Unsupervised Rewriter for Multi-Sentence Compression

Yang Zhao[1], **Xiaoyu Shen**[2], **Wei Bi**[3], **Akiko Aizawa**[4,1]
[1]The University of Tokyo, Tokyo, Japan zhao@is.s.u-tokyo.ac.jp
[2]Max Planck Institute for Informatics, Saarland, Germany xshen@mpi-inf.mpg.de
[3]Tencent AI Lab, Shenzhen, China victoriabi@tencent.com
[4]National Institute of Informatics, Tokyo, Japan aizawa@nii.ac.jp

Abstract

Multi-sentence compression (MSC) aims to generate a grammatical but reduced compression from multiple input sentences while retaining their key information. Previous dominating approach for MSC is the extraction-based word graph approach. A few variants further leveraged lexical substitution to yield more abstractive compression. However, two limitations exist. First, the word graph approach that simply concatenates fragments from multiple sentences may yield non-fluent or ungrammatical compression. Second, lexical substitution is often inappropriate without the consideration of context information. To tackle the above-mentioned issues, we present a neural rewriter for multi-sentence compression that does not need any parallel corpus. Empirical studies have shown that our approach achieves comparable results upon automatic evaluation and improves the grammaticality of compression based on human evaluation. A parallel corpus with more than 140,000 (sentence group, compression) pairs is also constructed as a by-product for future research.

1 Introduction

Multi-sentence compression (MSC) aims to generate a single shorter and grammatical sentence that preserves important information from a group of related sentences. Over the past decade, multi-sentence compression has attracted considerable attention owing to its potential applications, such as compressing the content to be displayed on screens with limited size (e.g., mobile devices) and benefiting other natural language processing tasks, such as multi-document summarization (Banerjee et al., 2015), opinion summarization, and text simplification. Most existing works rely on the word graph approach initialized in (Filippova, 2010), which offers a simple solution that copies fragments from different input sentences and concatenates them to form the final compression. Later on, a bunch of subsequent research works (Boudin and Morin, 2013; Banerjee et al., 2015; Luong et al., 2015; ShafieiBavani et al., 2016; Pontes et al., 2018; Nayeem et al., 2018) attempted to improve the word graph approach using a variety of strategies, such as keyphrase re-ranking. However, such extraction-based approach may yield non-fluent or ungrammatical compression. A previous study (Nayeem and Chali, 2017) has shown that word graph approaches produce more than 30% of the ungrammatical sentences (evaluated by a chart parser), which is partly due to the non-usage of rewording by these extraction-based approaches. In fact, human annotators tend to compress a sentence through several rewriting operations, such as substitution and rewording (Cohn and Lapata, 2008). Despite some research works that attempt to do the lexical substitution, it is often inappropriate without the consideration of context information.

To tackle the above-mentioned problems, we present herein an unsupervised rewriter to improve the grammaticality of compression while introducing an appropriate amount of novel words. Inspired by the unsupervised machine translation (Sennrich et al., 2015; Fevry and Phang, 2018), we adopted the back-translation technique to our setting. Unlike machine translation, in the case of compression task, multiple input sentences and single output compression usually do not have semantic equivalence, which complicates the application of the back-translation technique. Thus, we propose a rewriting scheme that first exploits word graph approach to produce coarse-grained compression (B), based on which we substitute words with their shorter synonyms to yield paraphrased sentence (C). A neural rewriter is subsequently applied to the semantically equivalent (B, C) pairs

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 2235–2240
Florence, Italy, July 28 - August 2, 2019.

in order to improve the grammaticality and encourage more novel words in compression. Our contributions are two-folds:(i) we present a neural rewriter for multi-sentence compression without any parallel data. This rewriter significantly improves the grammaticality and novel word rate, while maintaining the information coverage (informativeness) according to automatic evaluation and (ii) a large-scale multi-sentence compression corpus is introduced along with a manually created test set for future research. We release source code and data here[1].

2 Dataset Construction

The largest existing English corpus for multi-sentence compression is the Cornell corpus (McKeown et al., 2010), which has only 300 instances. We introduce herein a large-scale dataset by compiling the English Gigaword[2]. After preprocessing (e.g., filtering strange punctuations, etc.), 1.37 million news articles were yielded to group related sentences. The full procedure for the dataset construction is available here[3].

