Quality Assessment for Text Simplification (QATS)

Workshop Programme

Saturday, May 28, 2016

09:00 – 09:20	**Introduction** by Sanja Štajner
09:20 – 10:00	**Invited Talk** by Advaith Siddharthan

Session: General Track

10:00 – 10:30	Gustavo H. Paetzold and Lucia Specia *PLUMBErr: An Automatic Error Identification Framework for Lexical Simplification*
10:30 – 11:00	Coffee break
11:00 – 11:30	Sandeep Mathias and Pushpak Bhattacharyya *How Hard Can it Be? The E-Score - A Scoring Metric to Assess the Complexity of Text*
11:30 – 12:00	Sanja Štajner, Maja Popović and Hanna Béchara *Quality Estimation for Text Simplification*

12:00 – 12:15	**Shared Task: Introduction** by Maja Popović

Session: Shared Task 1

12:15 - 12:45	Maja Popović and Sanja Štajner *Machine Translation Evaluation Metrics for Quality Assessment of Automatically Simplified Sentences*
12:45 - 13:15	Sandeep Mathias and Pushpak Bhattacharyya *Using Machine Translation Evaluation Techniques to Evaluate Text Simplification Systems*
13:15 - 14:30	Lunch break

Session: Shared Task 2

14:30 – 15:00	Gustavo H. Paetzold and Lucia Specia *SimpleNets: Evaluating Simplifiers with Resource-Light Neural Networks*
15:00 – 15:30	Sergiu Nisioi and Fabrice Nauze *An Ensemble Method for Quality Assessment of Text Simplification*
15:30 – 16:00	Elnaz Davoodi and Leila Kosseim *CLaC @ QATS: Quality Assessment for Text Simplification*
16:00 – 16:30	Coffee break
16:30 – 17:30	**Round Table**
17:30 – 17:45	**Closing**

Table of Contents

Author Index

CLaC @ QATS: Quality Assessment for Text Simplification

Elnaz Davoodi and Leila Kosseim
Concordia University
Montreal, Quebec, Canada
e_davoo@encs.concordia.ca, kosseim@encs.concordia.ca

Abstract

This paper describes our approach to the 2016 QATS quality assessment shared task. We trained three independent Random Forest classifiers in order to assess the quality of the simplified texts in terms of grammaticality, meaning preservation and simplicity. We used the language model of Google-Ngram as feature to predict the grammaticality. Meaning preservation is predicted using two complementary approaches based on word embedding and WordNet synonyms. A wider range of features including TF-IDF, sentence length and frequency of cue phrases are used to evaluate the simplicity aspect. Overall, the accuracy of the system ranges from 33.33% for the overall aspect to 58.73% for grammaticality.

Keywords: Simplification, Word Embedding, Language Model

1. Introduction

Automatic text simplification is the process of reducing the complexity of a text to make it more accessible to a broader range of readers with different readability levels. While preserving its meaning as much as possible, a text's lexical, syntactic and discourse level features should be made more simple. However, evaluating the simplicity level of a text is still a challenging task for both humans and automatic systems.

Current approaches to automate text simplification vary depending on the type of simplification. Lexical simplification was the first effort in this area. In particular, Devlin and Tait (1998) introduced an approach of replacing words with their most common synonym based on frequency (Kučera et al., 1967). More recently, publicly available resources such as Simple English Wikipedia [1] and the Google 1T corpus [2] have been used to automate lexical simplification based on similar approaches such as common synonym replacement and context vectors (e.g. (Biran et al., 2011; Bott et al., 2012; Rello et al., 2013; Kauchak, 2013)).

Another approach to automatic text simplification involves syntactic simplification. Current work in this area aims to identify and simplify complex syntactic constructions such as passive phrases, embedded clauses, long sentences, etc. Initial work on syntactic simplification focused on the use of transformation rules in order to generate simpler sentences (e.g. Chandrasekar and Srinivas (1997)). Later, work have paid more attention on sentence splitting (e.g. Carroll et al. (1998)), rearranging clauses (e.g. Siddharthan (2006)) and dropping clauses (e.g. (Barlacchi and Tonelli, 2013; Štajner et al., 2013)). To our knowledge, Siddharthan (2003) is the only effort that specifically addressed the preservation of a text's discourse structure by resolving anaphora and ordering sentence.

