PolyUCOMP: Combining Semantic Vectors with Skip bigrams for Semantic Textual Similarity

Jian Xu **Qin Lu** **Zhengzhong Liu**
The Hong Kong Polytechnic University
Department of Computing
Hung Hom, Kowloon, Hong Kong
{csjxu, csluqin, hector.liu}@comp.polyu.edu.hk

Abstract

This paper presents the work of the Hong Kong Polytechnic University (PolyUCOMP) team which has participated in the Semantic Textual Similarity task of SemEval-2012. The PolyUCOMP system combines semantic vectors with skip bigrams to determine sentence similarity. The semantic vector is used to compute similarities between sentence pairs using the lexical database WordNet and the Wikipedia corpus. The use of skip bigram is to introduce the order of words in measuring sentence similarity.

1 Introduction

Sentence similarity computation plays an important role in text summarization, classification, question answering and social network applications (Lin and Pantel, 2001; Erkan and Radev, 2004; Ko et al., 2004; Ou et al., 2011). The SemEval 2012 competition includes a task targeted at Semantic Textual Similarity (STS) between sentence pairs (Eneko et al., 2012). Given a set of sentence pairs, participants are required to assign to each sentence pair a similarity score.

Because a sentence has only a limited amount of content words, it is not easy to determine sentence similarities because of the sparseness issue. Hatzivassiloglou et al. (1999) proposed to use linguistic features as indicators of text similarity to address the problem of sparse representation of sentences. Mihalcea et al. (2006) measured sentence similarity using component words in sentences. Li et al. (2006) proposed to incorporate the semantic vector and word order to calculate sentence similarity.

In our approach to the STS task, semantic vector is used and the semantic relatedness between words is derived from two sources: WordNet and Wikipedia. Because WordNet is limited in its coverage, Wikipedia is used as a candidate for determining word similarity.

Word order, however, is not considered in semantic vector. As semantic information are coded in sentences according to its order of writing, and in our systems, content words may not be adjacent to each other, we proposed to use skip bigrams to represent the structure of sentences. Skip bigrams, generally speaking, are pairs of words in a sentence order with arbitrary gap (Lin and Och, 2004a). Different from the previous skip bigram statistics which compare sentence similarities through overlapping skip bigrams (Lin and Och, 2004a), the skip bigrams we used are weighted by a decaying factor of the skipping gap in a sentence, giving higher scores to closer occurrences of skip bigrams. It is reasonable to assume that similar sentences should have more overlapping skip bigrams, and the gaps in their shared skip bigrams should also be similar.

The rest of this paper is organized as followed. Section 2 describes sentence similarity using semantic vectors and the order-sensitive skip bigrams. Section 3 gives the performance evaluation. Section 4 is the conclusion.

2 Similarity between Sentences

Words are used to represent a sentence in the vector space model. Semantic vectors are constructed for sentence representations with each entry corresponding to a word. Since the semantic vector does not consider word order, we further proposed to use skip bigrams to represent sentence structure. Moreover, these skip bigrams are

*First Joint Conference on Lexical and Computational Semantics (*SEM)*, pages 524–528,
Montréal, Canada, June 7-8, 2012. ©2012 Association for Computational Linguistics

weighted by a decaying factor based on the so called skip distance in the sentence.

2.1 Sentence similarity using Semantic Vector

Given a sentence pair, S_1 and S_2, for example,

S_1: Chairman Michael Powell and FCC colleagues at the Wednesday hearing.

S_2: FCC chief Michael Powell presides over hearing Monday.

