Exploring Neural Text Simplification Models

Sergiu Nisioi[1,3,*] **Sanja Štajner**[2,*] **Simone Paolo Ponzetto**[2] **Liviu P. Dinu**[1]

[1]Human Language Technologies Research Center, University of Bucharest, Romania
[2]Data and Web Science Group, University of Mannheim, Germany
[3]Oracle Corporation, Romania
{sergiu.nisioi,ldinu}@fmi.unibuc.ro
{sanja,simone}@informatik.uni-mannheim.de

Abstract

We present the first attempt at using sequence to sequence neural networks to model text simplification (TS). Unlike the previously proposed automated TS systems, our neural text simplification (NTS) systems are able to simultaneously perform lexical simplification and content reduction. An extensive human evaluation of the output has shown that NTS systems achieve almost perfect grammaticality and meaning preservation of output sentences and higher level of simplification than the state-of-the-art automated TS systems.

1 Introduction

Neural sequence to sequence models have been successfully used in many applications (Graves, 2012), from speech and signal processing to text processing or dialogue systems (Serban et al., 2015). Neural machine translation (Cho et al., 2014; Bahdanau et al., 2014) is a particular type of sequence to sequence model that recently attracted a lot of attention from industry (Wu et al., 2016) and academia, especially due to the capability to obtain state-of-the-art results for various translation tasks (Bojar et al., 2016). Unlike classical statistical machine translation (SMT) systems (Koehn, 2010), neural networks are being trained end-to-end, without the need to have external decoders, language models or phrase tables. The architectures are relatively simpler and more flexible, making possible the use of character models (Luong and Manning, 2016) or even training multilingual systems in one go (Firat et al., 2016).

Automated text simplification (ATS) systems are meant to transform original texts into different (simpler) variants which would be understood by wider audiences and more successfully processed by various NLP tools. In the last several years, great attention has been given to addressing ATS as a monolingual machine translation problem translating from 'original' to 'simple' sentences. So far, attempts were made at standard phrase-based SMT (PBSMT) models (Specia, 2010; Štajner et al., 2015), PBSMT models with added phrasal deletion rules (Coster and Kauchak, 2011) or reranking of the n-best outputs according to their dissimilarity to the output (Wubben et al., 2012), tree-based translation models (Zhu et al., 2010; Paetzold and Specia, 2013), and syntax-based MT with specially designed tuning function (Xu et al., 2016). Recently, lexical simplification (LS) was addressed by unsupervised approaches leveraging word-embeddings, with reported good success (Glavaš and Štajner, 2015; Paetzold and Specia, 2016).

To the best of our knowledge, our work is the first to address the applicability of neural sequence to sequence models for ATS. We make use of the recent advances in neural machine translation (NMT) and adapt the existing architectures for our specific task. We also perform an extensive human evaluation to directly compare our systems with the current state-of-the-art (supervised) MT-based and unsupervised lexical simplification systems.

2 Neural Text Simplification (NTS)

We use the OpenNMT framework (Klein et al., 2017) to train and build our architecture with two LSTM layers (Hochreiter and Schmidhuber, 1997), hidden states of size 500 and 500 hidden units, and a 0.3 dropout probability (Srivastava et al., 2014). The vocabulary size is set to 50,000 and we train the model for 15 epochs with plain SGD optimizer, and after epoch 8 we halve the

[*] Both authors have contributed equally to this work

learning rate. At the end of each epoch we save the current state of the model and predict the perplexity values of the models on the development set. We employ early-stopping and select the model resulted from the epoch with the best perplexity to avoid over-fitting. The parameters are initialized over uniform distribution with support [-0.1, 0.1]. Additionally, for the decoder we employ global attention in combination with input feeding as described by Luong et al. (2015). The architecture[1] is depicted in Figure 1, with the input feeding approach represented only for the last hidden state of the decoder.