In the remainder of this paper, we describe the methodology we used to measure the 4 simplification criteria of the QATS workshop: GRAMMATICALITY, MEANING PRESERVATION, SIMPLICITY and OVERALL. In Sections 2 and 3, the details of our submitted system are described, while Section 4 summarises our results.

2. System Overview

As can be seen in Figure 1, our system consisted of three independent supervised models in order to predict each of the three main aspects: GRAMMATICALITY, MEANING PRESERVATION and SIMPLICITY. We used 10 fold cross-validation in order to choose the best supervised models. The 4^{th} aspect (i.e. OVERALL) was predicted using the predictions of MEANING PRESERVATION and SIMPLICITY.

2.1. Grammaticality Prediction

In order to predict the quality of the simplified sentences from the point of view of grammaticality, we have used the log likelihood score of the sentences using the Google Ngram corpus[3]. To do this, the BerkeleyLM language modeling toolkit[4] was used (Pauls and Klein, 2011) to built a language model from the Google Ngram corpus, then the perplexity of all simple sentences in the training set were calculated. These log likelihood scores were used as features to feed a Random Forest classifier.

2.2. Meaning Preservation Prediction

The purpose of meaning preservation is to evaluate how close the meaning of the original sentence is with respect to its simple counterpart. To do this, we used two complementary approaches based on word embedding and the cosine measure.

2.2.1. Word Embedding

We used the Word2Vec package (Mikolov et al., 2013a; Mikolov et al., 2013b) to learn the representation of words on the Wikipedia dump[5]. We then trained a skip-gram model using the *deeplearing4j*[6] library. As a result, each