The term set of the vector space is first formed by taking only the content words in both sentences,

T={chairman, chief, colleagues, fcc, hearing, michael, monday, powell, presides, wednesday }

Each entry of the semantic vector corresponds to a word in the joint word set (Li et al., 2006). Then, the vector for each sentence is formed in two steps: For a word both in the term set T and in the sentence, the value for this word entry is set to 1. If a word is not in the sentence, the most similar word in the sentence will then be identified, and the corresponding *path* similarity value will be assigned to this entry. Let T be the term set with a sorted list of content words, $T=(t_1, t_2,\ldots, t_n)$. Without loss of generality, let a sentence $S=(w_1\ w_2 \ldots w_m)$ where w_j is a content word and w_j is a word in T. Let the vector space of the sentence S be $VS_s = (v_1, v_2, \ldots, v_n)$. Then the value of v_i is assigned as follows,

$$v_i = \begin{cases} 1 & if\ t_i \in S \\ \underset{w_j \in S}{\arg\max}(SIM(t_i, w_j)) & if\ t_i \notin S \end{cases}$$

where the similarity function $SIM(t_i, w_j)$ is calculated according to the *path* measure (Pedersen et al., 2004) using the WordNet, formally defined as,

$$SIM(t_i, w_j) = 1 / dist(t_i, w_j)$$

where $dist(t_i, w_j)$ is the shortest path from t_i, to w_j by counting nodes in the WordNet taxonomy. Based on this, the semantic vectors for the two example sentences will be,

$SV_{S1} = (1, 0.25, 1, 1, 1, 1, 0.33, 1, 0, 1)$ and

$SV_{S2} = (0.25, 1, 0, 1, 1, 1, 1, 1, 1, 0.33)$

Based on the two semantic vectors, the cosine metric is used to measure sentence similarity. In the WordNet, the entry *chairman* in the joint set is most similar to the word *chief* in sentence S_2. In practice, however, this entry might be closer to the word *presides* than to the word *chief*. Therefore, we try to obtain the semantic relatedness using the Wikipedia for sentence T and find that the entry *chairman* is closest to the word *presides*. The Wikipedia-based word relatedness utilizes the hyperlink structure (Milne & Witten, 2008). It first identifies the candidate articles, a and b, that discuss t_i and w_j respectively in this case and then compute relatedness between these articles,

$$rel(a,b) = \frac{\log(\max(|A|,|B|)) - \log(A \cap B)}{\log(|W|) - \log(\min(|A|,|B|))}$$

where A and B are sets of articles that link to a and b. W is the set of all articles in the Wikipedia. Finally, two articles that represent t_i and w_j are selected and their relatedness score is assigned to $SIM(t_i, w_j)$.

2.2 Sentence Similarity by Skip bigrams

Skip bigrams are pairs of words in a sentence order with arbitrary gaps. They contain the order-sensitive information between two words. The skip bigrams of a sentence are extracted as features which will be stacked in a vector space. Each skip bigram is weighted by a decaying factor with its skip distances in the sentence. To illustrate this, consider the following sentences S and T:

$S = w_1\, w_2\, w_1\, w_3\, w_4$ and $T = w_2\, w_1\, w_4\, w_5\, w_4$

where w denotes a word. It can be used more than once in a sentence. Each sentence above has a $C(5, 2)$[1] $= 10$ skip bigrams.

The sentence S has the following skip bigrams:

"w_1w_2", "w_1w_1", "w_1w_3", "w_1w_4", "w_2w_1", "w_2w_3" , "w_2w_4" , "w_1w_3", "w_1w_4", "w_3w_4"

The sentence T has the following skip bigrams:

"w_2w_1", "w_2w_4", "w_2w_5", "w_2w_4", "w_1w_4", "w_1w_5" , "w_1w_4" , "w_4w_5", "w_4w_4", "w_5w_4"

In the sentence S, we have two repeated skip bigrams "w_1w_4" and "w_1w_3". In the sentence T, we have "w_2w_4" and "w_1w_4" repeated twice. In this case, the weight of the recurring skip bigrams will be increased. Hereafter, vectors for S and T will be

[1] Combination: C(5,2)=5!/(2!*3!)=10.

formulated with each entry corresponding to a distinctive skip bigram.

