HLTC-HKUST: A Neural Network Paraphrase Classifier using Translation Metrics, Semantic Roles and Lexical Similarity Features

Dario Bertero, Pascale Fung
Human Language Technology Center
Department of Electronic and Computer Engineering
The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong
dbertero@ust.hk, pascale@ece.ust.hk

Abstract

This paper describes the system developed by our team (HLTC-HKUST) for task 1 of SemEval 2015 workshop about paraphrase classification and semantic similarity in Twitter. We trained a neural network classifier over a range of features that includes translation metrics, lexical and syntactic similarity score and semantic features based on semantic roles. The neural network was trained taking into consideration in the objective function the six different similarity levels provided in the corpus, in order to give as output a more fine-grained estimation of the similarity level of the two sentences, as required by subtask 2. With an F-score of 0.651 in the binary paraphrase classification subtask 1, and a Pearson coefficient of 0.697 for the sentence similarity subtask 2, we achieved respectively the 6th place and the 3rd place, above the average of what obtained by the other contestants.

1 Introduction

Paraphrase identification is the problem to determine whether two sentences have the same meaning, and is the objective of the task 1 of SemEval 2015 workshop (Xu et al., 2015).

Conventionally this task has been mainly evaluated on the Microsoft Research Paraphrase corpus (Dolan and Brockett, 2005), which consists of pairs of sentences taken out from news headlines and articles. News domain sentences are usually grammatically correct and of average to long length. The current state-of-the-art method to our knowledge on this corpus (Ji and Eisenstein, 2013) trains an SVM over latent semantic vectors, lexical and syntactic similarity features. Although their main objective was to show the effectiveness of a method based on latent semantic analysis, it is also evident that other features pertinent to different aspects of sentence similarity are able to boost the results. Previously Socher et al. (2011) used a recursive autoencoder to similarly obtain a vector representation of each sentence, again combining other lexical similarity features to improve the results. Other methods, such as Madnani et al. (2012) or Wan et al. (2006) used instead a more traditional supervised classification approach over different sets of features and different classifiers, most of which improved previous results.

Task 1 of SemEval 2015 workshop required to evaluate paraphrases on a new corpus, consisting of sentences taken from Twitter posts (Xu et al., 2014). Twitter sentences notoriously differ from those taken from news articles: the 140 characters limit makes the sentences short, with few words, lots of different abbreviations; they also include many misspelled and invented words, and often lack a correct grammatical structure. Another important difference is the six-level classification labels provided, compared to the binary labels of MSRP corpus, which allows a fine-grained evaluation of the similarity level between the sentences.

The task was divided into two subtasks. Subtask 1 was the classical binary paraphrase classification task, where given a pair of sentences the system had to identify if it is a paraphrase or not. Subtask 2 instead required the system to provide a score in the range $[0, 1]$ that measures the actual similarity level of the two sentences.

Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 23–28,
Denver, Colorado, June 4-5, 2015.

2 System Description

We chose a supervised machine learning strategy based on a multi-view set of features. Our first goal was to select the features in order to get a complete estimation of lexical, syntactic and semantic similarity between any given pair of sentences. In particular we were interested in what roles semantic features can play in this task. The second goal was to make use of a classifier which can take full advantage of the six level labeling provided in order to have good performance in both subtasks, identified in an artificial neural network.

2.1 Lexical and Syntactic Similarity Features

The first set of lexical features includes three binary indexes obtained from the analysis of the numerical tokens: the first of them is 1 if they are the same in both sentences or there are not any, the second is 1 only if they are the same, and the third is 1 if the tokens representing numbers of one sentences are the subset of the other (Socher et al., 2011). Two other features include the percentage of overlapping tokens, and the difference in sentence length. Another feature considers the word order: starting from one sentence we align the tokens that matches with the other sentence, and for each aligned pair we take the average of the differences of the absolute positions of the two elements, normalized by the length of the first sentence, and we do the same switching the order of the two sentences. Another group of features involves WordNet word synonym sets (Miller, 1995). We take from them, separately for nouns and verbs, the average of the path similarity scores obtained, among all word alignments, from the one which gives the maximum score. When the two words in the pair to be scored have multiple synonym sets we select the two sets that again are giving the highest score. Finally, in order to include an estimation of the level of similarity in the syntax parse tree of the sentences, we use the parse tree edit distance from the Zhang-Shasha algorithm (Zhang and Shasha, 1989; Wan et al., 2006).

