MAZA at SemEval-2016 Task 11: Detecting Lexical Complexity Using a Decision Stump Meta-Classifier

Shervin Malmasi
Macquarie University
Sydney, NSW, Australia
shervin.malmasi@mq.edu.au

Marcos Zampieri
Saarland Unviersity
Saarbrücken, Germany
marcos.zampieri@dfki.de

Abstract

This paper describes team *MAZA* entries for the 2016 SemEval Task 11: Complex Word Identification (CWI). The task is a binary classification task in which systems are trained to predict whether a word in a sentence is considered to be complex or not. We developed our two systems for this task based on classifier stacking using decision stumps and decision trees. Our best system, using contextual features, frequency information, and word and sentence length, achieved 91.2% accuracy and 30.8% F-Score. The system ranked 4^{th} among the 38 entries in the CWI task in terms of F-Score.

1 Introduction

Lexical simplification is a popular task in natural language processing and it was the topic of a successful SemEval task in 2012 (Specia et al., 2012). It consists of applying computational methods to substitute words or short phrases for simpler ones to improve text readability and comprehension aimed at a given target population (e.g. children, language learners, people with reading impairment, etc.). Lexical simplification is considered to be the sub-task of text simplification that deals with the lexicon while other sub-tasks address, for example, complex syntactic structures (Siddharthan, 2014).

To perform lexical simplification efficiently, computational methods should be first applied to identify which words in a text pose more difficulty to readers and they therefore good candidates for substitution (Shardlow, 2013). This task is called complex word identification (CWI) and it is the topic of the 2016 SemEval Task 11 with the same name.

The CWI shared task is modeled as a binary text classification task. Participants are provided with training data containing sentences and a label for each word in them containing a value of either 1 (for complex words) or 0 (for simple words). The label was attributed according to the judgment of human annotators that were required to indicate which words in the sentences could not be easily understood. Below, an example can be found of a sentence from the training set. Complex words are marked in bold.

(1) The name 'kangaroo mouse' refers to the species' **extraordinary** jumping ability, as well as its habit of **bipedal locomotion**.

In the example presented above, the CWI systems should label *extraordinary, bipedal* and *locomotion* as complex words.[1] To accomplish this task, the *MAZA* team applied a decision stump meta-classifier and a wide set of features that we will describe here.

2 Data

Organizers of the SemEval CWI task provided a training and test set comprising English sentences with each word annotated with a complex or simple label. According to the CWI task website[2]: '400 annotators were presented with several sentences and asked to select which words they did not understand

[1] Note that participants are free to consider *bipedal locomotion* as single words or as a multiword expression.

[2] http://alt.qcri.org/semeval2016/task11/

Proceedings of SemEval-2016, pages 916–920,
San Diego, California, June 16-17, 2016.

their meaning'. There was no scale or gradation, all words should be assigned as simple or complex.

The training set was composed of 2,237 sentences. It contains judgments made by 20 annotators over a set of 200 sentences. A word is considered to be complex if at least one of the 20 annotators assigned them as complex. Subsequently a test set with the same format was released containing 88,221 sentences. According to the organizers, the test set contains by judgments made over 9,000 sentences by a single annotator.

The proportion of training vs. test instances of 1:40 should also be noted as it represents and additional challenge to participants. This data split is different from other similar text classification shared tasks which provide much more training than test instances (at least 10:1) (Tetreault et al., 2013; Zampieri et al., 2015). Given the amount of training data, participating teams should employ efficient algorithms able to perform generalizations on a much larger test set.[3]

3 Features

We experimented with two types of features in our submissions. Each of these two classes, as described below, contains several features which we combine using a meta-classifier.

3.1 Frequency and Length Features

These are features based on the occurrence of the target word in a given reference corpus and its length. The idea is inspired by the Zipfian frequency distribution of words that indicate that the most frequent words in any language tend to be shorter (e.g. in English some of the most frequent words are: *it, the, an, and*). If we consider that frequent words are also likely to be short, our assumption is that complex words are likely to be, on average, both less frequent and longer than simple ones (Zipf, 1949). This assumption is also related to text readability and it has been tested in an experiment with dyslexic readers concluding that frequent words tend to improve readability while shorter words help text comprehension (Rello et al., 2013).

