5th International Conference on Knowledge Capture*, New York, USA, pp. 161-168, 2009.

[2] Dagan I., Glickman O., and Magnini B., "The Pascal Recognising Textual Entailment Challenge," *in Proceedings of the 1st PASCAL Machine Learning Challenges Workshop*, Southampton, UK, pp. 177-190, 2006.

[3] Dolan B., Quirk C., and Brockett C., "Unsupervised Construction of Large Paraphrase Corpora: Exploiting Massively Parallel News Sources," *in Proceedings of the 20th International Conference on Computational Linguistics*, NJ, USA, pp. 350-356, 2004.

[4] Elberrichi Z. and Abidi K., "Arabic Text Categorization: A Comparative Study of Different Representation Modes," *the International Arab Journal of Information Technology*, vol. 9, no. 5, pp. 465-470, 2012.

[5] Fernando S. and Stevenson M., "A Semantic Similarity Approach to Paraphrase Detection," *in Proceedings of the 11th Annual Research Colloquium of the UK Special Interest Group for Computational Linguistics*, Oxford, UK, pp. 45-52, 2008.

[6] Indurkhya N. and Damerau F., *Handbook of Natural Language Processing*, CRC Press, 2010.

[7] Jiang J. and Conrath W., "Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy," *in Proceedings of International Conference Research on Computational Linguistics*, Taiwan, pp. 1-15, 1997.

[8] Landauer K., Foltz W., and Laham D., "An Introduction to Latent Semantic Analysis," *Discourse Processes*, vol. 25, no. 2, pp. 259-284, 1998.

[9] Leacock C. and Chodorow M., "Combining Local Context and Wordnet Sense Similarity for Word Sense Identification," *WordNet: An Electronic Lexical Database*, Publisher: MIT Press, 2013.

[10] Lesk M., "Automatic Sense Disambiguation using Machine Readable Dictionaries: How to Tell a Pine Cone from an Ice Cream Cone," *in Proceedings of the 5th Annual International Conference on Systems Documentation*, New York, USA, pp. 24-26, 1986.

[11] Lin D., "An Information-Theoretic Definition of Similarity," *in Proceedings of the 5th International Conference on Machine Learning*, California, USA, pp. 296-304, 1998.

[12] Madnani N., Tetreault J., and Chodorow M., "Re-examining Machine Translation Metrics for Paraphrase Identification," *in Proceedings of Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Montr´eal, Canada, pp. 182-190, 2012.

[13] Maximum Matching., available at: http://www.cs.dartmouth.edu/~ac/Teach/CS105-Winter05/Notes/kavathekar-scribe.pdf, last visited 2013.

[14] Mihalcea R., Corley C., and Strapparava C., "Corpus-based and Knowledge-based Measures of Text Semantic Similarity," *in Proceedings of*

the American Association for Artificial Intelligence, Boston, USA, pp. 775-780, 2006.

[15] Pedersen T., Patwardhan S., and Michelizzi J., "WordNet::Similarity: Measuring the Relatedness of Concepts," *in Proceedings of the 19th National Conference on Artificial Intelligence*, California, USA, pp. 1024-1025, 2004.

[16] Rajkumar A. and Chitra A., "Paraphrase Recognition using Neural Network Classification," *the International Journal of Computer Application*, vol. 1, no. 29, pp. 43-48, 2010.

[17] Ramage D., Rafferty N., and Manning D., "Random Walks for Text Semantic Similarity," *in Proceedings of Workshop on Graph-based Methods for Natural Language Processing*, Pennsylvania, USA, pp. 23-31, 2009.

[18] Resnik P., "Using Information Content to Evaluate Semantic Similarity in a Taxonomy*," in Proceedings of the 14th International Joint Conference on Artificial Intelligence*, San Francisco, USA pp. 448-453, 2013.

[19] Rus V., McCarthy P., Lintean M., McNamara D., and Graesser A., "Paraphrase Identification with Lexico-Syntactic Graph Subsumption," *in Proceedings of the 21st International Florida Artificial Intelligence Research Society Conference*, Florida, USA, pp. 201-206, 2008.

[20] Salton G. and Buckley C., "Term Weighting Approaches in Automatic Text Retrieval," *Information Processing and Management*, vol. 24, no. 5, pp. 513-523, 1988.

[21] Sparck-Jones K., "A Statistical Interpretation of Term Specificity and its Application in Retrieval," *the Journal of Documentation*, vol. 28, no. 1, pp. 11-21, 1972.

[22] Toutanova K., Klein D., Manning C., and Singer Y., "Feature-Rich Part-of-Speech Tagging with a Cyclic Dependency Network," *in Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, Edmonton, Canada, pp. 252-259, 2003.

[23] Unsupervised Learning., available at: http://en.wikipedia.org/wiki/Unsupervised_learning, last visited 2013.

[24] Wu Z. and Palmer M., "Verb Semantics and Lexical Selection," *in Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics*, New Mexico, USA, pp. 133-138, 1994.

[25] Wubben S., Van den A., and Krahmer E., "Paraphrase Generation as Monolingual Translation: Data and Evaluation," available at: http://ilk.uvt.nl/~swubben/publications/INLG2010.pdf, last visited 2010.

[26] Zia U. and Wasif A., "Paraphrase Identification using Semantic Heuristic Features," *Research Journal of Applied Sciences*, *Engineering and Technology*, vol. 4, no. 22, pp. 4894-4904, 2012.

Majid Mohebbi received the MSc degree in software engineering from Shahid Beheshti University in 2013, Iran. His research interests include semantic similarity and NLP.

Alireza Talebpour received his MSc degree in Artificial Intelligence and PhD degrees in Image Processing from University of Surrey, United Kingdom. His research interests include image processing and pattern recognition, intelligent methods for classification of massive data.