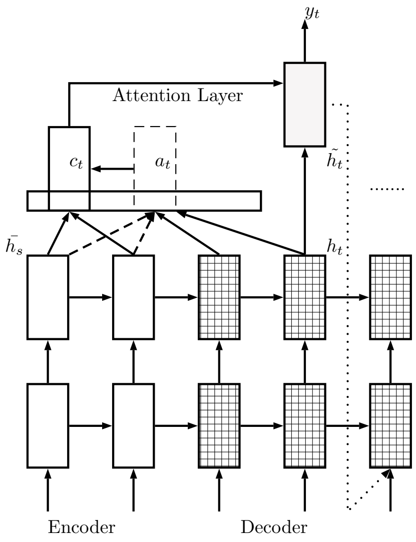

Figure 1: Architecture of the neural simplification model with global attention and input feeding.

For the attention layer, we compute a context vector c_t by using the information provided from the hidden states of the source sentence and by computing a weighted average with the alignment weights a_t. The new hidden state is obtained using a concatenation of the previous hidden state and the context vector:

$$\tilde{h}_t = \tanh W[c_t; h_t]$$

The global alignment weights a_t are being computed with a softmax function over the general scoring method for attention:

$$a_t(s) = \frac{\exp h_t^T W_{as} \bar{h}_s}{\sum_{s'} \exp h_t^T W_{as'} \bar{h}_{s'}}$$

Input feeding is a process that sends the previous hidden state obtained using the alignment method, to the input at the next step, presumably making the model keep track of anterior alignment decisions. Luong et al. (2015) showed this approach can increase the evaluation scores for neural machine translation, while in our case, for monolingual data, we believe it can be helpful to create better alignments. Our approach does not involve the use of character-based models (Sennrich et al., 2015; Luong and Manning, 2016) to handle out of vocabulary words and entities. Instead, we make use of alignment probabilities between the predictions and the original sentences to retrieve the original words.

2.1 Word2vec Embeddings

Furthermore, we are interested to explore whether large scale pre-trained embeddings can improve text simplification models. Kauchak (2013) indicates that combining normal data with simplified data can increase the performance of ATS systems. Therefore, we construct a secondary model (NTS-w2v) using a combination of pre-trained word2vec from Google News corpus (Mikolov et al., 2013a) of size 300 and locally trained embeddings of size 200. To ensure good representations of low-frequency words, we use word2vec (Řehůřek and Sojka, 2010; Mikolov et al., 2013b) to train skip-gram with hierarchical softmax and we set a window of 10 words.

Following Garten et al. (2015) who showed that simple concatenation can improve the word representations, we construct two different sets of embeddings for the encoder and for the decoder. The former are constructed using the word2vec trained on the original English texts combined with Google News and the later (decoder) embeddings are built from word2vec trained on the simplified version of the training data combined with Google News. To merge the local and global embeddings, we concatenate the representations for each word in the vocabulary, thus obtaining a new representation of size 500. If a word is missing in the global embeddings, we replace it with a sample from a Gaussian distribution with mean 0 and standard deviation of 0.9. The remaining parameters are left unchanged from the previous model description.

2.2 Prediction Ranking

To ensure the best predictions and the best simplified sentences at each step, we use beam search to sample multiple outputs from the two systems

[1] The architecture configurations, data, and the pre-trained models are released in `https://github.com/senisioi/NeuralTextSimplification`

described previously (NTS and NTS-w2v). Beam search works by generating the first k hypotheses at each step ordered by the log-likelihood of the target sentence given the input sentence. By default, we use a beam size of 5 and take the first hypothesis, but we also observe that higher beam size and lower-ranked hypotheses can generate good simplification results. Therefore, we generate the first two candidate hypotheses for each beam size from 5 to 12. We then attempt to find the best beam size and hypothesis based on two metrics: the traditional MT-evaluation metric, BLEU (Papineni et al., 2002; Bird et al., 2009) with NIST smoothing (Bird et al., 2009), and SARI (Xu et al., 2016), a recent text-simplification metric.

