SSMT: A Machine Translation Evaluation View to Paragraph-to-Sentence Semantic Similarity

Pingping Huang
Department of Linguistic Engineering
School of Software and Microelectronics
Peking University, China
girlhpp@163.com

Baobao Chang
Key Laboratory of Computational
Linguistics, Ministry of Education
Institute of Computational Linguistics
Peking University, China
chbb@pku.edu.cn

Abstract

This paper presents the system SSMT measuring the semantic similarity between a paragraph and a sentence submitted to the SemEval 2014 task3: Cross-level Semantic Similarity. The special difficulty of this task is the length disparity between the two semantic comparison texts. We adapt several machine translation evaluation metrics for features to cope with this difficulty, then train a regression model for the semantic similarity prediction. This system is straightforward in intuition and easy in implementation. Our best run gets 0.808 in Pearson correlation. METEOR-derived features are the most effective ones in our experiment.

1 Introduction

Cross level semantic similarity measures the similarity between different levels of text unit, for example, between a document and a paragraph, or between a phrase and a word.

Paragraph and sentence are the natural language units to convey opinions or state events in daily life. We can see posts on forums, questions and answers in Q&A communities and customer reviews on E-commerce websites, are mainly organised in these two units. Better similarity measurement across them will be helpful in clustering similar answers or reviews.

The paragraph-to-sentence semantic similarity subtask in SemEval2014 task3 (Jurgens et al., 2014) is the first semantic similarity competition across these two language levels. The special difficulty of this task is the length disparity between the compared pair: a paragraph contains 3.67 times the words of a sentence on average in the training set.

Semantic similarity on different levels, for example, on word level (Mikolov et al., 2013), sentences level (Bär et al., 2012), document level (Turney and Pantel, 2010), have been well studied, yet methods on one level can hardly be applied to a different level, let alone be applied for the cross-level tasks. The work of Pilehvar et al.(2013) was an exception. They proposed a unified method for semantic comparison at multi-levels all the way from comparing word senses to comparing text documents

Our work is inspired by automatic machine translation(MT) evaluation, in which different metrics are designed to compare the adequacy and fluency of a MT system's output, called hypothesis, against a gold standard translation, called reference. As MT evaluation metrics measure sentence pair similarity, it is a natural idea to generalize them for paragraph-sentence pair.

In this paper, we follow the motivations of several MT evaluation metrics yet made adaption to cope with the length disparity difficulty of this task, and combine these features in a regression model. Our system SSMT (**S**emantic **S**imilarity in view of **M**achine **T**ranslation evaluation) involves no extensive resource or strenuous computation, yet gives promising result with just a few simple features.

2 Regression Framework

In our experiment, we use features adapted from some MT evaluation metrics and combine them in a regression model for the semantic similarity measurement. We exploit the following two simple models:

A linear regression model is presented as:

$$y = w_1 x_i + w_2 x_i .. + w_n x_n + \varepsilon$$

Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pages 585–589,
Dublin, Ireland, August 23-24, 2014.

A log-linear model is presented as:

$$y = x_1^{w_1} \cdot x_2^{w_2} .. \cdot x_n^{w_n} \cdot e^{\varepsilon}$$

Where y is the similarity score, $\{x_1, x_2.., x_n\}$ are the feature values.

We can see that in a log-linear model, if any feature x_i get a value of 0, the output y will suck in 0 forever no matter what the values other features get. In our experiment we resort to smoothing to avoid this "0-trap" for some features (Section 4.3).

3 Features

MT evaluation metrics vary from lexical level to syntactic level to semantic level. We consider only lexical ones to avoid complicated steps like parsing or semantic role labelling, which are computational expensive and may bring extra noise.

But instead of directly using the MT evaluation metrics, we use the factors in them as features, the idea is that the overall score of the original metric is highly related to the length of both of the compared pair, but its factors are often related to the length of just one side yet still carry useful similarity information.

3.1 BLEU-Derived Features

As the most wildly used MT evaluation metric, BLEU (Papineni et al., 2002) uses the geometric mean of *n*-gram precisions to measure the hypotheses against references. It is a corpus-based and precision-based metric, and uses "*brevity penalty*" as a replacement for recall. Yet this *penalty* is meaningless on sentence level. Therefore we considers only the precision factors in BLEU:

$$P_{n_{BLEU}} = \frac{Ngram_{ref} \cap Ngram_{hyo}}{Ngram_{ref}}$$

We use the modified *n*-gram precision here and regard "paragraph" as "reference", and "sentence" as the "hypothesis". *N*= 1,2,3,4. We call these four features BLEU-derived features.