[1] http://www.cs.pomona.edu/~dkauchak/simplification/

[2] https://books.google.com/ngrams

[3] https://books.google.com/ngrams

[4] http://code.google.com/p/berkeleylm/

[5] http://www.cs.pomona.edu/~dkauchak/simplification/

[6] http://deeplearning4j.org/

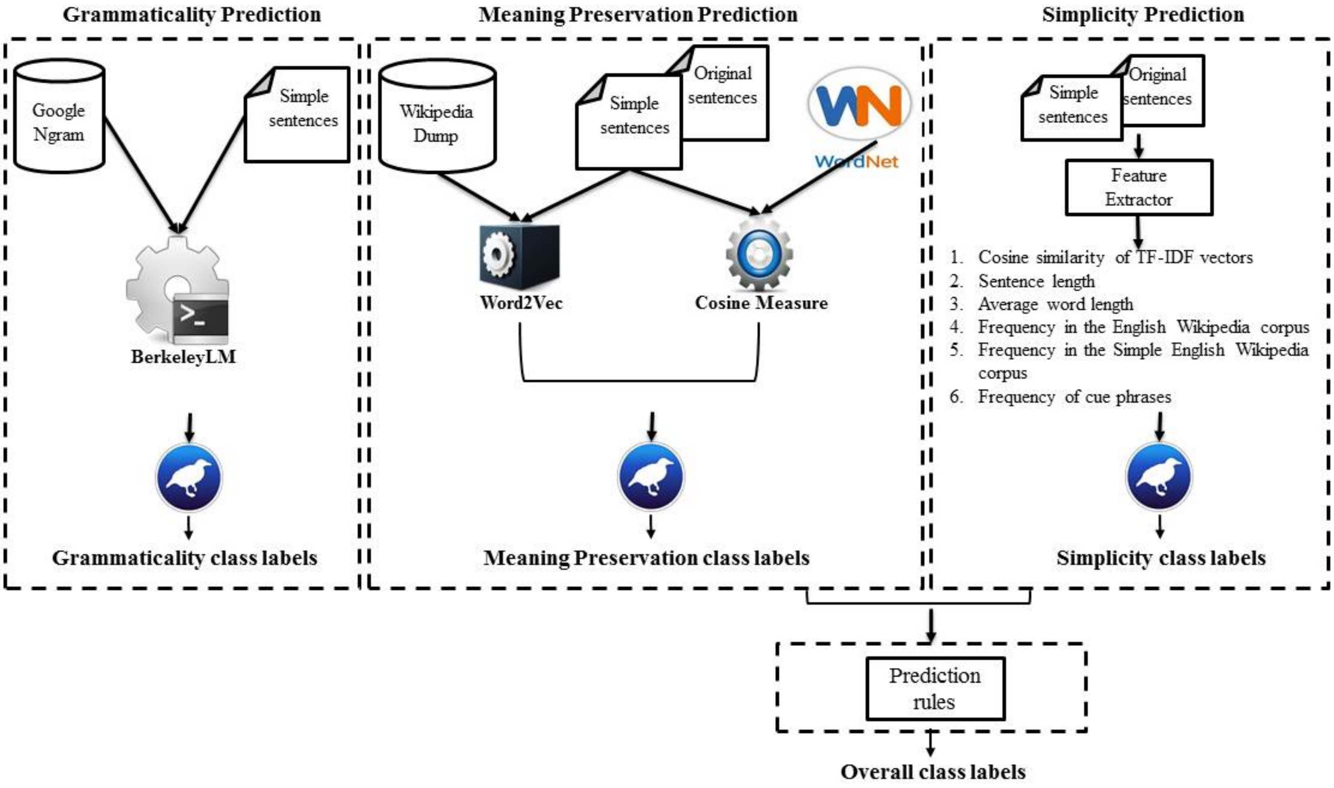

Figure 1: System Overview

word in the original sentence and its simple counterpart are represented as a vector. As calculating the similarity of two sentences using word embedding is still a challenging task, our approach to this problem was to use average similarity. To do so, we calculated the similarity of each word (each vector) in the original sentence to all the words in its simpler counterpart. Then using the average length of the original and simple pairs, we calculated the average similarity between a pair of sentences. This similarity was the first feature we fed to a Random Forest classifier.

2.2.2. Cosine Similarity of WordNet Synonyms

The second feature we used for meaning preservation was based on WordNet synonyms. Each sentence was represented as a vector of its constituent words. Then, using WordNet[7], all synonyms of each word were added to the corresponding vector of the sentence. However, as each word can have various part of speech (POS) tags, before expanding the vector, we first identified the POS of all the words in the sentence using the Stanford POS tagger (Manning et al., 2014). Afterwards, we filtered the synonyms according to the POS tags and added only those with the same POS tag of the word. As a result, each sentence was represented as a vector of words and their synonyms. Using the cosine similarity to calculate the similarity between corresponding vectors of pairs of sentences, we measured how close the meaning of two sentences were.

$$Cosine\_Sim(i) = \frac{\overrightarrow{O_i}.\overrightarrow{S_i}}{||\overrightarrow{O_i}|| \times ||\overrightarrow{S_i}||}$$

2.3. Simplicity Prediction

The purpose of simplicity prediction is to evaluate how simpler the simple sentences are compared to their original counterpart. As simplicity can be measured at various levels (i.e. lexical, syntactic and discourse), we considered the following sets of features in order to capture the changes at each level.

2.3.1. Vector Space Model Similarity

The first feature we considered in order to evaluate the simplicity of the simple sentences compared to their original counterpart, was the cosine similarity between the Term Frequecy-Inverse Document Frequency (TF-IDF) vectors of each pair. A cosine similarity of 1 indicates that no change has been made in the simplification process. However, before transforming sentences into their corresponding TF-IDF vector, we preprocessed them. First, stop words were removed, then all words were stemmed using the Porter Stemmer (Porter, 1980). As a result, each sentence was represented as a vector of the size of all the stems in all sentences. It is worth noting that in order to compute the inverse document frequency for each stem, we considered each sentence as a document. The cosine similarity between original and simple sentences of the i^{th} pair is calculated using Formula 1, where $\overrightarrow{O_i}$ and $\overrightarrow{S_i}$ represent the vectors of the original sentence and its simple counterpart correspondingly.

