FBK-HLT: An Effective System for Paraphrase Identification and Semantic Similarity in Twitter

Ngoc Phuoc An Vo
University of Trento,
Fondazione Bruno Kessler
Trento, Italy
ngoc@fbk.eu

Simone Magnolini
University of Brescia,
Fondazione Bruno Kessler
Trento, Italy
magnolini@fbk.eu

Octavian Popescu
IBM Research, T.J. Watson
Yorktown, US
o.popescu@us.ibm.com

Abstract

This paper reports the description and performance of our system, FBK-HLT, participating in the SemEval 2015, Task #1 "Paraphrase and Semantic Similarity in Twitter", for both subtasks. We submitted two runs with different classifiers in combining typical features (lexical similarity, string similarity, word n-grams, etc) with machine translation metrics and edit distance features. We outperform the baseline system and achieve a very competitive result to the best system on the first subtask. Eventually, we are ranked 4$^{\text{th}}$ out of 18 teams participating in subtask "Paraphrase Identification".

1 Introduction

Paraphrase identification/recognition is an important task that can be used as a feature to improve many other NLP tasks as Information Retrieval, Machine Translation Evaluation, Text Summarization, Question and Answering, and others. Besides this, analyzing social data like tweets of social network Twitter is a field of growing interest for different purposes. The interesting combination of these two tasks was brought forward as Shared Task #1 in the SemEval 2015 campaign for "Paraphrase and Semantic Similarity in Twitter" (Xu et al., 2015). In this task, given a set of sentence pairs, which are not necessarily full tweets, their topic and the same sentences with part-of-speech and named entity tags; participating system is required to predict for each pair of sentences is a paraphrase (Subtask 1) and optionally compute a graded score between 0 and 1 for their semantic equivalence (Subtask 2). We participate in this shared task with a system combining different features using a binary classifier. We are interested in finding out whether semantic similarity, textual entailment and machine translation evaluation techniques could increase the accuracy of our system. This paper is organized as follows: Section 2 presents the System Description, Section 3 describes the Experiment Settings, Section 4 reports the Evaluations, Section 5 shows the Error Analysis, and finally Section 6 is the Conclusions and Future Work.

2 System Description

In order to build our system, we extract and select several different linguistic features ranging from simple (word/string similarity, edit distance) to more complex ones (machine translation evaluation metrics), then we consolidate them by a binary classifier. Moreover, different features can be used independently or together with others to measure the semantic similarity and recognize the paraphrase of given sentence pair as well as to evaluate the significance of each feature to the accuracy of system's predictions. On top of this, the system is expandable and scalable for adopting more useful features aiming for improving the accuracy.

2.1 Data Preprocessing

In order to optimizing the system performance, we carefully analyze the given data and notice that Tweets' topic is a sentence part that is always present in both sentences; this redundant similarity in the pairs does not give any information about paraphrase as two sentences can always have a same topic, yet they are may be paraphrase or not. Hence, we remove

Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pages 29–33,
Denver, Colorado, June 4-5, 2015. ©2015 Association for Computational Linguistics

the topic from the sentences, and we did the same in the pairs with Part-of-Speech (POS) and named entity tags. We have not try our system with the topic inside tweets. As being suggested by the guideline of the task, we remove all the pairs with uncertain judgment, such as "debatable" (2, 3). After this data processing, we obtain two smaller datasets with very short texts, sometime reduced to a single word and with very poor syntactic structure. We split the original dataset into two subsets, in which one is composed by sentence pairs and the other one is composed by pairs with POS and named entity tags. Because of the simple structure of given datasets, after undergoing the preprocessing, we decide to focus on exploiting the lexical and string similarity information, rather than syntactic information.

2.2 Lexical and String Similarity

Firstly, for computing the lexical and string similarity between two sentences, we take advantage from the task baseline (Das and Smith, 2009) which is a system using a logistic regression model with eighteen features based on n-grams. This baseline system uses precision, recall and F1-score of 1-gram, 2-grams and 3-grams of tokens and stems from sentence pair to build a binary classification model for identifying paraphrase. We extract these eighteen features from baseline system, without modifications, to use in our classification model.

2.3 Machine Translation Evaluation Metrics

Other than similarity features, we also use evaluation metrics for machine translation as suggested in (Madnani et al., 2012) for paraphrase recognition on Microsoft Research paraphrase corpus (MSRP) (Dolan et al., 2004). In machine translation, the evaluation metric scores the hypotheses by aligning them to one or more reference translations. We take into consideration to use all the eight metrics proposed, but we find that adding some of them without a careful process of training on the dataset may decrease the performance of the system. Thus, we use two metrics for word alignment in our system, the METEOR and BLEU. We actually also take into consideration the metric TERp (Snover et al., 2009), but it does not make any improvement on system performance, hence, we exclude it.