2.2 Semantic Similarity Features

The way we evaluate the semantic similarity of each pair of sentences is through the analysis of the semantic roles. The first feature we choose in this sense is the semantic role based MEANT machine translation score (Lo et al., 2012), effective to provide, as shown by various experiments, a translation evaluation closer to human judges. This metric first annotates each sentence with semantic roles (Pradhan et al., 2004), then aligns them and computes a similarity score only within the aligned frames (Fung et al., 2007) using the Jaccard coefficient (Tumuluru et al., 2012). Another set of features is obtained by looking at the semantic roles themselves and their alignment without looking at the content: these include the percentage of semantic roles of one sentence that are also present in the other, the percentage of correct pairs of semantic roles after the alignment operated for MEANT, and a binary feature equal to 1 in case the semantic parser fails to give any output for at least one of the sentences. In this last case all the other features based on semantic roles are 0 except the MEANT score which is set to the value of the Jaccard coefficient between the whole sentences (Lo and Wu, 2013).

2.3 Translation Metrics

Previous work (Finch et al., 2005; Madnani et al., 2012) have shown that machine translation evaluation metrics are useful for the paraphrase recognition task, due to their ability to capture useful similarity information to correctly classify the sentence pairs.

The various translation metrics all take into account different aspects of sentence similarities. BLEU (Papineni et al., 2002) and the subsequent evaluation metrics such as NIST (Goutte, 2006) and SEPIA (Habash and Elkholy, 2008) look at n-gram overlaps between the source and the target sentences. While the most basic BLEU takes into consideration only n-gram overlap, the other metrics also consider synonyms, stemming, simple paraphrase patterns and the syntactic structure of the n-grams. Yet another set of metrics are based instead on different principles: TER (Snover et al., 2006) and TERp (Snover et al., 2009) count the number of edits needed to transform a sentence into the other, MAXSIM (Chan and Ng, 2008) evaluates lexical similarity performing a word-by-word matching and finding out how much the aligned words are similar in each meaning, BADGER (Parker, 2008) the distance between the compression of each sentence obtained from the Burrows-Wheeler transform algorithm (Burrows and

Wheeler, 1994), and MEANT which, as discussed in the previous section, scores the similarity of aligned semantic frames.

For each pair of sentences the scores are calculated first taking one of the sentences as the reference and the other as the sample and then vice-versa. Both scores are included as distinct features except in the case of BADGER, as it computes a distance between two objects without taking into account the direction. In case of BLEU and NIST we use the scores from unigrams up to 4-grams for BLEU (Madnani et al., 2012) and up to the maximum order which gives at least one result different than zero for NIST.

2.4 Classifier

To classify the sentence pairs we design a feedforward neural network. One of the main properties of the neural network is its ability to learn complex functions of the input values (Hornik et al., 1989). It follows that in our task, given the combination of features, the network would learn how to combine them effectively and take advantage of their mutual interaction. The neural network can also be trained using an objective function that takes into consideration a label not just binary but which can take multiple values in a given range. Therefore it has a good ability to determine as output a precise estimation of the similarity level of the sentence pair, particularly useful in subtask 2. During our experiments the results we obtained in the binary classification task over the development set with the neural network were always at least slightly higher than those obtained with an SVM we used as a comparison system, further justifying our neural network choice.

