ECNU: Leveraging on Ensemble of Heterogeneous Features and Information Enrichment for Cross Level Semantic Similarity Estimation

Tian Tian Zhu
Department of Computer Science and Technology
East China Normal University
51111201046@ecnu.cn

Man Lan*
Department of Computer Science and Technology
East China Normal University
mlan@cs.ecnu.edu.cn*

Abstract

This paper reports our submissions to the Cross Level Semantic Similarity (CLSS) task in SemEval 2014. We submitted one Random Forest regression system on each cross level text pair, i.e., Paragraph to Sentence (P-S), Sentence to Phrase (S-Ph), Phrase to Word (Ph-W) and Word to Sense (W-Se). For text pairs on P-S level and S-Ph level, we consider them as sentences and extract heterogeneous types of similarity features, i.e., string features, knowledge based features, corpus based features, syntactic features, machine translation based features, multi-level text features, etc. For text pairs on Ph-W level and W-Se level, due to lack of information, most of these features are not applicable or available. To overcome this problem, we propose several information enrichment methods using WordNet synonym and definition. Our systems rank the 2nd out of 18 teams both on Pearson correlation (official rank) and Spearman rank correlation. Specifically, our systems take the second place on P-S level, S-Ph level and Ph-W level and the 4th place on W-Se level in terms of Pearson correlation.

1 Introduction

Semantic similarity is an essential component of many applications in Natural Language Processing (NLP). Previous works often focus on text semantic similarity on the same level, i.e., paragraph to paragraph or sentence to sentence, and many effective text semantic measurements have been proposed (Islam and Inkpen, 2008), (Bär et al., 2012), (Heilman and Madnani, 2012). However, in many real world cases, the two texts may not always be on the same level. The Cross Level Semantic Similarity (CLSS) task in SemEval 2014 provides a universal platform to measure the degree of semantic equivalence between two texts across different levels. For each text pair on four cross levels, i.e., Paragraph to Sentence (P-S), Sentence to Phrase (S-Ph), Phrase to Word (Ph-W) and Word to Sense (W-Se), participants are required to return a similarity score which ranges from 0 (no relation) to 4 (semantic equivalence). We participate in all the four cross levels and take the second place out of all 18 teams both on Pearson correlation (official) and Spearman correlation ranks.

In this work, we present a supervised regression system for each cross level separately. For P-S level and S-Ph level, we regard the paragraph of P-S as a long sentence, and the phrase of S-Ph as a short sentence. Then we use various types of text similarity features including string features, knowledge based features, corpus based features, syntactic features, machine translation based features, multi-level text features and so on, to capture the semantic similarity between two texts. Some of these features are borrowed from our previous system in the Semantic Textual Similarity (STS) task in *SEM Shared Task 2013 (Zhu and Lan, 2013). Others followed the previous work in (Šaric et al., 2012) and (Pilehvar et al., 2013). For Ph-W level and W-Se level, since the text pairs lack contextual information, for example, word or sense alone no longer shares the property of sentence, most features used in P-S level and S-Ph level are not applicable or available. To overcome the problem of insufficient information in word and sense level, we propose several information enrichment methods to extend information with the aid of WordNet (Miller, 1995), which significantly improved the system performance.

The rest of this paper is organized as follows.

Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pages 265–270,
Dublin, Ireland, August 23-24, 2014.

Section 2 describes the similarity features used on four cross levels in detail. Section 3 presents experiments and the results of four cross levels on training data and test data. Conclusions and future work are given in Section 4.

2 Text Similarity Measurements

To estimate the semantic similarity on P-S level and S-Ph level, we treat the text pairs on both levels as traditional semantic similarity computation on sentence level and adopt 7 types of features, i.e., string features, knowledge based features, corpus based features, syntactic features, machine translation based features, multi-level text features and other features. All of them are borrowed from previous work due to their superior performance reported. For Ph-W level and W-Se level, since word and sense alone cannot be treated as sentence, we propose an information enrichment method to extend original text with the help of WordNet. Once the word or sense is enriched with its synonym and its definition description, we can thus adopt the previous features as well.