V_S = (“w_1w_2”, “w_1w_1”, “w_1w_3”, “w_1w_4”, “w_2w_1”, “w_2w_3”, “w_2w_4”, “w_3w_4”)’

V_T = (“w_2w_1”, “w_2w_4”, “w_2w_5”, “w_1w_4”, “w_1w_5”, “w_4w_5”, “w_4w_4”, “w_5w_4”)’

Now, the question remains how to weight the skip bigrams. Given Σ as a finite word set, let $S=w_1w_2...w_{|S|}$ be a sentence, $w_i \in \Sigma$and $1 \leqslant i \leqslant |S|$. A skip bigram of S, denoted by u, is defined by an index set $I=(i_1, i_2)$ of S $(1 \leqslant i_1 < i_2 \leqslant |S|$ and $u=S[I])$. The skip distance of $S[I]$, denoted by d_u (I), is the skip distance of the first word and the second word of u, calculated by i_2-i_1+1. For example, if S is the sentence of $w_1w_2w_1w_3w_4$ and u = w_1w_4, then there are two index sets, I_1=$[3,5]$ and I_2=$[1,5]$ such that u=$S[3,5]$ and u=$S[1,5]$, and the skip distances of $S[3,5]$ and $S[1,5]$ are 3 and 5. The weight of a skip bigram u for a sentence S with all its possible occurrences, denoted by $\phi_u(S)$, is defined as:

$$\phi_u(S) = \sum_{I:u=S[I]} \lambda^{d_u(I)}$$

where λ is the decay factor which penalizes the longer skip distance of a skip bigram. By doing so, for the sentence S, the complete word set is $\Sigma=\{w_1, w_2, w_3, w_4\}$. The weights for the skip bigrams are listed in Table 1:

u	$\phi_u(S)$	u	$\phi_u(S)$
w_1w_2	λ^2	w_2w_1	λ^2
w_1w_1	λ^3	w_2w_3	λ^3
w_1w_3	$\lambda^4+\lambda^2$	w_2w_4	λ^4
w_1w_4	$\lambda^5+\lambda^3$	w_3w_4	λ^2

Table 1: Skip bigrams and their Weights in *S*

In Table 1, if λ is set to 0.25, the weight of the skip bigram w_1w_2 in S is 0.25^2=0.0625, and w_1w_3 is 0.25^4 +0.25^2=0.064. Similarly, the skip bigrams and weights in the sentence T can be obtained. With the skip bigram-based vectors, cosine metric is then used to compute similarity between S and T.

3 Experiments

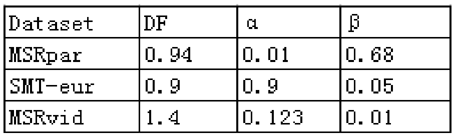
In the STS task, three training datasets are available: MSR-Paraphrase, MSR-Video and SMTeuroparl (Eneko et al., 2012). The number of sentence pairs for three dataset is 750, 750 and 734.

In the following experiments, Let S_{WN}, S_{WIKI} and S_{SKIP} denote similarity measures of the vector space representation using WordNet, Wikipedia and skip bigrams, respectively. The three similarity measures are linearly combined as S_{COMB}:

$$S_{COMB} = \alpha \times S_{WN} + \beta \times S_{WIKI} + (1-\alpha-\beta) \times S_{SKIP}$$

where α and β are weight factors for S_{WN} and S_{WIKI} in the range [0,1]. If α is set to 1, only the WordNet-based similarity measure is used; if α is 0, the Wikipedia and skip bigram measures are used.

