Automatic Text Simplification for People with Intellectual Disabilities

Ping Chen
Department of Engineering
University of Massachusetts Boston
100 Morrissey Blvd., Boston, MA 02125-3393
Ping.chen@umb.edu
http://www.cs.umb.edu/~pchen

John Rochford
UMass Medical School/E.K. Shriver Center
john.rochford@umassmed.edu

David N. Kennedy
Department of Psychiatry
Division of Neuroinformatics
Center for Reproducible Neuroimaging Computation University of Massachusetts Medical Center David.Kennedy@umassmed.edu

Soussan Djamasbi
Worcester Polytechnic Institute
djamasbi@WPI.EDU
http://uxdm.wpi.edu/

Peter Fay
IBM Accessibility
peter_fay@us.ibm.com

Will Scott
IBM Accessibility | IBM Research
scottw1@us.ibm.com

Text simplification (TS) aims to reduce the lexical and structural complexity of a text, while still retaining the semantic meaning. Current automatic TS techniques are limited to either lexical-level applications or manually defining a large amount of rules. In this paper, we propose to simplify text from both level of lexicons and sentences. We conduct preliminary experiments to find that our approach shows promising results.

Keywords: Automatic Text Simplification, Natural Language Processing

1. Introduction

Approximately 10% of the world's population has an intellectual disability. Many people with ID face challenges while understanding text passages. Most have reading levels below their own mental age. However, by adolescence, people with mild intellectual impairment, who comprise the vast majority of people with ID, can often achieve up to the sixth-grade reading level. Regardless, limitations in literacy and reading comprehension present significant challenges for this population; and can thwart opportunities for decision-making and many aspects of independent living.

Information and communication technologies (ICT), in particular the World Wide Web, are increasingly a primary vehicle by which people obtain employment, interact with community agencies, make purchases, and conduct many other essential tasks of daily living. Examples where simplified text in ICT will have a positive impact on people with ID include:

- 80% of Fortune 500 companies require online job applications
- Banking
- Socializing

For those who have limited literacy, the complex text found on the vast majority of websites renders the web inaccessible and, by extension, contributes to the disparities often experienced by people with ID.

Text Simplification (TS) aims to simplify the lexical, grammatical, or structural complexity of text while retaining its semantic meaning. It can help various groups of people, including children, non-native speakers, the functionally illiterate, and people with cognitive disabilities, to understand text better [23].

Automatic TS is a complicated natural language processing (NLP) task, it consists of lexical, syntactic, or discourse simplification levels [8]. Usually hand- crafted, supervised and unsupervised methods based on resources like English Wikipedia (EW) and Simple English Wikipedia (SEW) [2] [10] [6] [25] [26] are utilized for extracting simplification rules, where EW and SEW are widely used for collecting aligned sentence pairs from paired articles.

It is very easy to mix up automatic text simplification task and automatic summarization task. TS is different to text summarization as the focus of text summarization is to reduce the length and the redundancy content. However,

text simplification usually keeps all content, and the outputs are not necessarily shorter. TS is also strongly related to but distinct from Machine Translation (MT). Since we can regard the original English and the simplified English as two different languages.

Automatic TS research has been developed for decades. In the early works, TS is considered as a preprocessor for other NLP tasks such as machine translation, parsing, summarization [4]. Then in [3], a TS system for language impaired readers is developed. The system consists of lexical tagger, morphological analyzer, parser, syntactic simplifier, lexical simplifier, etc, which is similar to the most state-of-the-art text simplification system. Semantic evaluation task also motivated the research for text simplification, such as in SemEval-2007 Task 10: English Lexical Substitution Task and SemEval-2012 Task 1: English Lexical Simplification [11] [20]. TS is a largely unsolved task, while this field is fast moving and research into this area is regularly produced [17]. In this survey, we will study the state-of-the-art automatic text simplification research for lexical and sentence simplification.

2. Lexical Simplification

Lexical simplification (LS) simplifies text mainly by substituting infrequently-used and difficult words with frequently-used and easier words. In order to generate substituting rules, most LS systems refer to lexical semantic resources like WordNet [13] [2] [3] by selecting synonyms based word frequency, or utilize English Wikipedia (EW) and Simple English Wikipedia (SEW).

The general process for lexical simplification includes: identification of complex words; finding synonyms or similar words by various similarity measures; ranking and selecting the best candidate word based on criteria such as language model; and keeping the grammar and syntax of a sentence correct [23]. To consider the candidates for substitution, word that shares a common lemma, is a prefix or suffix of another, has same part of speech, and the part of speech is labelled as proper noun must be removing [10].

In the work of [2], they proposed an unsupervised method for learning pairs of complex and simpler synonyms, and a context aware method for substituting one for the other, without requiring aligned simplex and complex sentence pairs on EW and SEW. The definition of the complexity of a word is based on two measures: the corpus complexity (word frequency) and the lexical complexity (word length). The final complexity is given by the product of the two. Similarity measures are used to decide if transformations should be applied. On

the other hand, [10] leveraged a data set of 137K aligned sentence pairs between EW and SEW to learn lexical simplification rules, and used feature-based approaches (candidate probability, frequency, language model, context frequency) [20] to learn a ranking function in SVM to make decision for transformation. For training and evaluation of the model, they collected human labeling of 500 lexical simplification examples from Amazon Mechanical Turk. Since all of the aforementioned methods is the dependence on simplified corpora and WordNet, in contrast, [9] proposed a LS system which only requires large corpus of regular text to obtain word embedding [12] [14] to get similar words of the complex word.