2.1 Preprocessing

For P-S level and S-Ph level, we perform text preprocessing before we extract semantic similarity features. Firstly, the Stanford parser[1] is used for sentence tokenization and parsing. Specifically, the tokens *n't* and *'m* are replaced with *not* and *am*. Secondly, the Stanford POS Tagger[2] is used for POS tagging. Thirdly, we use Natural Language Toolkit[3] for WordNet based Lemmatization, which lemmatizes the word to its nearest base form that appears in WordNet, for example, was is lemmatized as is rather than be.

2.2 Features on P-S Level and S-Ph Level

We treat all text pairs of P-S level and S-Ph level as sentences and then extract 7 types of similarity features as below. Totally we get 52 similarity features. Generally, these similarity features are represented as numerical values.

String features. Intuitively, if two texts share more strings, they are considered to be more semantic similar. We extract 13 string based features in consideration of the common sequence shared by two texts. We chose the Longest Common Sequence (LCS) feature (Zhu and Lan, 2013), the N-gram Overlap feature (n=1,2,3) and the Weighted Word Overlap feature (Šaric et al., 2012). All these features are computed from original text and from the processed text after lemmatization as well. Besides, we also computed the N-gram Overlap on character level, named Character N-gram (n=2,3,4).

Knowledge based features. Knowledge based similarity estimation relies on the semantic network of words. In this work we used the knowledge based features in our previous work (Zhu and Lan, 2013), which include four word similarity metrics based on WordNet: ***Path*** similarity (Banea et al., 2012), ***WUP*** similarity (Wu and Palmer, 1994), ***LCH*** similarity (Leacock and Chodorow, 1998) and ***Lin*** similarity (Lin, 1998). Then two strategies, i.e., the best alignment strategy and the aggregation strategy, are employed to propagate the word similarity to the text similarity. Totally we get 8 knowledge based features.

Corpus based features. Latent Semantic Analysis (LSA) (Landauer et al., 1997) is a widely used corpus based measure when evaluating text similarity. In this work we use the Vector Space Sentence Similarity proposed by (Šaric et al., 2012), which represents each sentence as a single distributional vector by summing up the LSA vector of each word in the sentence. Two corpora are used to compute the LSA vector of words: New York Times Annotated Corpus (NYT) and Wikipedia. Besides, in consideration of different weights for different words, they also calculated the weighted LSA vector for each word. In addition, we use the Co-occurrence Retrieval Model (CRM) feature from our previous work (Zhu and Lan, 2013) as another corpus-based feature. The CRM is calculated based on a notion of substitutability, that is, the more appropriate it is to substitute word w_1 in place of word w_2 in a suitable natural language task, the more semantically similar they are. At last, 6 corpus based features are extracted.

Syntactic features. Dependency relations of sentences often contain semantic information. In this work we follow two syntactic dependency similarity features presented in our previous work (Zhu and Lan, 2013), i.e., Simple Dependency Overlap and Special Dependency Overlap. The Simple Dependency Overlap measures all dependency relations while the Special Dependency Overlap fea-

[1] http://nlp.stanford.edu/software/lex-parser.shtml
[2] http://nlp.stanford.edu/software/tagger.shtml
[3] http://nltk.org/

ture only focuses on the primary roles extracted from several special dependency relations, i.e., subject, object and predict.

Machine Translation based features. Machine translation (MT) evaluation metrics are designed to assess whether the output of a MT system is semantically equivalent to a set of reference translations. This type of feature has been proved to be effective in our previous work (Zhu and Lan, 2013). As a result, we extend the original 6 lexical level MT metrics to 10 metrics, i.e., *WER*, *TER*, *PER*, *BLEU*, *NIST*, *ROUGE-L*, *GTM-1*,*GTM-2*, *GTM-3* and *METEOR-ex*. All these metrics are calculated using the Asiya Open Toolkit for Automatic Machine Translation (Meta-) Evaluation[4].