2.3 Dataset

To train our models, we use the publicly available dataset provided by Hwang et al. (2015) based on manual and automatic alignments between standard English Wikipedia and Simple English Wikipedia (EW–SEW). We discard the uncategorized matches, and use only *good matches* and *partial matches* which were above the 0.45 threshold (Hwang et al., 2015), totaling to 280K aligned sentences (around 150K full matches and 130K partial matches). It is one of the largest freely available resources for text simplification, and unlike the previously used EW–SEW corpus[2] (Kauchak, 2013), which only contains *full matches* (167K pairs), the newer dataset also contains *partial matches*. Therefore, it is not only larger, but it also allows for learning sentence shortening (dropping irrelevant parts) transformations (see Table 3, Appendix A).

	original	**simplified**
locations	158,394	127,349
persons	161,808	127,742
organizations	130,679	101,239
misc	95,168	71,138
vocabulary	187,137	144,132
tokens	7,400,499	5,634,834

Table 1: The number of tokens and entities in the corpus.

We use the Stanford NER system (Finkel et al., 2005) to get an approximate number of locations, persons, organizations and miscellaneous entities in the corpus. A brief analysis of the vocabulary is rendered in Table 1.

The dataset we use contains an abundant amount of named entities and consequently a large amount of low frequency words, but the majority of entities are not part of the model's 50,000 words vocabulary due to their small frequency. These words are replaced with 'UNK' symbols during training. At prediction time, we replace the unknown words with the highest probability score from the attention layer. We believe it is important to ensure that the models learn good word representations, either during the model training or through word2vec, in order to accurately create alignments between source and target sentences.

Given that in TS there is not only one best simplification, and that the quality of simplifications in Simple English Wikipedia has been disputed before (Amancio and Specia, 2014; Xu et al., 2015), for tuning and testing we use the dataset previously released by Xu et al. (2016), which contains 2000 sentences for tuning and 359 for testing, each with eight simplification variants obtained by eight Amazon Mechanical Turkers.[3] The tune subset is also used as reference corpus in combination with BLEU and SARI to select the best beam size and hypothesis for prediction reranking.

3 Evaluation

For the first 70 original sentences of the Xu *et al.*'s (2016) test set[4] we perform three types of human evaluation to assess the output of our best systems and three ATS systems of different architectures: (1) the PBSMT system with reranking of n-best outputs (Wubben et al., 2012), which represent the best PBSMT approach to ATS, trained and tuned over the same datasets as our systems; (2) the state-of-the-art SBMT system (Xu et al., 2016) with modified tuning function (using SARI) and using PPDB paraphrase database (Ganitkevitch et al., 2013);[5] and (3) one of the state-of-the-art unsupervised lexical simplification (LS) systems that leverages word-embeddings (Glavaš and

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

[3] None of the 359 test sentences was present in the datasets we used for training and tuning.

[4] https://github.com/cocoxu/simplification/

[5] For the first two systems, we use publicly available output at: https://github.com/cocoxu/simplification/tree/master/data/systemoutputs

Štajner, 2015).[6]

We evaluate the output of all systems using three types of human evaluation.

Correctness and Number of Changes. First, we count the total number of changes made by each system (*Total*), counting the change of a whole phrase (e.g. "*become defunct*" $\rightarrow$ "*was dissolved*") as one change. Those changes that preserve the original meaning and grammaticality of the sentence (assessed by two native English speakers) and, at the same time, make the sentence easier to understand (assessed by two non-native fluent English speakers) are marked as *Correct*. In the case of content reduction, we instructed the annotators to count the deletion of each array of consecutive words as one change and consider the meaning unchanged if the main information of the sentence was retained and unchanged. The sentences for which the two annotators did not agree were given to a third annotator to obtain the majority vote.

Grammaticality and Meaning Preservation. Second, three native English speakers rate the grammaticality (*G*) and meaning preservation (*M*) of each (whole) sentence with at least one change on a 1–5 Likert scale (1 – very bad; 5 – very good). The obtained inter-annotator agreement (quadratic Cohens kappa) was 0.78 for G and 0.63 for M.