2.1 Group Related Sentences

The prerequisite for multi-sentence compression is that all input sentences should be related to the same topic or event. Inspired by (McKeown et al., 2010), if the sentences are too similar, one of the input sentences could be directly treated as a compression. In contrast, if the sentences are too dissimilar (no interaction), they may describe different events or topics. Both cases should be avoided because sentence compression would not be necessary. Here we use bi-gram similarity, which exhibited the highest accuracy (90%)[4]. We empirically arrived at 0.2 of the lower threshold of the bigram similarity to avoid very dissimilar sentences and 0.7 of the upper threshold of the bigram similarity to avoid near-identical sentences. As presented in Table 1, 140,572 sentence groups were finally yielded out of 1.37 million new articles. We refer to this as the Giga-MSC dataset.

[1]http://github.com/code4ai

[2]https://catalog.ldc.upenn.edu/LDC2011T07 English Gigaword, a comprehensive archive of newswire text data containing seven distinct international sources.

[3]http://github.com/code4ai/data

[4]Human judges were asked to evaluate whether the sentences in a group revolved around the same topic or event. A total of 45 out of 50 sentence groups were judged to be qualified.

# of sentences in a group	# of groups
2	133,123
3	6,633
4	816
In total	140,572

Table 1: Statistics of created Giga-MSC dataset.

2.2 Giga-MSC Dataset Annotation

We randomly selected 150 sentences for human annotation, which were used as reference compression in the automatic evaluation. Two annotators[5] were asked to generate one single reduced grammatical compression that satisfies two conditions:(1) conveys the important content of all the input sentences and (2) should be grammatically correct. We are interested in how the human annotators will perform this task without vocabulary constraints; hence, we did not tell them to introduce as little new vocabulary as possible in their compression as several previous works did (Boudin and Morin, 2013; Luong et al., 2015). Inter-agreement score Fleiss' Kappa (Artstein and Poesio, 2008) was also computed. The score was 0.43, demonstrating that moderate agreement was reached.

3 Methodology

Figure 1 illustrates our rewriting approach consisting of three steps.

3.1 Step.1 (A→B)

Given m input sentences, $s_1, s_2, ..., s_m$, called A, we use the keyphrase word graph approach (Boudin and Morin, 2013) to obtain coarse-grained compression, called B.

3.2 Step.1 (B→C)

C is yielded by substituting words and phrases in B with synonyms. We first identified all the multiword expressions in a sentence and determined all the synonyms in WordNet 3.0[6]. Keep in mind that our goal is to shorten the sentence as much as possible, we specifically substituted multiword expressions, such as *police_officer, united_states_of_america*, with their **shorter** synonyms *policeman and u.s.*. Because the size of synonyms in the WordNet dictionary is relatively limited, we also exploit PPDB 2.0[7] to replace

[5]Both annotators are native English speakers and not authors of this paper.