The reference corpus we used was the English section of the DSL corpus collection (DSLCC) (Tan et al., 2014). This corpus seems to be an appropriate choice for our task as it was designed for language variety discrimination. For this reason, it contains English texts from both England and the United States. This ensures a desired variability in terms of spelling and word combination between the two most representative English varieties.

The frequency and length features we use are:

- **Word Probability**: The probability of the word occurring in the reference corpus.
- **Word Length**: The number of characters in the word. Our aforementioned intuition is that longer words tend to be both less frequent and more complicated to readers (Zipf, 1935; Zipf, 1949).
- **Sentence Length**: The number of characters in the sentence to which the target word belongs.

3.2 Context Features

This set of features is based on estimating the likelihood of the target word within its context in a sentence. For a given target word w_i, we calculate six different types of probability, as described here.

The probabilities were extracted using the Microsoft Web N-gram service[4] which is based on web-scale data.

- **Conditional Probabilities:** We estimate the conditional probability of w_i, given its preceding context. Two probabilities are calculated: the probability of w_i given the previous word and the probability of w_i given the previous two words.
- **Joint Probabilities** Additionally, we also extract the joint probability of w_0 and its surrounding words. We derived such joint probabilities for $\{w_{i-2}, w_{i-1}, w_i\}$, $\{w_{i-1}, w_i\}$, $\{w_i, w_{i+1}\}$ and $\{w_i, w_{i+1}, w_{i+2}\}$.

[3] As noted by Zampieri and Tan (2014) in the Chinese Error Correction Shared Task (Yu et al., 2014)

[4] http://research.microsoft.com/en-us/collaboration/focus/cs/web-ngram.aspx

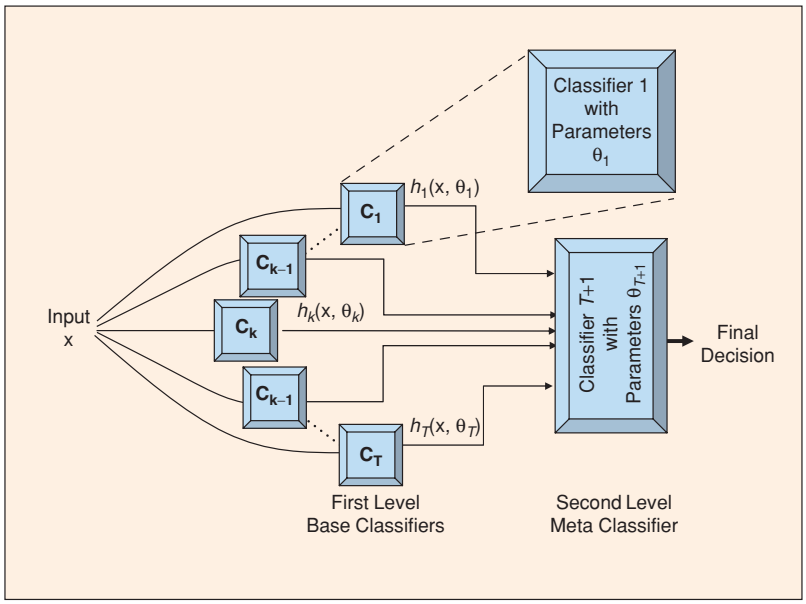

Figure 1: An illustration of a meta-classifier architecture. Image reproduced from Polikar (2006).

4 Experimental Setup

We employed a meta-classifier for our entry, also referred to as classifier stacking. A meta-classifier architecture is generally composed of an ensemble of base classifiers that each make predictions for all of the input data. Their individual predictions, along with the gold labels are used to train a second-level meta-classifier that learns to predict the final decision for an input, given the decisions of the individual classifiers.