Simplicity of sentences. Third, the three non-native fluent English speakers were shown original (reference) sentences and target (output) sentences, one pair at the time, and asked whether the target sentence is: +2 – much simpler; +1 – somewhat simpler; 0 – equally difficult; -1 – somewhat more difficult; -2 – much more difficult, than the reference sentence. The obtained inter-annotator agreement (quadratic Cohens kappa) was 0.66.

While the correctness of changes takes into account the influence of each individual change on grammaticality, meaning and simplicity of a sentence, the *Scores (G and M)* and *Rank (S)* take into account the mutual influence of all changes within a sentence.

4 Results and Discussion

The results of the human evaluation (Table 2) revealed that all NTS models achieve higher percentage of correct changes and more simplified output than any of the state-of-the-art ATS systems with different architectures (PBSMT-R, SBMT, and LightLS). We also notice that the best models according to BLEU are obtained with hypothesis 1 and the maximum beam size for both models, while the SARI re-ranker prefers hypothesis 2 and beam size 5 for the first NTS and the maximum beam size for the custom word embeddings model.

The NTS with custom word2vec embeddings ranked with the text simplification specific metric (SARI) obtained the highest total number of changes among the neural systems, one of the highest percentage of correct changes, the second highest simplicity score, and solid grammaticality and meaning preservation scores. An example of the output of different systems is presented in Table 4 (Appendix A).

The use of different metrics for ranking the NTS predictions optimizes the output towards different evaluation objectives: SARI leads to the highest number of total changes, BLEU to the highest percentage of correct changes, and the default beam scores to the best grammaticality (G) and meaning preservation (M). In addition, custom composed global and local word embeddings in combination with SARI metric improve the default translation system, given the joint scores for each evaluation criterion.

Here is important to note that for ATS systems, the precision of the system (correctness of changes, grammaticality, meaning preservation, and simplicity of the output) is more important than the recall (the total number of changes made). The low recall would just leave the sentences similar to their originals thus not improving much the understanding or reading speed of the target users, or not improving much the NLP systems in which they are used as a pre-processing step. A low precision, on the other hand, would make texts even more difficult to read and understand, and would worsen the performances of the NLP systems in which ATS is used as a pre-processing step.

5 Conclusions

We presented a first attempt at modelling sentence simplification with a neural sequence to sequence model. Our extensive human evaluation showed that our NTS systems, if the output is ranked with the right metric, can significantly[7] outperform the best phrase-based and syntax-based MT approaches, and unsupervised lexical ATS approach,

[6] For the LightLS system (Glavaš and Štajner, 2015) we use the output of the original system provided by the authors.

[7] Wilcoxon's signed rank test, $p < 0.001$.

Approach	Changes		Scores		Rank	SARI	BLEU
	Total	Correct	G	M	S		
NTS default (beam 5, hypothesis 1)	36	72.2%	**4.92**	**4.31**	+0.46	30.65	84.51
NTS SARI (beam 5, hypothesis 2)	72	51.6%	4.19	3.62	+0.38	37.25	80.69
NTS BLEU (beam 12, hypothesis 1)	44	73.7%	4.77	4.15	+**0.92**	30.77	84.70
NTS-w2v default (beam 5, hypothesis 1)	31	54.8%	4.79	4.17	+0.21	31.11	**87.50**
NTS-w2v SARI (beam 12, hypothesis 2)	110	68.1%	4.53	3.83	+**0.63**	36.10	79.38
NTS-w2v BLEU (beam 12, hypothesis 1)	61	**76.9%**	4.67	4.00	+0.40	30.67	85.03
PBSMT-R (Wubben et al., 2012)	**171**	41.0%	3.10	2.71	−0.55	34.07	67.79
SBMT (SARI+PPDB) (Xu et al., 2016)	143	34.3%	4.28	3.57	+0.03	**38.59**	73.62
LightLS (Unsupervised) (Glavaš and Štajner, 2015)	132	26.6%	4.47	2.67	−0.01	34.96	83.54

Table 2: Human evaluation results (the highest scores by each evaluation criterion are shown in bold).

by grammaticality, meaning preservation and simplicity of the output sentences, the percentage of correct transformations, while at the same time achieving more than 1.5 changes per sentence, on average. Furthermore, we discovered that NTS systems are capable of correctly performing significant content reduction, thus being the only TS models proposed so far which can jointly perform lexical simplification and content reduction.