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

[7]https://paraphrase.prg

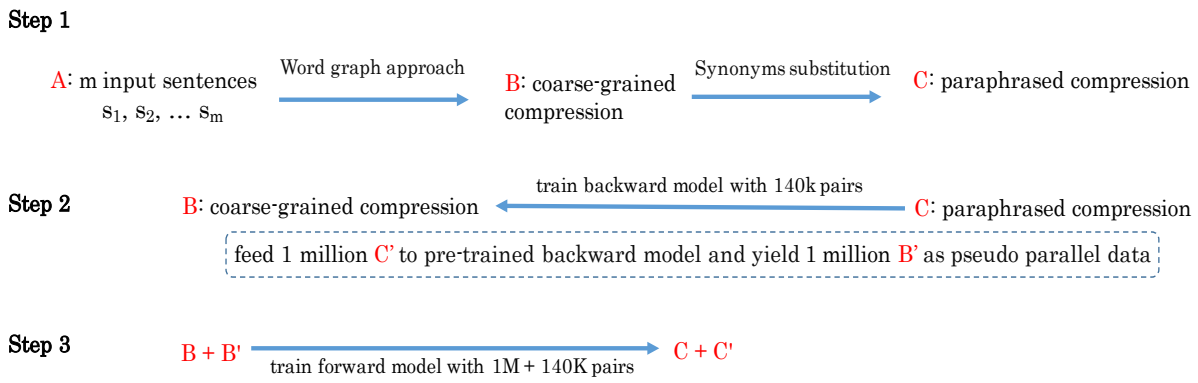

Figure 1: Graphic illustration for the rewriter model. A refers to multiple input sentences. B denotes a single compressed sentence using the word graph approach. C is the paraphrased sentence. C' is a large-scale and in-domain monolingual corpus, while B' refers to the predicted compression by a pre-trained backward model given C' as input. $B + B'$ and $C + C'$ are the mixing datasets.

the nouns, verbs, and adjectives with their shorter counterparts. For example, the verb *demonstrating* is converted into *proved*. By using the Giga-MSC dataset we created, 140,000 (A, B, C) tuples are yielded. Lexical substitution might lead to non-fluency C but significantly increases the number of novel words. Therefore, the next steps focus on creating pseudo parallel data to boost the fluency of C while attempting to maintain the rate of novel words.

3.3 Step2 (C→B)

Because the yielded B and C are semantically equivalent, we train a backward model (C→B) using 140,000 (C, B) pairs. The backward model consisted of a three-layer bi-directional LSTM encoder and a uni-directional decoder with attention mechanism. After the backward model was trained, one million grammatical in-domain sentences C' were given as input to generate one million B' The average length of C' was similar to that of C (30.2 tokens). We also found that C' maintained a novel rate of approximately 8.9, as compared to B'.

3.4 Step.3 (B+B'→C+C')

We merge the training data (coarse-grained compression B and non-fluent paraphrasing compression C) and the pseudo parallel data (pseudo sentence B' and grammatical sentence C') to jointly learn a forward model that consisted of a three-layer LSTM encoder and decoder. The vocabulary and word embedding were shared between both backward and forward models. We expect that because the grammatical C' accepts the majority of training data, it will improve the fluency of C.

4 Experiments

4.1 Datasets

We used two datasets to evaluate the model performance. First is the Giga-MSC dataset detailed in Section 2. A total of 150 annotated sentences were used as the ground truth for testing. Second is the Cornell dataset (McKeown et al., 2010).

4.2 Baseline Approaches

We considered (#1) the word graph approach (Filippova, 2010), and an advanced version (#2) keyphrase-based word graph model (Boudin and Morin, 2013) augmented with keyphrase identification (Wan and Xiao, 2008), as our word graph baselines. Additionally, (#3) the hard paraphrasing (Hard-Para) approach directly substituted words and phrases with their shorter synonyms by using WordNet and PPDB 2.0 (size M is chosen with 463,433 paraphrasing pairs). (#4) Seq2seq model was trained using (B, C) pairs. We considered both of them as comparison approaches as well. The training details are presented in Appendix 1. We release the source code here[8].

4.3 Out-of-Vocabulary (OOV) Word Handling

Both datasets were from the news domain; hence, there are lots of organizations and names that are out of vocabulary. We tackled this problem by exploiting the approach in (Fevry and Phang, 2018).

[8]https://github.com/code4ai/code

Model	METEOR	NN-1	NN-2	NN-3	NN-4	Comp. rate
Ground truth	-	8.6	28.0	40.0	49.1	0.50
#1 WG (Filippova, 10)	0.29	0.0	0.0	2.8	6.8	0.34
#2 KWG (Boudin+, 13)	0.36	0.0	0.0	1.1	3.1	0.52
#3 Hard Para.	0.35	10.1	19.7	29.1	38.0	0.51
#4 Seq2seq with attention	0.33	12.7	24.0	34.7	44.4	0.49
#5 Our rewriter (RWT)	**0.36**	9.0	17.4	25.7	33.8	0.50

Table 2: Results for the Giga-MSC dataset.

Model	METEOR	NN-1	NN-2	NN-3	NN-4	Comp. rate
Ground truth	-	5.2	15.8	23.2	29.6	0.49
#1 WG (Filippova, 10)	0.33	0.0	1.7	5.5	9.8	0.34
#2 KWG (Boudin+, 13)	**0.45**	0.0	1.8	4.6	8.0	0.52
#3 Hard Para.	0.38	9.2	19.0	28.7	37.7	0.50
#4 Seq2seq with attention	0.37	8.4	18.3	27.6	36.3	0.52
#5 Our rewriter (RWT)	0.40	8.1	17.0	26.0	34.3	0.50

Table 3: Results for the Cornell dataset.