However, three main challenges exist for lexical simplification approach. First, a great number of transformation rules are required for a reasonable coverage; second, even for the same word, different transformation rules should be applied based on the specific context; third, the syntax and semantic meaning of the sentence must be retained.

3. Sentence Simplification

At sentence level, reading difficulty stems from lexical or syntactic complexity. Therefore, sentence simplification usually has two steps: lexical simplification and syntactic simplification. Splitting, dropping, reordering, and substitution are widely accepted as the significant simplification operations [26]. [15] described a corpus of original and simplified news articles, and analyzed the syntactic features for decisions about sentence splitting, and for which position and redundancy information to keep or drop.

In [26], they proposed a sentence simplification model by tree transformation based on Statistical Machine Translation (SMT), and provide a probabilistic model for each of the operation rules: splitting, dropping, reordering and substitution. TF-IDF similarity measure is used for align sentences from SEW and EW. [22] proposed a general method for learning how to iteratively simplify a sentence, thus decomposing complicated syntax into small, easy-to-process pieces based on the parsing tree, the method applies hand-written transformation rules corresponding to basic syntactic patterns. In [18], they formalized the interactions that take place between syntax and discourse during the simplification process, and described how various generation issues like sentence ordering, cue-word selection, referring-expression generation, determiner choice and pronominal use can be resolved. [19] described a text simplification system that uses a synchronous grammar defined over typed

dependencies. [24] presented a data-driven model based on quasi-synchronous grammar, a formalism that can naturally capture structural mismatches and complex rewrite operations. [25] explored text simplification rules based on the edit histories in SEW.

The limitation of aforementioned methods requires syntax parser or hand-crafted rules to simplify sentence. [23] proposed to use RNN (Recurrent Neural Network) Encoder-Decoder for text simplification. RNN Encoder-Decoder is a very popular deep neural network model that obtains great success in machine translation task [5] [21] [1]. However, it is difficult to train the model due to the lack of paired simple and complex sentences.

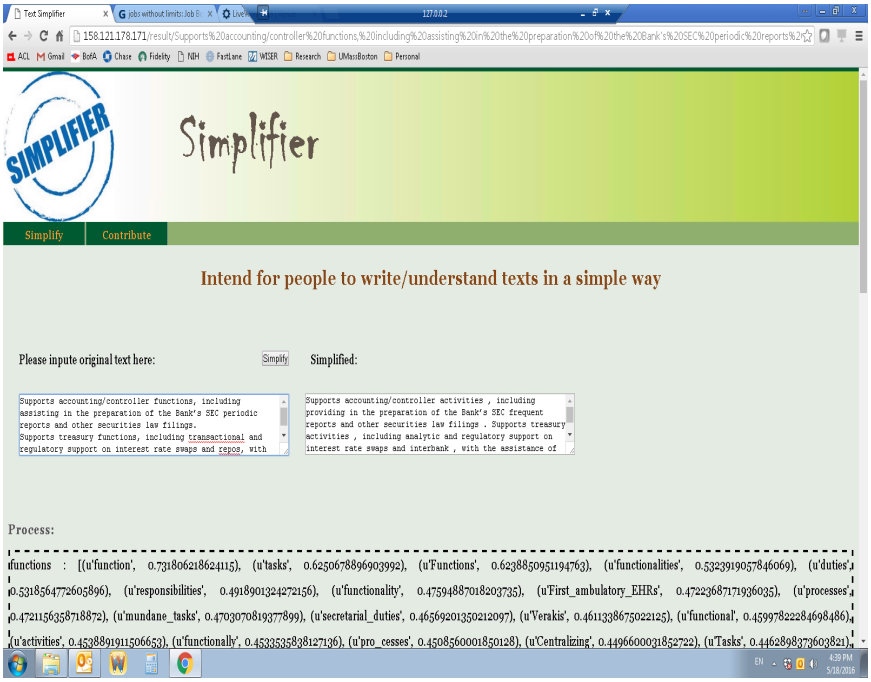

Fig. 1. Our prototype text simplification system

4. Experiment

Most of the work just utilized the annotated data for evaluation. It is still an open problem of how to measure simplicity automatically. Typical measures take into account surface text factors such as sentence length, syllable count, word frequency [17] [2]. They can give a good estimation, but are far from accurate. Two other important terms are readability and understandability,

readability defines how easy to read a text, and understandability is the amount of information people can gain from the text [17]. Unfortunately, there is no existing automatic way to use readability or understandability to evaluate TS. Eye tracking has been used successfully as a technique for cognitive load in reading, language acquisition, etc. Therefore, some works apply eye tracking to evaluate machine translation and text simplification [16] [7].