This setup is illustrated in Figure 1. This meta-classifier attempts to learn from the collective knowledge represented by the ensemble of local classifiers. The first step in such a setup is to create the set of base classifiers that form the first layer of the architecture. We describe this process below.

4.1 Ensemble Construction

Our ensemble was created using a set of decision stump classifiers. A decision stump is a decision tree trained using only a single feature (Iba and Langley, 1992); it is usually considered a weak learner.

We used the features listed in Section 3 to create an ensemble of 9 classifiers. Each classifier predicts every input and assigns a probability output to each of the two possible labels.

Classifiers ensembles have proved to an efficient and robust alternative in other text classification tasks such as language identification (Malmasi and Dras, 2015a) and grammatical error detection (Xiang et al., 2015). This motivated us to try this approach in the CWI SemEval task.

4.2 Meta-classifier

For our meta-classifier, we adopted a decision tree with bootstrap aggregating (bagging). The inputs to each decision tree are the two probability outputs from each decision stump in our ensemble, along with the original gold label. 200 bagged decision trees were created using this input. The final label was selected through a plurality voting process over the entire set of bagged decision trees.

4.3 Systems

Using the methods described so far, we created two different systems for the shared task. They are summarized next:

- **MAZA A:** Our first system used only the frequency and length features described in Section 3.1.
- **MAZA B:** The second system we created combined the frequency and length features used in *MAZA A* with the addition of the contextual features we described in Section 3.2.

Rank	Team	System	Accuracy	Precision	Recall	F-score	G-score
1	PLUJAGH	SEWDFF	0.922	0.289	0.453	0.353	0.608
2	LTG	System2	0.889	0.220	0.541	0.312	0.672
3	LTG	System1	0.933	0.300	0.321	0.310	0.478
4	**MAZA**	B	0.912	0.243	0.420	0.308	0.575
5	HMC	DecisionTree25	0.846	0.189	0.698	0.298	0.765
6	TALN	RandomForest_SIM.output	0.847	0.186	0.673	0.292	0.750
7	HMC	RegressionTree05	0.838	0.182	0.705	0.290	0.766
8	MACSAAR	RFC	0.825	0.168	0.694	0.270	0.754
9	TALN	RandomForest_WEI.output	0.812	0.164	0.736	0.268	0.772
10	UWB	All	0.803	0.157	0.734	0.258	0.767
11	PLUJAGH	SEWDF	0.795	0.152	0.741	0.252	0.767
12	JUNLP	RandomForest	0.795	0.151	0.730	0.250	0.761
13	SV000gg	Soft	0.779	0.147	0.769	0.246	0.774
14	MACSAAR	NNC	0.804	0.146	0.660	0.240	0.725
15	JUNLP	NaiveBayes	0.767	0.139	0.767	0.236	0.767
16	SV000gg	Hard	0.761	0.138	0.787	0.235	0.773
17	USAAR	entropy	0.869	0.148	0.376	0.212	0.525
18	**MAZA**	A	0.773	0.115	0.578	0.192	0.661
19	BHASHA	DECISIONTREE	0.836	0.118	0.387	0.181	0.529
20	BHASHA	SVM	0.844	0.119	0.363	0.179	0.508

Table 1: The top 20 systems submitted to the shared task, ranked by their F-score.

We expected the system *B* to perform better, but we were interested in quantifying the impact of the contextual features on the test set results by the comparing the two systems.

5 Results

We present in Table 1 the best 20 out of 38 systems ranked by their F-score. We present the results obtained in terms of Accuracy, Recall, Precision, F-Score, and G-Score.[5] The complete results and more information about the evaluation can be found in the CWI shared task report paper (Paetzold and Specia, 2015).

As expected, our second system, *MAZA B* that incorporated contextual features along with frequency and length features performed better, ranking in 4th place overall. Our first system, *MAZA A* obtained performance more than 11 percentage points worse than the *B* system, coming in 18th place.

Our results show that the contextual features we applied in the *MAZA B* submission are very informative for this task. This suggests that the complexity of a word is strongly tied to the context in which it is being used and it cannot be solely determined by how frequent or how long the word is.