We choose a two layer standard configuration (hidden and output layer), where we fix the size of the hidden layer large enough at three times the size of the input layer; the hyperbolic tangent ($\tanh$) and the sigmoid are used respectively as the non-linear activation functions of the hidden layer and the output layer. Due to this choice the output assumes values in the interval $[0, 1]$, which is also exactly the output range required in subtask 2. The network weights, with the exception of the ones associated to the bias terms set at zero, are initialized (Glorot and Bengio, 2010) with uniform values in the range:

$$w_{t=0} \in \left[-\alpha \left(\frac{6}{n_{in} + n_{out}} \right)^{\frac{1}{2}}, \alpha \left(\frac{6}{n_{in} + n_{out}} \right)^{\frac{1}{2}} \right] \qquad (1)$$

Where $\alpha = 1$ in case the activation function is the hyperbolic tangent, and $\alpha = 4$ with the sigmoid. We train the model using standard backpropagation algorithm, taking the cross-entropy as the cost objective function:

$$E = -l \ \log(y) - (1 - l) \log(1 - y) + R \qquad (2)$$

where y is the network output, l the objective value (both in the range $[0, 1]$), and R is an L2 regularization term.

3 Experiments

3.1 Corpus

We made use of the corpus provided for the contest (Xu et al., 2014), made of a training set of 13063 sentence pairs, a development set of 4727 pairs, and a test set of 972 pairs released a few days before the deadline without the labels. Each pair of sentences was labeled by five users via Amazon Mechanical Turk, hence providing a six-level classification label (from $(5, 0)$ when all the five user classify the pair as a paraphrase, to $(0, 5)$ when none of them identifies the pair to be a paraphrase).

3.2 Experimental Setup

The neural network was setup with a hidden layer dimension of three times the input. The development set was used to tune the L2 regularization coefficient, set at $\gamma = 0.01$, as well as the learning rate and the other hyperparameters, and to have a measure of improvement against the official thresholding baseline provided for the task (Das and Smith, 2009). To implement the neural network we used THEANO Python toolkit (Bergstra et al., 2010).

We train the network with all the sentences provided in the training set. The objective label of the cross-entropy objective function was set to 1.0 for pairs labeled $(5, 0)$ and $(4, 1)$, 0.75 for pairs labeled $(3, 2)$, 0.5 for pairs labeled $(2, 3)$ and 0.0 for pairs labeled $(0, 5)$. This choice allowed a more fine training for task 2, where a continuous similarity value must be estimated, without altering too much the behavior in the binary estimation task 1.

The training procedure was repeated several times, each time with a different random initialization of the weights and with a different random pair order. In order to avoid overfitting, in each run the training was

	Subtask 1			Subtask 2			
Description	Precision	Recall	F-score	Precision	Recall	F-score	Pearson
Subtask 1 best (ASOBEK)	0.680	0.669	0.674	0.732	0.531	0.616	0.475
Subtask 2 best (MITRE)	0.569	0.806	0.667	0.750	0.686	0.716	0.619
Our method, run 2	0.574	0.754	0.652	0.738	0.611	0.669	0.545
Our method, run 1	0.594	0.720	0.651	0.697	0.657	0.676	0.563
Baseline (Das and Smith, 2009)	0.679	0.520	0.589	0.674	0.543	0.601	0.511
Contest average result	0.600	0.626	0.581	0.645	0.626	0.631	0.483

Table 1: Result comparison between our method and the winners of subtask 1 and subtask 2.

stopped when the best results on the development set were obtained. The final results were taken from the run that yielded the best accuracy, and in case of tie the best F1 score, on the development set for subtask 1.

Run 2 instead was an attempt to include latent semantic vectors obtained through the procedure described in Ji and Eisenstein (2013) and added to the network from an extra layer whose output was concatenated to the features input vector.