In this project, we have preliminarily implemented some techniques discussed above into a real-world system available at: http:// http://158.121.178.171/contribute/. A screen shot of our system is shown in Fig 1.

Conclusion

Text simplification is a challenging task in Natural Language Processing. This is a preliminary study in solving the text simplification problem for people with intellectual disabilities. Unlike the machine translation task, there are very few text simplification training corpora online. So our future work includes collecting complex and simple sentence pairs from online resources such as English Wikipedia and Simple English Wikipedia, and training our model using these large parallel corpora.

Acknowledgments

This project is partially supported by funding from University of Massachussetts Medical School.

References

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473, 2014.

[2] Or Biran, Samuel Brody, and No´emie Elhadad. 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. Association for Computational Linguistics, 2011.

[3] John Carroll, Guido Minnen, Darren Pearce, Yvonne Canning, Siobhan Devlin, and John Tait. Simplifying text for language-impaired readers. In Proceedings of EACL, volume 99, pages 269–270, 1999.

[4] Raman Chandrasekar, Christine Doran, and Bangalore Srinivas. Motivations and methods for text simplification. In Proceedings of the 16th conference on Computational linguistics-Volume 2, pages 1041–1044. Association for Computational Linguistics, 1996.

[5] Kyunghyun Cho, Bart Van Merri¨enboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078, 2014.

[6] William Coster and David Kauchak. Simple English wikipedia: a new text simplification task. In Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: short papers-Volume 2, pages 665–669. Association for Computational Linguistics, 2011.

[7] Stephen Doherty, Sharon OBrien, and Michael Carl. Eye tracking as an mt evaluation technique. Machine translation, 24(1):1–13, 2010.

[8] Lijun Feng. Text simplification: A survey. The City University of New York, Tech. Rep, 2008.

[9] Goran Glava˘s and Sanja S˘tajner. Simplifying lexical simplification: Do we need simplified corpora? page 63, 2015.

[10] Colby Horn, Cathryn Manduca, and David Kauchak. Learning a lexical simplifier using wikipedia. In Proceedings of the 52th Annual Meeting of the Association for Computational Linguistics (ACL-2014, Short Papers), pages 458–463, 2014.

[11] Diana McCarthy and Roberto Navigli. Semeval-2007 task 10: English lexical substitution task. In Proceedings of the 4th International Workshop on Semantic Evaluations, pages 48–53. Association for Computational Linguistics, 2007.

[12] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean.

Distributed representations of words and phrases and their compositional- ity. In Advances in neural information processing systems, pages 3111–3119, 2013.

[13] George A Miller. Wordnet: a lexical database for English. Communications of the ACM, 38(11):39–41, 1995.

[14] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In EMNLP, volume 14, pages 1532–1543, 2014.

[15] Sarah E Petersen and Mari Ostendorf. Text simplification for language learners: a corpus analysis. In SLaTE, pages 69–72. Citeseer, 2007.

[16] Luz Rello, Ricardo Baeza-Yates, Laura Dempere-Marco, and Horacio Saggion. Frequent words improve readability and short words improve understandability for people with dyslexia. In Human-Computer Interaction–INTERACT 2013, pages 203–219. Springer, 2013.

[17] Matthew Shardlow. A survey of automated text simplification. International Journal of Advanced Computer Science and Applications, 4(1), 2014.

[18] Advaith Siddharthan. Syntactic simplification and text cohesion. Research on Language & Computation, 4(1):77–109, 2006.

[19] Advaith Siddharthan and Angrosh Mandya. Hybrid text simplification using synchronous dependency grammars with hand-written and automatically harvested rules. In Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, pages 722–731, 2014.

[20] Lucia Specia, Sujay Kumar Jauhar, and Rada Mihalcea. Semeval-2012 task 1: English lexical simplification. In 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, pages 347–355. Association for Computational Linguistics, 2012.

[21] Ilya Sutskever, Oriol Vinyals, and Quoc VV Le. Sequence to sequence learning with neural networks. In Advances in neural information process- ing systems, pages 3104–3112, 2014.

[22] David Vickrey and Daphne Koller. Sentence simplification for semantic role labeling. In ACL, pages 344–352, 2008.

[23] Tong Wang, Ping Chen, John Rochford, and Jipeng Qiang. Text simplification using neural machine translation. In Thirtieth AAAI Conference on Artificial Intelligence, 2016.

[24] Kristian Woodsend and Mirella Lapata. Learning to simplify sentences with quasi-synchronous grammar and integer programming. In Proceedings of the

conference on empirical methods in natural language processing, pages 409–420. Association for Computational Linguistics, 2011.

[25] Mark Yatskar, Bo Pang, Cristian Danescu-Niculescu-Mizil, and Lillian Lee. For the sake of simplicity: Unsupervised extraction of lexical simplifications from wikipedia. In Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics, pages 365–368. Association for Computational Linguistics, 2010.

[26] Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. A monolingual tree-based translation model for sentence simplification. In Proceedings of the 23rd international conference on computational linguistics, pages 1353–1361. Association for Computational Linguistics, 2010.