2.3.1 METEOR (Metric for Evaluation of Translation with Explicit ORdering)

We use the latest version of METEOR (Denkowski and Lavie, 2014) that find alignments between sentences based on exact, stem, synonym and paraphrase matches between words and phrases. We used the system as distributed on its website, using only the "norm" option that tokenizes and normalizes punctuation and lowercase as suggested by documentation.[1] We compute the word alignment scores on sentences and on sentences with part-of-speech and named entity tags, as our idea is that if two sentences are similar, their tagged version also should be similar.

2.3.2 BLEU (Bilingual Evaluation Understudy)

We use another metric for machine translation BLEU (Papineni et al., 2002) that is one of the most commonly used and because of that has an high reliability. It is computed as the amount of n-gram overlap, for different values of n=1,2,3, and 4, between the system output and the reference translation, in our case between sentence pairs. The score is tempered by a penalty for translations that might be too short. BLEU relies on exact matching and has no concept of synonymy or paraphrasing.

2.4 Edit Distance

We use the edit distance between sentences as a feature; for that we used the Excitement Open Platform (EOP) (Magnini et al., 2014). To obtain the edit distance, we use EDITS Entailment Decision Algorithm (EDITS EDA), this algorithm classifies the pairs on the base of their edit distance, we take only this one without considering the entailment or not entailment decision. We configure the system to use lemmas and synonyms as identical words to compute sentence distance, the system normalizes the score on the number of token of the shortest sentence. We choose this configuration because it returns the best performance evaluated on training and development data.

2.5 Classification Algorithms

We build two systems for the task with different classifiers, to optimize the Accuracy and F1-score. We use WEKA (Hall et al., 2009) to obtain robust and efficient implementation of the classifiers. We try several classification algorithms in WEKA, among

[1] http://www.cs.cmu.edu/ alavie/METEOR/index.html

Classifier / Features	Baseline features (n-grams)	Baseline +METEOR	Baseline +METEOR +TERp	Baseline +METEOR +BLEU	Baseline +METEOR +BLEU +EditDistance
Baseline (Das and Smith, 2009)	72.4				
EOP EditDistance	73.3				
VotedPerceptron	73.7	75.6	75.5	75.8	**76.2**
MultiLayerPerceptron	73.9	75.6	75.3	75.4	**76.1**

Table 1: Accuracy obtained on development dataset using different classifiers with different features.

others, we find that the VotedPerceptron (with exponent 0.8) and MultilayerPerceptron (with learn rate 0.1; momentum 0.3 and N 10000) return the best performance for the evaluation on training and development data.

3 Experiment Settings

For Subtask 1, we train two models with different feature settings using the VotedPerceptron and MultilayerPerception classification algorithms on the training dataset and we evaluate these models on the development dataset. Finally, we use the same models for the evaluation on the test dataset. In table 1, we report the Accuracy results obtained by using different classifiers with different features. Our chosen classification algorithms outperform the baseline and EOP EditDistance (standalone setting). Table 2 shows F1-score obtained with different classifiers on our best set of features, and our classification algorithms again perform much better the baseline and EOP EditDistance.

For Subtask 2, due to no training data is given for computing the semantic similarity, a different approach is needed. We do not use a classifier, our similarity score is simply the average between METEOR score and edit distance score.

Classifier	F1
Baseline (Das and Smith, 2009)	.502
EOP EditDistance	.609
VotedPerceptron	**.746**
MultiLayerPerceptron	**.741**

Table 2: F1-score obtained using different classifiers on the best set of features (baseline + METEOR + BLEU + EditDistance).

	Subtask1			Subtask2
Team	Prec	Rec	F1	Pearson
Baseline$^{\text{(logistic reg)}}$	.679	.520	.589	.511
Baseline$^{\text{(WTMF)}}$	.450	.663	.536	.350
Baseline$^{\text{(random)}}$	.192	.434	.266	.017
ASOBEK$^{(1^{\text{st}}\text{ Subtask1})}$	.680	.669	.674	.475
MITRE$^{(1^{\text{st}}\text{ Subtask2})}$	.569	.806	.667	.619
FBK-HLT$^{\text{(voted)}}$	**.685**	.634	.659	.462
FBK-HLT$^{\text{(multilayer)}}$	.676	.549	.606	.480

Table 3: Paraphrase and Semantic Similarity Results.

4 Evaluations

We submit two runs using two models described in the Section 3 for both subtasks. In the Table 3, we report the performance of our two runs against the baselines and best systems in each subtask. In Subtask 1, our runs outperform all three baselines and achieve very competitive results to the best system ASOBEK. In the run FBK-HLT$^{\text{(voted)}}$, we even achieve a better Precision than the best system. In Subtask 2, though we apply a simple computation method for semantic similarity by averaging the word alignment score and EditDistance, we still have better results than two our of three baselines.