[7] https://wordnet.princeton.edu/

2.3.2. Sentence Length

Traditional approaches to readability level assessment have identified text length as an important feature to measure complexity (e.g. Kincaid et al. (1975)). Following this, we investigated the influence of sentence length in terms of the number of open class words only. By ignoring closed class words, we eliminated the effect of words which do not contribute much to the meaning of the sentence. Thus, we considered the difference between the length of pairs of sentences as our second feature for simplicity prediction.

2.3.3. Average Word Length

According to Kincaid et al. (1975), not only can the number of words in the sentence be an indicator of simplicity level, but also its length in terms of the number of characters. To account for this, we also considered the difference between the average number of characters between pairs of sentences. Using this feature along with the number of words of each sentence (see Section 2.3.2), we investigated not only the influence of the length of sentence, but also the length of each word in the sentence.

2.3.4. Frequency in the English Wikipedia Corpus

The frequency of each word in the regular English Wikipedia can be an indicator of the simplicity level of the word. We expected that words in the original sentences would be more frequent in the regular English Wikipedia than words of the simple sentences. Thus, we calculated the difference between the average frequency of all words of the original sentence and their simple counterpart. To do this, we preprocessed both pairs of sentences and the regular English Wikipedia corpus, in order to remove stop words and then stem the remaining words.

2.3.5. Frequency in the Simple English Wikipedia Corpus

The Simple English Wikipedia corpus[8] is an aligned corpus of 60K *¡regular, simple¿* pairs of Wikipedia articles. We used this corpus in order to calculate the average frequency of words of each pair of sentences. We expected the words of simpler sentences to be more frequent in the Simple Wikipedia articles compared to the original sentences. To do this, we performed the same preprocessing as described in Section 2.3.4. and used the average frequency of the sentence's stems as features.

2.3.6. Frequency of Cue Phrases

The last feature we considered to predict the simplicity aspect was the difference in the usage of cue phrases. Cue phrases are special terms such as *however, because, since, etc.* which connect text segments and mark their discourse purpose. Several inventories of cue phrases have been proposed (e.g. (Knott, 1996; Prasad et al., 2007)). For our work, we used the list of 100 cue phrased introduced by Prasad et al. (2007) and calculated the difference between the frequency of cue phrases across pairs of sentences. It is worth noting that cue phrases may be used to explicitly signal discourse relations between text segments or may be used in a non-discourse context. However, here we considered both discourse and non-discourse usage of cue phrases.

2.4. Overall Prediction

The last aspect to be predicted evaluated the combination of all other aspects. According to our analysis of the training data set, this aspect depended mostly on the SIMPLICITY and the MEANING PRESERVATION aspects. Our prediction of this aspect was based only on a simple set of rules using the predictions of these two aspects. The following shows the rules we used to predict the value of this aspect.

- If **both** *simplicity* and *meaning preservation* are classified as GOOD, then *overall* = GOOD,
- If **at least** one of *simplicity* or *meaning preservation* is classified as BAD, then *overall* = BAD,
- otherwise, *overall* = OK.

3. Data and Results

The training set contains 505 pairs of original and simple sentences. The original sentences were taken from the news domain and from Wikipedia and the simple counterparts were automatically simplified using various text simplification systems. Thus, the simple counterparts may contain various types of simplifications such as lexical, syntactic or mixture of both. Table 1 shows the distribution of the data for each of the four aspects. As can be seen, none of the aspects have a normal distribution over the class-labels.

Value(%) Aspect	Good	Ok	Bad
Grammaticality	75.65	14.26	10.09
Meaning preservation	58.22	26.33	15.45
Simplicity	52.68	30.29	17.03
Overall	26.33	46.14	27.53

Table 1: Distribution of data

For our participation, we submitted one run for GRAMMATICALITY and MEANING PRESERVATION and three runs for the SIMPLICITY and OVERALL aspects. The three runs had different classification threshold to assign class labels. Our official results are listed in Table 2. MAE and RMSE stand for Mean Average Error and Root Mean Square Error correspondingly.