Multi-level text Features. (Pilehvar et al., 2013) presented a unified approach to semantic similarity at multiple levels from word senses to text documents through the semantic signature representation of texts (e.g., sense, word or sentence). Given initial nodes (senses), they performed random walks on semantic network like WordNet, then the resulting frequency distribution over all nodes in WordNet served as semantic signature of the text. By doing so the similarity of two texts can be computed as the similarity of two semantic signatures. In this work, we borrowed their semantic signature method and adopted 3 similarity measures to estimate two semantic signatures, i.e., Cosine similarity, Weighted Overlap and Top-k Jaccard (k=250, 500).

Other Features. Besides, other simple surface features from texts, such as numbers, symbols and length of texts, are extracted. Following (Šaric et al., 2012) we adopt relative length difference, relative information content difference, numbers overlap, case match and stocks match.

2.3 Features on Ph-W Level

For Ph-W level, since word and phrase no longer share the property of sentence, most features used for sentence similarity estimation are not applicable for this level. Therefore, we adopt the following features as the basic feature set for Ph-W level.

String features. This type contains two features. The first is a boolean feature which records whether the word appears in the phrase. The second is the Weighted Word Overlap feature mentioned in Section 2.2.

Knowledge based features. As described in Section 2.2, we compute the averaged score and the maximal score between word and phrase using the four word similarity measures based on WordNet, i.e., ***Path***, ***WUP***, ***LCH*** and ***Lin***.

Corpus based features. We adopt the Vector Space Similarity described in Section 2.2. Specifically, for word the single distributional vector is the LSA vector of itself.

Multi-level text Features. As described in Section 2.2, since the semantic signatures are proposed for various kinds of texts (e.g., sense, word or sentence), they serve as one basic feature.

Obviously, the above features extracted from the phrase-word pair is significantly less than the features used in P-S level and S-Ph level. This is because the information contained in phrase-word pair is much less than that in sentences and paragraphs. To overcome this information insufficient problem, we propose an information enrichment method based on WordNet to extend the initial word in Ph-W level as below.

Word Expansion with Definition. For the word part in Ph-W level, we extract its definition in terms of its most common concept in WordNet and then replace the initial word with this definition. This gives a much richer set of initial single word. Since a word may have many senses, not all of this word definition expansion are correct. But we show below empirically that using this expanded set improves performance. By doing so we treat the phrase and the definition of the original word as two sentences, and thus, all features described in Section 2.2 are calculated.

2.4 Features on W-Se Level

For W-Se level, the information that a word and a sense carry is less than other levels. Hence, the basic features that can be extracted from the original word-sense pair are even less than Ph-W level. Therefore the basic features we use for W-Se level are as follows.

String features. Two boolean string features are used. One records whether the word-sense pair shares the same POS tag and another records whether the word-sense pair share the same word.

Knowledge based features. As described in Section 2.2, four knowledge-based word similarity measures based on WordNet are calculated.

Multi-level text Features. The multi-level text features are the same as Ph-W level.

In consideration of the lack of contextual infor-

[4]http://nlp.lsi.upc.edu/asiya/

mation between word-sense pair, we also propose three information enrichment methods in order to generate more effective information for word and sense with the aid of WordNet.

Word Expansion with Synonyms. For the word part in W-Se level, we extract its synonyms with the help of WordNet, then update the values of above basic features if its synonyms achieve higher feature value than the original word itself.

Sense Expansion with Definition. For the sense in W-Se level, we directly use its definition in WordNet to enrich its information. By doing so the similarity estimation of W-Se level can be converted to that of word-phrase level, therefore we use all basic features for Ph-W level described in Section2.3.

Word-Sense Expansion with Definition. Unlike the above two expansion methods which focus only on one part of W-Se level, the third method is to enrich information for word and sense together by using their definitions in WordNet. As before we extract the word definition in terms of its most common concept in WordNet and then replace the initial word with this definition. Then we use all features in Section 2.2.