3.3 Results and Discussion

F-measure and Pearson coefficient were the official evaluation metrics used to rank respectively subtask 1 and subtask 2. In subtask 1 – binary evaluation of the sentence pairs – we achieved an F-score of 0.651 and ranked 6th over 18 methods, the best method (ASOBEK) achieved an F-score of 0.674. In subtask 2, which was aimed at finding a similarity score in the range $[0, 1]$, with a Pearson coefficient of 0.563 we reached the 3rd place among 13 methods (the other five provided only a binary output), with the winner (MITRE) obtaining a Pearson score of 0.619. A summary and comparison of our results with the winners of the two subtasks, with the average results and with the supervised official baseline (n-gram overlapping features with logistic regression from Das and Smith (2009)) is shown in table 1. For both tasks our results are above the average both in term of ranking and average results.

Semantic features were useful to identify paraphrases, as they improved the accuracy and F-score on the development set by 0.6%. But often the shallow semantic parser failed to give an output for many sentences, limiting their potential contribution. This is due to two main reasons. The first one is the imperfect accuracy of the semantic parser itself, also observed in previous experiments where we employed it, which fails to analyze sentences containing certain patterns and predicates. The second reason, more specific to Twitter domain, is that some sentences lack a valid predicate or a proper grammatical structure. This prevents the semantic parser from giving an accurate output.

The inclusion on latent semantic features in run 2 proved to be ineffective, as it improved subtask 1 F-score by less than 0.001, and gave a worse performance in subtask 2. During the evaluation phase other experiments were tried as using the latent semantic vectors of Guo and Diab (2012), or using the vectors as described in Ji and Eisenstein (2013) instead of the extra layer, and other modifications, all without obtaining any perceptible improvement when the system was tested on the development set. The non-perfect implementation and usage of these features, together with the fact they might not be suitable to be applied to Twitter domain, may explain this lack of improvement.

4 Conclusions

We have used a neural network classifier, with a combination of multiple views of lexical, syntactic and semantic information, as the system which participated in SemEval 2015 task 1, whose goal was to classify paraphrases in Twitter. The inaccurate semantic parsing is the main reason which prevented us from obtain higher results. A possible future directions that can improve the quality of the semantic roles annotations, apart from improving the semantic parser, is to apply an effective lexical normalization method (such as Han and Baldwin (2011)), and eventually find ways to reconstruct the predicate in case it is missing.

Acknowledgments

This work is partially funded by the Hong Kong PhD Fellowship Scheme and by grant number 1314159-0PAFT20F003 of the Ping An Research Institute.

References

James Bergstra, Olivier Breuleux, Frédéric Bastien, Pascal Lamblin, Razvan Pascanu, Guillaume Desjardins, Joseph Turian, David Warde-Farley, and Yoshua Bengio. Theano: a CPU and GPU Math Expression Compiler. In *Proceedings of the Python for Scientific Computing Conference (SciPy)*, June 2010.

Michael Burrows and David J. Wheeler. A Block-sorting Lossless Data Compression Algorithm. Technical report, 1994.

Yee Seng Chan and Hwee Tou Ng. Maxsim: A Maximum Similarity Metric for Machine Translation Evaluation. In *ACL*, pages 55–62, 2008.

Dipanjan Das and Noah A. Smith. Paraphrase Identification as Probabilistic Quasi-Synchronous Recognition. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1*, pages 468–476, 2009.

William B. Dolan and Chris Brockett. Automatically Constructing a Corpus of Sentential Paraphrases. In *Proc. of IWP*, 2005.

Andrew Finch, Young-Sook Hwang, and Eiichiro Sumita. Using Machine Translation Evaluation Techniques to Determine Sentence-level Semantic Equivalence. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*, pages 17–24, 2005.

Pascale Fung, Zhaojun Wu, Yongsheng Yang, and Dekai Wu. Learning Bilingual Semantic Frames: Shallow Semantic Parsing vs. Semantic Role Projection. In *11th Conference on Theoretical and Methodological Issues in Machine Translation (TMI 2007)*, pages 75–84, 2007.

Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010.

Cyril Goutte. Automatic Evaluation of Machine Translation Quality. *Presentation at the European Community, Xerox Research Centre Europe, on January*, 2006.

Weiwei Guo and Mona Diab. Modeling Sentences in the Latent Space. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*, pages 864–872, 2012.

Nizar Habash and Ahmed Elkholy. Sepia: Surface Span Extension to Syntactic Dependency Precision-based mt Evaluation. In *Proceedings of the NIST metrics for machine translation workshop at the association for machine translation in the Americas conference, AMTA-2008. Waikiki, HI*, 2008.

Bo Han and Timothy Baldwin. Lexical Normalisation of Short Text Messages: Makn Sens a #Twitter. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 368–378, 2011.

Kurt Hornik, Maxwell Stinchcombe, and Halbert White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

Yangfeng Ji and Jacob Eisenstein. Discriminative Improvements to Distributional Sentence Similarity. In *EMNLP*, pages 891–896, 2013.

Chi-kiu Lo and Dekai Wu. Meant at wmt 2013: A tunable, accurate yet inexpensive semantic frame based mt evaluation metric. In *8th Workshop on Statistical Machine Translation*, page 422, 2013.

Chi-kiu Lo, Anand Karthik Tumuluru, and Dekai Wu. Fully Automatic Semantic mt Evaluation. In *Proceedings of the Seventh Workshop on Statistical Machine Translation*, pages 243–252, 2012.

Nitin Madnani, Joel Tetreault, and Martin Chodorow. Re-examining Machine Translation Metrics for Paraphrase Identification. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 182–190, 2012.

George A. Miller. Wordnet: A Lexical Database for English. *Communications of the ACM*, 38(11): 39–41, 1995.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a Method for Automatic

Evaluation of Machine Translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318, 2002.

Steven Parker. Badger: A New Machine Translation Metric. *Metrics for Machine Translation Challenge*, 2008.

Sameer S. Pradhan, Wayne Ward, Kadri Hacioglu, James H. Martin, and Daniel Jurafsky. Shallow Semantic Parsing using Support Vector Machines. In *HLT-NAACL*, pages 233–240, 2004.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. A Study of Translation Edit Rate with Targeted Human Annotation. In *Proceedings of association for machine translation in the Americas*, pages 223–231, 2006.

Matthew G. Snover, Nitin Madnani, Bonnie Dorr, and Richard Schwartz. Ter-Plus: paraphrase, semantic, and alignment enhancements to Translation Edit Rate. *Machine Translation*, 23(2-3):117–127, 2009.

Richard Socher, Eric H. Huang, Jeffrey Pennin, Christopher D. Manning, and Andrew Y. Ng. Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection. In *Advances in Neural Information Processing Systems*, pages 801–809, 2011.

Anand Karthik Tumuluru, Chi-kiu Lo, and Dekai Wu. Accuracy and robustness in measuring the lexical similarity of semantic role fillers for automatic semantic mt evaluation. In *Proceedings of the 26th Pacific Asia Conference on Language, Information, and Computation*, 2012.

Stephen Wan, Mark Dras, Robert Dale, and Cécile Paris. Using Dependency-Based Features to Take the Para-farce out of Paraphrase. In *Proceedings of the Australasian Language Technology Workshop*, 2006.

Wei Xu, Alan Ritter, Chris Callison-Burch, William B. Dolan, and Yangfeng Ji. Extracting Lexically Divergent Paraphrases from Twitter. *Transactions Of The Association For Computational Linguistics*, 2:435–448, 2014.

Wei Xu, Chris Callison-Burch, and William B. Dolan. SemEval-2015 Task 1: Paraphrase and semantic similarity in Twitter (PIT). In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval)*, 2015.

Kaizhong Zhang and Dennis Shasha. Simple fast algorithms for the editing distance between trees and related problems. *SIAM journal on computing*, 18(6):1245–1262, 1989.