4. Bibliographical References

Barlacchi, G. and Tonelli, S. (2013). Ernesta: A sentence simplification tool for children's stories in Italian. In *Computational Linguistics and Intelligent Text Processing (CICLing-2013)*, pages 476–487.

Biran, O., Brody, S., and Elhadad, N. (2011). Putting it simply: A context-aware approach to lexical simplification. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2*, pages 496–501.

[8] http://www.cs.pomona.edu/~dkauchak/simplification/

System Name	Accuracy	MAE	RMSE
Grammaticality-Davoodi-RF-perplexity	58.73%	27.38	34.66
Meaning preservation-Davoodi-RF	49.21%	30.56	36.31
Simplicity-Davoodi-RF-0.5	34.92%	43.25	45.32
Simplicity-Davoodi-RF-0.6	35.71%	41.67	44.48
Simplicity-Davoodi-RF-0.7	35.71%	40.48	44.48
Overall-Davoodi-0.5	34.13%	41.27	44.21
Overall-Davoodi-0.6	32.54%	42.06	45.67
Overall-Davoodi-0.7	33.33%	41.27	45.16

Table 2: Official Results of our system at QATS.

Bott, S., Rello, L., Drndarevic, B., and Saggion, H. (2012). Can Spanish be simpler? LexSiS: Lexical simplification for spanish. In *Coling*, pages 357–374.

Carroll, J., Minnen, G., Canning, Y., Devlin, S., and Tait, J. (1998). Practical simplification of english newspaper text to assist aphasic readers. In *Proceedings of the AAAI-98 Workshop on Integrating Artificial Intelligence and Assistive Technology*, pages 7–10.

Chandrasekar, R. and Srinivas, B. (1997). Automatic induction of rules for text simplification. *Knowledge-Based Systems*, 10(3):183–190.

Devlin, S. and Tait, J. (1998). The use of a psycholinguistic database in the simplification of text for aphasic readers. *Linguistic Databases*, pages 161–173.

Kauchak, D. (2013). Improving text simplification language modeling using unsimplified text data. In *Proceeding of ACL (Volume 1: Long Papers)*, pages 1537–1546.

Kincaid, J. P., Fishburne, J., Robert, P., Rogers, R. L., and Chissom, B. S. (1975). Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. Technical report, DTIC Document.

Knott, A. (1996). *A data-driven methodology for motivating a set of coherence relations*. Ph.D. thesis, The University of Edinburgh: College of Science and Engineering: The School of Informatics.

Kučera, H., Francis, W. N., et al. (1967). Computational analysis of present-day American English. Technical report, Brown University Press.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J. R., Bethard, S., and McClosky, D. (2014). The Stanford CoreNLP Natural Language Processing Toolkit. In *Proceedings of 52^{nd} Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS-2013)*, pages 3111–3119.

Pauls, A. and Klein, D. (2011). Faster and smaller n-gram language models. In *Proceedings of the 49^{th} Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 258–267.

Porter, M. F. (1980). An algorithm for suffix stripping. *Program*, 14(3):130–137.

Prasad, R., Miltsakaki, E., Dinesh, N., Lee, A., Joshi, A., Robaldo, L., and Webber, B. L. (2007). The Penn Discourse Treebank 2.0 annotation manual. `https://www.seas.upenn.edu/~pdtb/`.

Rello, L., Baeza-Yates, R., Dempere-Marco, L., and Saggion, H. (2013). Frequent words improve readability and short words improve understandability for people with dyslexia. In *Human-Computer Interaction–INTERACT 2013*, pages 203–219.

Siddharthan, A. (2003). Preserving discourse structure when simplifying text. In *Proceedings of the European Natural Language Generation Workshop (ENLG), 11^{th} Conference of the European Chapter of the Association for Computational Linguistics (EACL'03)*, pages 103–110.

Siddharthan, A. (2006). Syntactic simplification and text cohesion. *Research on Language & Computation*, 4(1):77–109.

Štajner, S., Drndarevic, B., and Saggion, H. (2013). Corpus-based sentence deletion and split decisions for Spanish text simplification. *Computación y Sistemas*, 17(2):251–262.