3 Experiment and Results

We adopt supervised regression model for each cross level. In order to compare the performance of different regression algorithms, we perform 5-fold cross validation on training data for each cross level. We used several regression algorithms including Support Vector Regression (SVR) with 3 different kernels (i.e., linear, polynomial and rbf), Random Forest, Stochastic Gradient Descent (SGD) and Decision Tree implemented in the scikit-learn toolkit (Pedregosa et al., 2011). The system performance is evaluated in Pearson correlation (r) (official measure) and Spearman's rank correlation (ρ).

3.1 Results on Training Data

Table 1 and Table 2 show the averaged performance of different regression algorithms in terms of Pearson correlation (r) and Spearman's rank correlation (ρ) on the training data of P-S level and S-Ph level using 5-fold cross validation, where the standard deviation is given in brackets. The results show that Random Forest performs the best both on P-S level and S-Ph level whether in (r) or (ρ). We also find that the results of P-S level are better than that of S-Ph level, and the reason may be that paragraph and sentence pair contain more information than the sentence and phrase pair.

Regression Algorithm	r (%)	ρ (%)
SVR, ker=rbf	80.70 ($\pm$1.47)	79.90 ($\pm$1.66)
SVR, ker=poly	73.78 ($\pm$1.57)	74.41 ($\pm$1.89)
SVR, ker=linear	80.43 ($\pm$1.13)	79.46 ($\pm$1.51)
Random Forest	**80.92** ($\pm$1.40)	**80.20** ($\pm$2.00)
SGD	77.61 ($\pm$0.76)	77.14 ($\pm$1.49)
Decision Tree	73.23 ($\pm$2.14)	71.84 ($\pm$2.55)

Table 1: Results of different algorithms using 5-fold cross validation on training data of P-S level

Regression Algorithm	r (%)	ρ (%)
SVR, ker=rbf	66.14 ($\pm$5.14)	65.76 ($\pm$5.93)
SVR, ker=poly	58.93 ($\pm$2.29)	63.62 ($\pm$4.15)
SVR, ker=linear	66.78 ($\pm$4.51)	66.34 ($\pm$4.90)
Random Forest	**73.18** ($\pm$5.23)	**70.30** ($\pm$5.51)
SGD	63.18 ($\pm$3.61)	64.80 ($\pm$4.21)
Decision Tree	67.66 ($\pm$6.76)	66.03 ($\pm$6.64)

Table 2: Results of different algorithms using 5-fold cross validation on training data of S-Ph level

Table 3 shows the results of different regression algorithms and different feature sets in terms of r and ρ on the training data of Ph-W level using 5-fold cross validation, where the basic features are denoted as Feature Set A and their combination with word definition expansion features are denoted as Feature Set B. The results show that almost all algorithms performance have been improved by using word definition expansion feature except Decision Tree. This proves the effectiveness of the information enrichment method we proposed in this level. Besides, Random Forest achieves the best performance again with r=44% and ρ=41%. However, in comparison with P-S level and S-Ph level, all scores in Table 3 drop a lot even with information enrichment method. The possible reason may be two: the reduction of information on Ph-W level and our information enrichment method brings in a certain noise as well.

For W-Se level, in order to examine the performance of different information enrichment methods, we perform experiments on 4 different feature sets from A to D, where feature set A contains the basic features, feature set B, C and D add one information enrichment method based on former feature set. Table 4 and 5 present the r and ρ results of 4 feature sets using different regression algorithms. From Table 4 and 5 we see that most correlation scores are below 40% and

Regression Algorithm	r (%)		ρ (%)	
	Feature Set A[1]	Feature Set B[2]	Feature Set A	Feature Set B
SVR, ker=rbf	34.67 (±4.34)	42.62 (±6.36)	33.26 (±4.24)	40.87 (±6.24)
SVR, ker=poly	19.00 (±4.26)	24.06 (±5.55)	21.13 (±4.86)	28.35 (±6.11)
SVR, ker=linear	34.87 (±4.65)	41.91 (±2.05)	35.42 (±5.05)	42.69 (±0.55)
Random Forest	43.17 (±7.72)	**44.00** (±6.88)	40.34 (±5.71)	**41.80** (±6.76)
SGD	26.20 (±3.37)	38.69 (±4.60)	23.55 (±5.01)	38.00 (±2.64)
Decision Tree	39.22 (±7.54)	32.22 (±12.74)	38.90 (±6.03)	31.64 (±10.47)