Acknowledgments

This work has been partially supported by a grant of the Romanian National Authority for Scientific Research and Innovation, CNCS/CCCDI UEFISCDI, project number PN-III-P2-2.1-53BG/2016, within PNCDI III, and by the SFB 884 on the Political Economy of Reforms at the University of Mannheim (project C4), funded by the German Research Foundation (DFG).

References

Marcelo Adriano Amancio and Lucia Specia. 2014. An Analysis of Crowdsourced Text Simplifications . In *Proceedings of the 3rd Workshop on Predicting and Improving Text Readability for Target Reader Populations (PITR)*. pages 123–130.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR* abs/1409.0473. http://arxiv.org/abs/1409.0473.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. 2016. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*. Association for Computational Linguistics, pages 131–198.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. pages 1724–1734.

William Coster and David Kauchak. 2011. Simple English Wikipedia: a new text simplification task. In *Proceedings of ACL&HLT*. pages 665–669.

Jenny Rose Finkel, Trond Grenager, and Christopher Manning. 2005. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd annual meeting on association for computational linguistics*. Association for Computational Linguistics, pages 363–370.

Orhan Firat, KyungHyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. *CoRR* abs/1601.01073.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The Paraphrase Database. In *Proceedings of NAACL-HLT*. pages 758–764.

Justin Garten, Kenji Sagae, Volkan Ustun, and Morteza Dehghani. 2015. Combining distributed vector representations for words. In *Proceedings of NAACL-HLT*. pages 95–101.

Goran Glavaš and Sanja Štajner. 2015. Simplifying Lexical Simplification: Do We Need Simplified Corpora? In *Proceedings of the ACL&IJCNLP 2015 (Volume 2: Short Papers)*. pages 63–68.

Alex Graves. 2012. *Supervised sequence labelling with recurrent neural networks*, volume 385. Springer.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation* 9(8):1735–1780.

William Hwang, Hannaneh Hajishirzi, Mari Ostendorf, and Wei Wu. 2015. Aligning Sentences from Standard Wikipedia to Simple Wikipedia. In *Proceedings of NAACL&HLT*. pages 211–217.

David Kauchak. 2013. Improving text simplification language modeling using unsimplified text data. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. ACL, pages 1537–1546.

G. Klein, Y. Kim, Y. Deng, J. Senellart, and A. M. Rush. 2017. OpenNMT: Open-Source Toolkit for Neural Machine Translation. *ArXiv e-prints* .

Philipp Koehn. 2010. Statistical machine translation.

Minh-Thang Luong and Christopher D Manning. 2016. Achieving open vocabulary neural machine translation with hybrid word-character models. *arXiv preprint arXiv:1604.00788* .

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *EMNLP*. The Association for Computational Linguistics, pages 1412–1421.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *Proceedings of Workshop at International Conference on Learning Representations*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. pages 3111–3119.

Gustavo Henrique Paetzold and Lucia Specia. 2013. Text simplification as tree transduction. In *Proceedings of the 9th Brazilian Symposium in Information and Human Language Technology*. pages 116–125.

Gustavo Henrique Paetzold and Lucia Specia. 2016. Unsupervised lexical simplification for non-native speakers. In *Proceedings of the 30th AAAI*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of ACL*.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. ELRA, Valletta, Malta, pages 45–50.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909* .

Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2015. Building end-to-end dialogue systems using generative hierarchical neural network models. *arXiv preprint arXiv:1507.04808* .