3.2 ROUGE-L-Derived Features

ROUGE-L (Lin and Och, 2004) measures the largest common subsequence(LCS) between a compared pair. BLEU implies the *n*-gram to be consecutive, yet ROUGE-L allows for gaps between them. By considering only in-sequence words, ROUGE-L captures sentence level structure in a natural way, then:

$$\begin{aligned} R_{lcs} &= \frac{LCS(ref, hyo)}{length(hyo)} \\ P_{lcs} &= \frac{LCS(ref, hyo)}{length(ref)} \\ F_{lcs} &= \frac{(1+\beta^2)R_{lcs}P_{lcs})}{R_{lcs} + \beta^2 P_{lcs}} \end{aligned}$$

Where $LCS(ref, hoy)$ is the length of LCS of the compared pair. We set $\beta = 1$, which means we don't want to make much distinction between the "reference" and "hypothesis" here. We call these three features ROUGE-L-derived features.

3.3 ROUGE-S-Derived Features

ROUGE-S (Lin and Och, 2004) uses skip-bigram co-occurrence statistics for similarity measurement. One advantage of skip-bigram over BLEU is that it does not require consecutive matches but is still sensitive to word order. Given the reference of length n, and hypothesis of length m, then:

$$\begin{aligned} P_{skip2} &= \frac{skip2(ref, hyo)}{C(m, 2)} \\ R_{skip2} &= \frac{skip2(ref, hyo)}{C(n, 2)} \\ F_{skip2} &= \frac{(1+\beta^2)P_{skip2}R_{skip2}}{R_{skip2} + \beta^2 P_{skip2}} \end{aligned}$$

Where C is combination, and $skip2(ref, hyo)$ is the number of common skip-bigrams. We also set $\beta = 1$ here, and call these three indicators ROUGE-S-derived features.

3.4 METEOR-Derived Features

METEOR (Banerjee and Lavie, 2005) evaluates a hypothesis by aligning it to a reference translation and gives sentence-level similarity scores. It uses a generalized concept of unigram mapping that matches words in the following types: exact match on words surface forms , stem match on words stems, synonym match according to the synonym sets in WordNet, and paraphrase match (Denkowski and Lavie, 2010).

METEOR also makes distinction between content words and function words. Each type of match m_i is weighted by w_i, let $(m_i(h_c), m_i(h_f))$ be the number of content and function words covered by this type in the hypothesis, and

$(m_i(r_c), m_i(r_f))$ be the counts in the reference, then:

$$P = \frac{\sum_{i=1} w_i \cdot (\delta \cdot m_i(h_i) + (1-\delta) \cdot m_i(h_f))}{\delta \cdot \mid h_c \mid + (1-\delta) \cdot \mid h_f \mid}$$

$$R = \frac{\sum_{i=1} w_i \cdot (\delta \cdot m_i(r_i) + (1-\delta) \cdot m_i(r_f))}{\delta \cdot \mid r_c \mid + (1-\delta) \cdot \mid r_f \mid}$$

$$F_{mean} = \frac{P \cdot R}{\alpha P + (1-\alpha) R}$$

To account for word order difference, the fragmentation penalty is calculated using the total number of matched words(m) and the number of chunks[1](ch) in the hypothesis:

$$Pen = \gamma \cdot \left(\frac{ch}{m}\right)^{\beta}$$

And the final METEOR score is:

$$Score = (1 - Pen) \cdot F_{mean}$$

Parameters $\alpha, \beta, \gamma, \delta$and $w_i...w_n$ are tuned to maximize correlation with human judgements (Denkowski and Lavie, 2014). We use Meteor1.5 system[2] for scoring. Parameters are tuned on WMT12, and the paraphrase table is extracted on the WMT data.

We use the p, r, $frag(frag = ch/m)$ and $score$ as features and call them METEOR-derived features.