5 Error Analysis

In this section, we conduct an analysis of the misclassifications that our best system, FBK-HLT$^{\text{(voted)}}$, makes on test dataset. We extract and show some randomly selected examples in which our system classifies incorrectly, both false positive or false negative; and then we analyze the possible causes for the misclassification. This inspection yields not only the top sources of error for our approach but also uncovers sources of unclear annotations in dataset.

True Positive	True Negative	False Positive	False Negative
111	612	51	64

Table 4: Error Analysis.

5.1 False positive

[1357] *omg Family Guy is killing me right now - OMG we were quoting family guy*
[1357] *family guy is trending in the US - Family guy is so racist or maybe they just point out the racism in America*
[4135] *hahaha that sounds like me - That sounds totally reasonable to me*
[5211] *The world of jenks is such a real show - Jenks from the World of Jenks is such a good person*
[128] *Anyone trying to see After Earth sometime soon - Me and my son went to see After Earth last night*

Though all these sentence pairs share many word similarity/matching and alignments, they are annotated as non-paraphrase. For example, the sentence pair **[4135]** has very high word matching and alignment after removing the common topic "sounds", but the important words "like" and "reasonable" which differ the meaning between two sentences, are not really semantically captured and distinguished by our system. As our system does not use any semantic feature, this kind of semantic difference is difficult to distinguish,leading to false positive case.

5.2 False negative

[4220] *Hell yeah Star Wars is on - Star Wars and lord of the rings on tv*
[785] *Chris Davis is putting the team on his back - Chris Davis doing what he does*
[400] *Rafa Benitez deserves a hell of a thank you - Any praise for Benitez from my Chelsea followers*
[2832] *Classy gesture by the Mets for Mariano - real class shown by The Mets Mo Rivera is a legend*
[4062] *Shonda is a freaking genius - THAT LADY IS AMAZING I LOVE SHONDA*

This case is opposite to the previous case, even though these sentence pairs do not share many word similarity and alignment, they are annotated as paraphrase. We can possibly propose some hypothesis as follows:

Extra information Though the pairs **[4220]** and **[400]** may not be paraphrase according to the paraphrase definition in the literature (Bhagat and Hovy, 2013), they are annotated as paraphrase in the gold-standard labels. We notice that as one sentence contains more extra information than the other one, it leads to low word similarity and alignment, which makes our system make wrong classification.
Specific knowledge-base In this case, the pairs **[785]** and **[2832]** require a specific knowledge-base, which is about baseball, to recognize the paraphrase; hence, even for human without any related knowledge, it might be difficult detect the paraphrase.
Common sense Though both sentences of the pair **[4062]** do not share any word similarity/alignment, they have a positive polarity that may allow identifying the paraphrase. This case may be easy for human to identify the paraphrase, yet it is difficult for machine to capture the same perception.

Table 4 shows that we can improve our system performance by reducing the false positive and false negative. In other word, we need to exploit more semantic features to make correct classification. However, according to our analysis for the false negative, it is difficult to cover these cases.

6 Conclusions and Future Work

In this paper, we describe a system participating in the SemEval 2015, Task #1 "Paraphrase and Semantic Similarity in Twitter", for both subtasks. We present a supervised system which considers multiple features at low level, such as lexical, string similarities, word alignment and edit distance. The performance of our runs is much better than the baselines and very competitive to the best system; we are ranked 4th of total 18 teams in Subtask 1.
A lower result was obtained in Subtask 2, as the chosen features have not really acquired the semantic similarity judgment. Hence, we expect to study more useful features (e.g the POS information, semantic word similarity) to improve our system performance on both identifying paraphrase and computing semantic similarity scores.

References

Rahul Bhagat and Eduard Hovy. 2013. What is a paraphrase? *Computational Linguistics*, 39(3):463–472.

Dipanjan Das and Noah A Smith. 2009. 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-Volume 1*, pages 468–476.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the EACL 2014 Workshop on Statistical Machine Translation*.

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *Proceedings of the 20th international conference on Computational Linguistics*, page 350.

Mark Hall, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H Witten. 2009. The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1):10–18.

Nitin Madnani, Joel Tetreault, and Martin Chodorow. 2012. 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.

Bernardo Magnini, Roberto Zanoli, Ido Dagan, Kathrin Eichler, Günter Neumann, Tae-Gil Noh, Sebastian Pado, Asher Stern, and Omer Levy. 2014. The excitement open platform for textual inferences. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (ACL 2014)*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318.

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

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