Lucia Specia. 2010. Translating from complex to simplified sentences. In *Proceedings of the 9th international conference on Computational Processing of the Portuguese Language (PROPOR)*. Springer Berlin Heidelberg, volume 6001 of *Lecture Notes in Computer Science*, pages 30–39.

Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15(1):1929–1958.

Sanja Štajner, Hannah Bechara, and Horacio Saggion. 2015. A Deeper Exploration of the Standard PB-SMT Approach to Text Simplification and its Evaluation. In *Proceedings of ACL&IJCNLP (Volume 2: Short Papers)*. pages 823–828.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144* .

Sander Wubben, Antal van den Bosch, and Emiel Krahmer. 2012. Sentence simplification by monolingual machine translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (ACL): Long Papers - Volume 1*. Association for Computational Linguistics, pages 1015–1024.

Wei Xu, Chris Callison-Burch, and Courtney Napoles. 2015. Problems in Current Text Simplification Research: New Data Can Help. *Transactions of the Association for Computational Linguistics (TACL)* 3:283–297.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics* 4:401–415.

Z. Zhu, D. Berndard, and I. Gurevych. 2010. A Monolingual Tree-based Translation Model for Sentence Simplification. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*. pages 1353–1361.

A Appendix - Data Sample and System Output

Match	Transformation	Sentence pair
Full	syntactic simplification; reordering of sentence constituents	“During the 13th century, gingerbread was brought to Sweden by German immigrants.” and “German immigrants brought it to Sweden during the 13th century.”
Full	lexical paraphrasing	“During the 13th century, gingerbread was brought to Sweden by German immigrants.” and “German immigrants brought it to Sweden during the 13th century.”
Partial	strong paraphrasing	“Gingerbread foods vary, ranging from a soft, moist loaf cake to something close to a ginger biscuit.” and “Gingerbread is a word which describes different sweet food products from soft cakes to a ginger biscuit.”
Partial	adding explanations	“Humidity is the amount of water vapor in the air.” and “Humidity (adjective: humid) refers to water vapor in the air, but not to liquid droplets in fog, clouds, or rain.”
Partial	sentence compression; dropping irrelevant information	“Falaj irrigation is an ancient system dating back thousands of years and is used widely in Oman, the UAE, China, Iran and other countries.” and “The ancient falaj system of irrigation is still in use in some areas.”

Table 3: Examples of full and partial matches from the EW–SEW dataset (Hwang et al., 2015).

System	Output
NTS-w2v default	Perry Saturn (with terri) defeated Eddie Guerrero (with chyna) to win the WWF European Championship (8:10); Saturn pinned Guerrero after a diving elbow drop.
NTS-w2v SARI	Perry Saturn **pinned Guerrero to win the WWF European Championship.**
NTS-w2v BLEU	Perry Saturn pinned Guerrero after a diving **drop** drop.
NTS default	**He** (with terri) defeated Eddie Guerrero (with chyna) to win the WWF European Championship (8:10); Saturn pinned Guerrero after a diving elbow drop.
NTS BLEU/SARI	**He** defeated Eddie Guerrero (with Chyna) to win the WWF European Championship (8:10); Saturn pinned Guerrero after a diving elbow drop.
LightLS (Glavaš and Štajner, 2015)	Perry Saturn (with terri) defeated Eddie Guerrero (with chyna) to win the WWF European Championship (8:10); Saturn pinned Guerrero after a **swimming shoulder fall**.
SBMT (Xu et al., 2016)	Perry Saturn (with terri) **beat** Eddie Guerrero (with chyna) to win the WWF European **League** (8:10); Saturn pinned Guerrero after a diving elbow drop.
PBSMT-R (Wubben et al., 2012)	Perry Saturn with terri **and** Eddie Guerrero **,** chyna **,** to win the European Championship **then-wwf** 8:10); **he** pinned Guerrero after a diving elbow drop.
Original	Perry Saturn (with terri) defeated Eddie Guerrero (with chyna) to win the WWF European Championship (8:10); Saturn pinned Guerrero after a diving elbow drop.

Table 4: Output examples, differences to the original sentence are shown in bold.