4 Experiment and Discussion

4.1 Data Set

The SemEval2014 task3 subtask gives a training set of 500 paragraph-sentence pairs, with human annotated continuous score of $0 - 4$. These pairs are labelled with genres of "Newswire/ cqa[3]/ metaphoric/ scientific/ travel/ review". Systems are asked to predict the similarity scores for 500 pairs in the test set. Performance is evaluated in Pearson correlation and Spearman correlation.

4.2 Data Processing

To avoid meaningless n-gram match "*the a*", or words surface form difference, we employ very simple data processings here: for features derived from BLEU, ROUGE-L and ROUGE-S, we remove stop words and stem the sentences with coreNLP[4]. For METEOR-derived features, we use the tool's option for text normalization before matching.

4.3 Result

Though texts with different genres may have different regression parameters, we just train one model for all for simplicity. Table 1 compares the result. Run1 is submitted as SSMT in the official evaluation. It's a log-linear model. We choose more dense features for log-linear model and use smoothing to avoid the "0-trap" mentioned in (Section 2). The features include $P_{1,2_{BLEU}}$,$P_{ROUGE-L}$,$P_{ROUGE-S}$ 4 features, and 4 METEOR-derived features, altogether 8 features. When calculation the first 4 features, we plus 1 to both numerator and denominator as smoothing. Run2 is a linear-regression model with the same features as Run1. Run3 is a simple linear regression model, which is free from the "0-trap", thus we use all the 14 features without smoothing. We use Matlab for regression. The baseline is officially given using LCS.

Run	Regression	Pearson	Spearman
Baseline	LCS	0.527	0.613
run1	log-linear	0.789	0.777
run2	linear	0.794	0.777
run3	linear	**0.808**	**0.792**

Table 1: System Performance.

4.4 System Analysis

We compares the effectiveness of different features in a linear regression model. Table 2 shows the result. "All" refers to all the features, "-METEOR" means the feature set excludes METEOR-derived features. We can see the METEOR-derived features are the most effective ones here.

Figure 1 shows the performance of our system submitted as SSMT in the SemEval2014 task3 competition. It shows quite good correlation with the gold standard.

A well predicted example is the #trial-p2s-5 pair in the trial set:

Paragraph: *Olympic champion Usain Bolt regained his 100m world title and won a fourth individual World Championships gold with a season's best of 9.77 seconds in Moscow. In heavy*

[1] Chunk is defined as a series of matched unigrams that is contiguous and identically ordered in both sentences

[2] https://www.cs.cmu.edu/ alavie/METEOR/

[3] cqa:Community Question Answering site text

[4] http://nlp.stanford.edu/software/corenlp.shtml

Feature	Pearson	Spearman
All	0.808	0.792
- METEOR	**0.772**	**0.756**
- ROUGE-L	0.802	0.789
- ROUGE-S	0.807	0.793
- BLEU	0.807	0.790

Table 2: Effectiveness of Different Features. "-METEOR" means the feature set excluding METEOR-derived features.

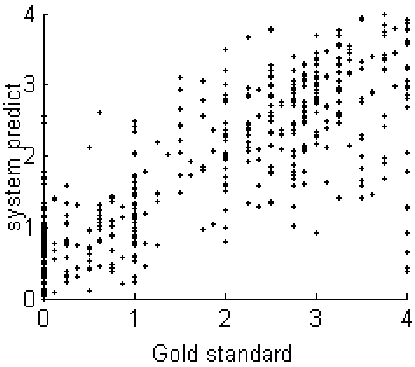

Figure 1: Result Scatter of SSMT.

rain, the 26-year-old Jamaican made amends for his false start in Daegu two years ago and further cemented his status as the greatest sprinter in history. The six-time Olympic champion overtook Justin Gatlin in the final stages, forcing the American to settle for silver in 9.85. Bolt's compatriot Nesta Carter (9.95) claimed bronze, while Britain's James Dasaolu was eighth (10.21).

Sentence: *Germany's Robert Harting beats Iran's Ehsan Hadadi and adds the Olympic discus title to his world crown.*

The system gives a prediction of 1.253 against the gold standard 1.25. We can see that topic words like "*Olympic*" , "*world crown*", "*beats*" in the short text correspond to expressions of "*world title* " , "*champion*" across several sentences in the long text, but this pair of texts are not talking about the same event. The model captures and models this commonness and difference very well .