[1] Feature Set A = basic feature set
[2] Feature Set B = Feature Set A + Word Definition Expansion Features

Table 3: Results of different algorithms using 5-fold cross validation on training data of Ph-W level

the performance of W-Se level is the worst among all these four levels. This illustrates that the less information the texts contain, the worse performance the model achieves. Again the Random Forest algorithm performs the best among all algorithms. Again almost all information enrichment features perform better than Feature set A. This illustrates that these information enrichment methods do help to improve performance. When we observe the three information enrichment methods, we find that feature set C performs the best. In comparison with feature set C, feature set B only used word synonyms to expand information and this expansion is quite limited. Feature set D performs better than B but still worse than C. The reason may be that when we extend sense with its definition, the definition is accurate and exactly represents the meaning of sense. However since a word often contains more than one concepts, and when we use the definition of the most common concept to extend word, such extension may not be correct and the generated information may contain more noise and/or change the original meaning of word.

3.2 Results on Test Data

According to the experiments on training data, we select Random Forest as the final regression algorithm. The number of trees in Random Forest *n* is optimized to 50 and the rest parameters are set to be default. All features in Section 2.2 are used on P-S level, S-Ph level and Ph-W level. For W-Se level, we take all features except word-sense definition expansion feature which has been shown to impair the system performance. For each level, all training examples are used to learn the corresponding regression model. According to the official results released by organizers, Table 6 and Table 7 list the top 3 systems in terms of r (official) and ρ. Our final systems rank the second both in terms of r and ρ and also achieve the second place on P-S level, S-Ph level and Ph-W level, as well as the 4th place on W-Se level in terms of official Pearson correlation.

Team	P-S	S-Ph	Ph-W	W-Se	r Rank
SimCompass	0.811	0.742	0.415	0.356	1
ECNU	0.834	0.771	0.315	0.269	2
UNAL-NLP	0.837	0.738	0.274	0.256	3

Table 6: Pearson Correlation (official) on test data

Team	P-S	S-Ph	Ph-W	W-Se	ρ Rank
SimCompass	0.801	0.728	0.424	0.344	1
ECNU	0.821	0.757	0.306	0.263	2
UNAL-NLP	0.820	0.710	0.249	0.236	6

Table 7: Spearman Correlation on test data

4 Conclusion

We build a supervised Random Forest regression model for each cross level. For P-S and S-Ph level, we adopt the ensemble of heterogeneous similarity features, i.e., string features, knowledge based features, corpus based features, syntactic features, machine translation based features, multi-level text features and other features to capture the semantic similarity between two texts with distinctively different lengths. For Ph-W and W-Se level, we propose information enrichment methods to lengthen original texts in order to generate more semantic features, which has been proved to be effective. Our submitted final systems rank the 2nd out of 18 teams both on Pearson Rank (official rank) and Spearman Rank, and also rank the second place on P-S level, S-Ph level and Ph-W level, as well as the 4th place on W-Se level in terms of Pearson correlation. In future work we will focus on information enrichment methods which bring in more accurate information and less noises.

Acknowledgments

This research is supported by grants from National Natural Science Foundation of China

Regression Algorithm	Feature Set A[1]	Feature Set B[2]	Feature Set C[3]	Feature Set D[4]
SVR, ker=rbf	29.85 (±7.29)	34.49 (±5.55)	36.80 (±6.46)	22.19 (±6.49)
SVR, ker=poly	24.62 (±3.63)	29.27 (±3.53)	26.55 (±1.27)	25.89 (±5.63)
SVR, ker=linear	29.58 (±5.88)	34.87 (±3.97)	35.96 (±1.75)	34.57 (±3.75)
Random Forest	22.87 (±5.59)	33.97 (±1.78)	**40.43** (±3.00)	37.54 (±3.20)
SGD	26.32 (±7.31)	27.36 (±6.44)	32.50 (±6.02)	18.00 (±6.13)
Decision Tree	23.40 (±5.65)	26.33 (±3.86)	33.64 (±6.97)	31.86 (±3.95)