Given an input sequence, we first identified all OOV tokens and numbered them in order. We stored the map from the numbered OOV tokens (e.g., OOV1 and OOV2) to words. The corresponding word embeddings were also assigned to each numbered OOV token. We then applied the same numbering system to the target. At the inference, we replaced any output OOV tokens with their corresponding words using the map that was stored beforehand, which allowed us to produce words that were not in the vocabulary.

5 Results and Analysis

METEOR metric (n-gram overlap with synonyms) was used for automatic evaluation. The novel n-gram rate[9] (e.t., NN-1, NN-2, NN-3, and NN-4) was also computed to investigate the number of novel words that could be introduced by the models. Table 2 and Table 3 present the results and below are our observations: (i) keyphrase word graph approach (#2) is a strong baseline according to the METEOR metric. In comparison, the proposed rewriter (#5) yields comparable result on the METEOR metric for the Giga-MSC dataset but lower result for the Cornell dataset. We speculate that it may be due to the difference in the ground-truth compression. 8.6% of novel unigrams exist in the ground-truth compression of the Giga-MSC dataset, while only 5.2% of novel unigrams exist in that of the Cornell dataset, (ii) Hard Para.(#3), Seq2seq (#4), and our rewriter (#5) significantly increase the number of novel n-grams, and the proposed rewriter (#5) seemed to be a better trade-off between the information coverage (measured by METEOR) and the introduction of novel n-grams across all methods, (iii) on comparing with Seq2seq (#4) and our rewriter (#5), we found that adding pseudo data helps to decrease the novel words rate and increase the METEOR score on both datasets.

Method	Informativeness	Grammaticality
KWG	1.06	1.19
RWT	1.02	**1.40**†

Table 4: Human evaluation for informativeness and grammaticality. † stands for significantly better than KWG with 0.95 confidence.

Human Evaluation As METEOR metric cannot measure the grammaticality of compression, we asked two human raters[10] to assess 50 compressed sentences out of the Giga-MSC test dataset in terms of informativeness and grammaticality. We used 0-2 point scale (2 pts: excellent; 1 pts: good; 0 pts: poor), similar to previous work (we recommend readers to refer to Appendix 2 for the 0-2 scale point evaluation details). Table 4 shows the

[9]Novel n-gram rate = $1 - \frac{|S \cap C|}{|C|}$ where S refers to the set of words from all input sentences while C refers set of words from compression.

[10]Both raters are native English speakers and not authors of this paper.

$Sentence_1$	Alleged Russian mobster Alimzhan Tokhtakhounov, accused of conspiring to fix skating events at the 2002 Winter Olympics in salt lake city, has returned to Moscow, the Kommersant daily reported wednesday.
$Sentence_2$	US prosecutors accused Tokhtakhounov of conspiring to fix the artistic skating events at the salt lake city games with the assistance of the French and Russian judges.
KWG	US prosecutors accused Tokhtakhounov, accused of conspiring to fix the artistic skating events at the salt lake city, has returned to Moscow, the Kommersant daily reported wednesday.
RWT	Tokhtakhounov, accused of **conspiracy** to fix the artistic skating events at the salt lake **town**, has returned to Moscow, the Kommersant daily reported.

Table 5: Case study. The words in bold are paraphrase, while the underlined words are ungrammatical parts in the compression. KWG refers to word-graph baseline and RWT refers to our rewriter.

average ratings for informativeness and readability. From that, we found that our rewriter (RWT) significantly improved the grammaticality of compression in comparison with the keyphrase word graph approach, implying that the pseudo data may contribute to the language modeling of the decoder, thereby improving the grammaticality.