[5]According to the organizers, the G-score is the harmonic mean between Accuracy and Recall.

6 Conclusion

In this paper we described our systems for SemEval 2016 Task 11: Complex Word Identification (CWI). Our best system, *MAZA B* was ranked 4th in terms of F-Score among 38 entries in the shared task. We consider the results we obtained to be very positive given the amount of teams participating in the task.

We applied a meta-classifier approach where each target word is classified by several base classifiers, and another classifier learns to predict the final label using the outputs of those classifiers. Our system's competitive performance in task suggests that this is a promising approach for this task.

Future work could look at how additional language resources could be used for this task. Analyzing the language produced by learners could provide insight into the limitations of learners' vocabulary. Learner corpora, widely used in the task of Native Language Identification (Malmasi and Dras, 2014; Malmasi and Dras, 2015b) could be useful here.

Acknowledgments

We would like to thank the SemEval CWI organizers, Gustavo Paetzold and Lucia Specia, for organizing this event. We also thank the anonymous reviewers for their constructive comments.

References

Wayne Iba and Pat Langley. 1992. Induction of one-level decision trees. In *Proceedings of the ninth international conference on machine learning*, pages 233–240.

Shervin Malmasi and Mark Dras. 2014. Chinese Native Language Identification. In *Proceedings of EACL*.

Shervin Malmasi and Mark Dras. 2015a. Language identification using classifier ensembles. In *Proceedings of the LT4VarDial Workshop*.

Shervin Malmasi and Mark Dras. 2015b. Multilingual Native Language Identification. In *Natural Language Engineering*.

Gustavo Henrique Paetzold and Lucia Specia. 2015. Semeval 2016 task 11: Complex word identification. In *Proceedings of SemEval*.

Robi Polikar. 2006. Ensemble based systems in decision making. *Circuits and Systems Magazine, IEEE*, 6(3):21–45.

Luz Rello, Ricardo Baeza-Yates, Laura Dempere-Marco, and Horacio Saggion. 2013. Frequent words improve readability and short words improve understandability for people with dyslexia. In *Proceedings of INTERACT*.

Matthew Shardlow. 2013. A comparison of techniques to automatically identify complex words. In *Proceedings of the ACL Student Research Workshop*.

Advaith Siddharthan. 2014. A survey of research on text simplification. *ITL-International Journal of Applied Linguistics*, 165(2):259–298.

Lucia Specia, Sujay Kumar Jauhar, and Rada Mihalcea. 2012. Semeval-2012 task 1: English lexical simplification. In *Proceedings of SemEval*.

Liling Tan, Marcos Zampieri, Nikola Ljubešic, and Jörg Tiedemann. 2014. Merging comparable data sources for the discrimination of similar languages: The dsl corpus collection. In *Proceedings of the BUCC workshop*.

Joel Tetreault, Daniel Blanchard, and Aoife Cahill. 2013. A report on the first native language identification shared task. In *Proceedings of the BEA Workshop*.

Yang Xiang, Xiaolong Wang, Wenying Han, and Qinghua Hong. 2015. Chinese grammatical error diagnosis using ensemble learning. In *Proceedings of the NLP-TEA Workshop*.

Liang-Chih Yu, Lung-Hao Lee, and Li-Ping Chang. 2014. Overview of grammatical error diagnosis for learning chinese as a foreign language. In *Proceedings of ICCE*.

Marcos Zampieri and Liling Tan. 2014. Grammatical error detection with limited training data: The case of chinese. In *Proceedings of ICCE*.

Marcos Zampieri, Liling Tan, Nikola Ljubešic, Jörg Tiedemann, and Preslav Nakov. 2015. Overview of the DSL shared task 2015. In *Proceedings of the LT4VarDial Workshop*.

George Kingsley Zipf. 1935. *The Psychobiology of Language*. Houghton Mifflin.

George Kingsley Zipf. 1949. *Human behavior and the principle of least effort.* Addison-wesley Press.