[1] Feature Set A = basic feature set
[2] Feature Set B = Feature Set A + Synonym Expansion
[3] Feature Set C = Feature Set B + Sense Definition Expansion Features
[4] Feature Set D = Feature Set C + Word-Sense Definition Expansion Features

Table 4: Results of different algorithms using 5-fold CV on training data of W-Se level (r (%))

Regression Algorithm	Feature Set A	Feature Set B	Feature Set C	Feature Set D
SVR, ker=rbf	28.41 (±8.99)	29.61 (±6.23)	34.18 (±6.36)	22.90 (±6.78)
SVR, ker=poly	23.05 (±7.53)	22.47 (±4.47)	21.63 (±4.37)	25.37 (±7.25)
SVR, ker=linear	27.29 (±7.02)	31.79 (±4.00)	34.75 (±3.55)	34.19 (±3.06)
Random Forest	19.66 (±6.75)	31.98 (±3.21)	**38.57** (±3.60)	37.56 (±3.15)
SGD	24.12 (±7.98)	24.62 (±6.36)	29.27 (±5.86)	23.05 (±11.23)
Decision Tree	22.30 (±5.25)	25.09 (±3.64)	31.99 (±7.81)	30.51 (±5.27)

Table 5: Results of different algorithms using 5-fold CV on training data of W-Se level (ρ (%))

(No.60903093) and Shanghai Knowledge Service Platform Project (No. ZF1213).

References

Carmen Banea, Samer Hassan, Michael Mohler, and Rada Mihalcea. 2012. Unt: A supervised synergistic approach to semantic text similarity. pages 635–642. First Joint Conference on Lexical and Computational Semantics (*SEM).

Daniel Bär, Chris Biemann, Iryna Gurevych, and Torsten Zesch. 2012. Ukp: Computing semantic textual similarity by combining multiple content similarity measures. pages 435–440. First Joint Conference on Lexical and Computational Semantics (*SEM).

Michael Heilman and Nitin Madnani. 2012. Ets: Discriminative edit models for paraphrase scoring. pages 529–535. First Joint Conference on Lexical and Computational Semantics (*SEM).

Aminul Islam and Diana Inkpen. 2008. Semantic text similarity using corpus-based word similarity and string similarity. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 2(2):10.

Thomas K Landauer, Darrell Laham, Bob Rehder, and Missy E Schreiner. 1997. How well can passage meaning be derived without using word order? a comparison of latent semantic analysis and humans. In *Proceedings of the 19th annual meeting of the Cognitive Science Society*, pages 412–417.

Claudia Leacock and Martin Chodorow. 1998. Combining local context and wordnet similarity for word sense identification. *WordNet: An electronic lexical database*, 49(2):265–283.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the 15th international conference on Machine Learning*, volume 1, pages 296–304. San Francisco.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Fabian Pedregosa, Gaël. Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Mohammad Taher Pilehvar, David Jurgens, and Roberto Navigli. 2013. Align, disambiguate and walk: A unified approach for measuring semantic similarity. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL 2013)*.

Frane Šaric, Goran Glavaš, Mladen Karan, Jan Šnajder, and Bojana Dalbelo Bašic. 2012. Takelab: Systems for measuring semantic text similarity. pages 441–448. First Joint Conference on Lexical and Computational Semantics (*SEM).

Zhibiao Wu and Martha Palmer. 1994. Verbs semantics and lexical selection. In *Proceedings of the 32nd annual meeting on Association for Computational Linguistics*, pages 133–138. Association for Computational Linguistics.

Tian Tian Zhu and Man Lan. 2013. Ecnucs: Measuring short text semantic equivalence using multiple similarity measurements. *Atlanta, Georgia, USA*, page 124.