But Figure 1 also reveals an interesting phenomenon: the system seldom gives the boundary scores of 0 or 4. In other words, it tends to overscore or underscore the boundary conditions. An example in point is the #trial-p2s-17 pair in the trial data, it is actually the worst predicted pair by our system in the trail set:

Paragraph: *A married couple who met at work is not a particularly rare thing. Three in ten workers who have dated a colleague said in a recent survey by CareerBuilder.com that their office romance eventually led to marriage.*

Sentence: *Marrying a coworker isn't uncommon given that 30% of workers who dated a coworker ended up marrying them.*

The system gives a 1.773 score against the gold standard of 4. It should fail to detect the equality of expressions between "*three in ten*" and "*30%*". Thus better detection of phrase similarity is desired. We think this is the main reason to underscore the similarity. For test pairs with the genre of "Metaphoric", the system almost underscores all of them. This failure has been expected, though. Because "Metaphoric" pairs demand full understanding of the semantic meaning and paragraph structure, which is far beyond the reach of lexical match metrics.

5 Conclusion

MT evaluation metrics have been directly used as features in paraphrase (Finch et al., 2005) detection and sentence pair semantic comparison (Souza et al., 2012). But paragraph-to-sentence pair faces significant length disparity, we try a way out to alleviate this impact yet still follow the motivations underlying these metrics. By factorizing down the original metrics, the linear model can flexibly pick out factors that are not sensitive to the length disparity problem.

We derive features from BLEU, ROUGE-L, ROUGE-S and METEOR, and show that METEOR-derived features make the most significant contributions here. Being easy and light, our submitted SSMT achieves 0.789 in Pearson and 0.777 in Spearman correlation, and ranks 11 out of the 34 systems in this subtask. Our best try achieves 0.808 in Pearson and 0.786 in Spearman correlation.

Acknowledgements

This work is supported by National Natural Science Foundation of China under Grant No.61273318 and National Key Basic Research Program of China 2014CB340504.

References

Andrew Finch, Yong S. Hwang, Eiichiro Sumita. Using machine translation evaluation techniques to determine sentence-level semantic equivalence. *Proceedings of the Third International Workshop on Paraphrasing(IWP2005)*, 2005: 17-24.

Chin Y. Lin,Franz J. Och. Automatic evaluation of machine translation quality using longest common subsequence and skip-bigram statistics. *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics. ACL*, 2004: 605.

Daniel Bär, Chris Biemann, Iryna Gurevych, et al. Ukp: Computing semantic textual similarity by combining multiple content similarity measures. *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation. ACL*, 2012: 435-440.

David Jurgens, Mohammad Taher Pilehvar, and Roberto Navigli. SemEval-2014 Task 3: Cross-Level Semantic Similarity. *In Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval-2014).*, August 23-24, 2014, Dublin, Ireland.

George Miller,Christiane Fellbaum. WordNet. http://wordnet.princton.edu/, 2007.

José G C de Souza, Matteo Negri, Yashar Mehdad. FBK: machine translation evaluation and word similarity metrics for semantic textual similarity. *Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation. ACL*, 2012: 624-630.

Kishore Papineni, Salim Roukos, Todd Ward, et al. BLEU: a method for automatic evaluation of machine translation. *Proceedings of the 40th annual meeting on association for computational linguistics. ACL*, 2002: 311-318.

Michael Denkowski, Alon Lavie. Extending the METEOR machine translation evaluation metric to the phrase level. *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics. ACL*, 2010: 250-253.

Michael Denkowski, Alon Lavie. Meteor Universal: Language Specific Translation */Evaluation for Any Target Language translation. Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*, 2014.

Mohammad T Pilehvar, David Jurgens, Roberto Navigli. Align, Disambiguate and Walk: A Unified Approach for Measuring Semantic Similarity *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics,ACL*, 2013: 1341-1351.

Peter D. Turney and Patrick Pantel. From frequency to meaning: Vector space models of semantics *Artificial Intelligence Research*, 2010. 37(1): 141-188

Satanjeev Banerjee, Alon Lavie. METEOR: an automatic metric for MT Evaluation with improved correlation with human judgements. *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization.*, 2005: 65-72.

Tomas Mikolov, Kai Chen, Greg Corrado, et al. Efficient estimation of word representations in vector space. 2013. *arXiv preprint arXiv:1301.3781*,