Because each dataset has a different representation for sentences, the parameter configurations for them are different. For the word similarity using the lexical resource WordNet, the *path* measure is used in experiments. To get word relatedness from the English Wikipedia, the Wikipedia Miner tool[2] is used. When computing sentence similarity based on the skip bigrams, the decaying factor (DF) must be specified beforehand. Hence, parameter configurations for the three datasets are listed in Table 2:



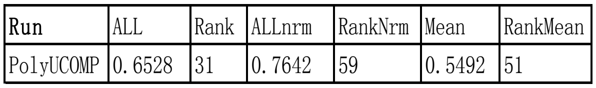
Dataset	DF	α	β
MSRpar	0.94	0.01	0.68
SMT-eur	0.9	0.9	0.05
MSRvid	1.4	0.123	0.01

Table 2: Parameter Configurations

In the testing phase, five testing dataset are provided. In addition to three test datasets drawn from the publicly available datasets used in the training phase, two surprise datasets are given. They are SMTnews and OnWN (Eneko et al., 2012). SMTnews has 399 pairs of sentences and OnWN contains 750 sentence pairs. The parameter configurations for these two surprise datasets are the same as those for the dataset MSR-Paraphrase.

The official scoring is based on Pearson correlation. If the system gives the similarity scores close to the reference answers, the system will attain a high correlation value. Besides, three other evaluation metrics (***ALL, ALLnrm, Mean***) based on the Pearson correlation are used (Eneko et al., 2012).

Among the 89 submitted systems, the results of our system are given in Table 3:



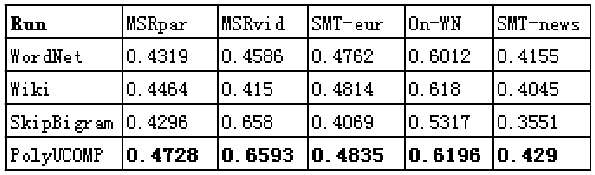
Run	ALL	Rank	ALLnrm	RankNrm	Mean	RankMean
PolyUCOMP	0.6528	31	0.7642	59	0.5492	51

Table 3: Performance using Different Metrics

[2] http://wikipedia-miner.cms.waikato.ac.nz/

Using the *ALL* metric, our system ranks 31, but for *ALLnrm* and *Mean* metrics, our system ranking is decreased to 59 and 51. In terms of *ALL* metric, our system achieves a medium performance, implying that our system correlates well with human assessments. In terms of *ALLnrm* and *Mean* metrics, our system performance degrades a lot, implying that our system is not well correlated with the reference answer when each dataset is normalized into the aggregated dataset using the least square error or the weighted mean across the datasets.

To see how well each of the individual vector space models performed on the evaluation sets, we experiment on the five datasets using vectors based on WordNet, Wikipedia (Wiki), SkipBigram and PolyuCOMP (a combination of the three vectors). Table 4 gives detailed results of each dataset.



Run	MSRpar	MSRvid	SMT-eur	On-WN	SMT-news
WordNet	0.4319	0.4586	0.4762	0.6012	0.4155
Wiki	0.4464	0.415	0.4814	0.618	0.4045
SkipBigram	0.4296	0.658	0.4069	0.5317	0.3551
PolyUCOMP	**0.4728**	**0.6593**	**0.4835**	**0.6196**	**0.429**

Table 4: Pearson Correlation for each Dataset

Table 4 shows that after combining three vector representations, each dataset obtains the best performance. The WordNet-based approach gives a better performance than Wikipedia-based approach in MSRvid dataset. The two approaches, however, give similar performance in other four datasets. This is because the sentences in the MSRvid dataset are too short with limited amount of content words. It is difficult to capture the meaning of a sentence without distinguishing words in consecutive positions. This is why the order-sensitive SkipBigram approach gives better performance than the other two approaches. For example,

A woman is playing a game with a man.
A man is playing piano.

Using the semantic vectors, we will get high similarity scores, but the two sentences are dissimilar. If the skip bigram approach is used, the similarity score between sentences will be 0, which correlates with human judgment. In parameter configurations for the MSRvid dataset, higher weight (1-0.123-0.01=0.867) is also given to skip bigrams. It is interesting to note that the decaying factor for this dataset is **1.4** and is not in the range from 0 to 1 inclusive. This is because higher decaying factor helps to capture semantic meaning between words that span afar. For example,

A man is playing a flute.
A man is playing a bamboo flute.

In this sentence pair, the second sentence is entailed by the first one. The similarity can be captured by assigned larger decay factor to weigh the skip bigram “playing flute” in two sentences. Hence, if the value of the decay factor is greater than 1, the two sentences will become much more similar. After careful investigation, these two sentences are similar to a large extent. In this sense, a higher decaying factor would help capture the meaning between sentence pairs. This is quite different from the other four datasets which focus on shared skip bigrams with smaller decaying factor.

4 Conclusions and Future Work

In the Semantic Textual Similarity task of SemEval-2012, we proposed to combine the semantic vector with the order-sensitive skip bigrams to capture the meaning between sentences. First, a semantic vector is derived from either the WordNet or Wikipedia. The WordNet simulates the common human knowledge about word concepts. However, WordNet is limited in its word coverage. To remedy this, Wikipedia is used to obtain the semantic relatedness between words. Second, the proposed approach also considers the impact of word order in sentence similarity by using skip bigrams. Finally, the overall sentence similarity is defined as a linear combination of the three similarity metrics. However, our system is limited in its approaches. In future work, we would like to apply machine learning approach in determining sentence similarity.

References

David Milne , Ian H. Witten. 2008. An Effective, Low-cost Measure of Semantic Relatedness Obtained from Wikipedia Links. In *Proceedings of the first AAAI Workshop on Wikipedia and Artificial Intelligence (WIKIAI'08),* Chicago, I.L

Dekang Lin and Patrick Pantel. 2001. Discovery of Inference Rules for Question Answering. *Natural Language Engineering*, 7(4):343-360.

Eneko Agirre, Daniel Cer, Mona Diab and Aitor Gonzalez-Agirre. 2012. SemEval-2012 Task 6: A Pilot on Semantic Textual Similarity. In *Proceedings of the 6th International Workshop on Semantic Evaluation (SemEval 2012), in conjunction with the First Joint Conference on Lexical and Computational Semantics (*SEM 2012)*.

Gunes Erkan and Dragomir R. Radev. 2004. Lexrank: Graph-based Lexical Centrality as Salience in Text Summarization. *Journal of Artificial Intelligence Research*, 22: 457–479.

Lin, Chin-Yew and Franz Josef Och. 2004a. Automatic Evaluation of Machine Translation Quality Using Longest Common Subsequence and Skip bigram Statistics. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL 2004)*, Barcelona, Spain.

Ou Jin, Nathan Nan Liu, Yong Yu and Qiang Yang. 2011. Transferring Topical Knowledge from Auxiliary Long Text for Short Text Understanding. In: *Proceedings of the 20th ACM Conference on Information and Knowledge Management (ACM CIKM 2011)*. Glasgow, UK.

Rada Mihalcea and Courtney Corley. 2006. Corpus-based and Knowledge-based Measures of Text Semantic Similarity. In *Proceeding of the Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference*.

Ted Pedersen, Siddharth Patwardhan and Jason Michelizzi. 2004. WordNet::Similarity—Measuring the Relatedness of Concepts. In *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI, San Jose, CA),* pages 144–152.

Vasileios Hatzivassiloglou, Judith L. Klavans , Eleazar Eskin. 1999. Detecting Text Similarity over Short Passages: Exploring Linguistic Feature Combinations via Machine Learning. In *Proceeding of Empirical Methods in natural language processing and Very Large Corpora*.

Youngjoong Ko, Jinwoo Park, and Jungyun Seo. 2004. Improving Text Categorization using the Importance of Sentences. *Information Processingand Management*, 40(1): 65–79.

Yuhua Li, David Mclean, Zuhair B, James D. O'shea and Keeley Crockett. 2006. Sentence Similarity Based on Semantic Nets and Corpus Statistics. *IEEE Transactions on Knowledge and Data Engineering*, 18(8), 1138–1149.