Context Awareness Evaluation Because several novel words were introduced in Hard Para. (#3), Seq2seq (#4), and our rewriter (#5), we were interested to determine whether the compressions generated by these models were context-aware. We herein considered an out-of-the-box context-aware encoder, BERT (Devlin et al., 2018). The evaluation proceeded as follows: As for a sentence with N words, $S = [w_1, w_2, ..., w_N]$, we sequentially masked each word at a time and calculated the average likelihood using this formula: $CXT(S) = \frac{1}{n}\sum_{i=1}^{n} -logp(w_i|c)$

where $c = [w_1, ...w_{i-1}, w_{i+1}, ..., w_n]$. We used the implementation mentioned in[11]. The low likelihood $CTX(S)$ may suggest a better context awareness. As presented in Table 6, the proposed rewriter achieves the lowest likelihood on both datasets, thereby indicating better context awareness in its generated compression.

Case Study To illustrate the pros and cons of the proposed rewriter, as listed in Table 5, we conducted a case study where two sentences were given as input and two compression outputs were produced by KWG and RWT. We observed that the RWT corrected the ungrammatical parts (e.t., underlined words,) generated by KWG. However, paraphrasing was not always accurate because phrases such as *salt lake city* are fixed collocations. This may degrade the informativeness of the compression.

[11]https://github.com/xu-song/bert-as-language-model

Method	Giga-MSC		Cornell	
	Base	Large	Base	Large
Hard Para.	354.6	473.6	273.1	316.7
Seq2seq	249.1	219.1	326.1	388.3
Ours	**148.5**	**158.4**	**203.9**	**277.4**

Table 6: Context awareness scores for three models. Base and Large refer to the different model configurations of BERT.

6 Conclusion

In this work, we propose a coarse-to-fine rewriter for multi-sentence compression with a specific focus on improving the quality of compression. The experimental results show that the proposed method produced more grammatical sentences, meanwhile introducing novel words in the compression. Furthermore, we presented an approach for the evaluation of context-awareness which may shed light on automatic evaluation for quality of sentence by virtue of pre-trained models. In the future, we will consider extending the current approach to the single document or multiple document summarization.

Acknowledgments

This study is supported by the Japan Science and Technology Agencys (JST) CREST program JPMJCR1513. We are thankful to the reviewers' helpful comments. We also thank professor McKeown for referring us to their data.

References

Ron Artstein and Massimo Poesio. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4):555–596.

Siddhartha Banerjee, Prasenjit Mitra, and Kazunari Sugiyama. 2015. Multi-document abstractive summarization using ilp based multi-sentence compression. In *IJCAI*, pages 1208–1214.

Regina Barzilay and Kathleen R McKeown. 2005. Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–328.

Florian Boudin and Emmanuel Morin. 2013. Keyphrase extraction for n-best reranking in multi-sentence compression. In *North American Chapter of the Association for Computational Linguistics (NAACL)*.

Trevor Cohn and Mirella Lapata. 2008. Sentence compression beyond word deletion. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 137–144. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Thibault Fevry and Jason Phang. 2018. Unsupervised sentence compression using denoising auto-encoders. *arXiv preprint arXiv:1809.02669*.

Katja Filippova. 2010. Multi-sentence compression: Finding shortest paths in word graphs. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 322–330. Association for Computational Linguistics.

An-Vinh Luong, Nhi-Thao Tran, Van-Giau Ung, and Minh-Quoc Nghiem. 2015. Word graph-based multi-sentence compression: Re-ranking candidates using frequent words. In *Knowledge and Systems Engineering (KSE), 2015 Seventh International Conference on*, pages 55–60. IEEE.

Kathleen McKeown, Sara Rosenthal, Kapil Thadani, and Coleman Moore. 2010. Time-efficient creation of an accurate sentence fusion corpus. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 317–320. Association for Computational Linguistics.

Mir Tafseer Nayeem and Yllias Chali. 2017. Paraphrastic fusion for abstractive multi-sentence compression generation. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 2223–2226. ACM.

Mir Tafseer Nayeem, Tanvir Ahmed Fuad, and Yllias Chali. 2018. Abstractive unsupervised multi-document summarization using paraphrastic sentence fusion. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1191–1204.

Elvys Linhares Pontes, Stéphane Huet, Thiago Gouveia da Silva, Andréa carneiro Linhares, and Juan-Manuel Torres-Moreno. 2018. Multi-sentence compression with word vertex-labeled graphs and integer linear programming. In *Proceedings of the Twelfth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-12)*, pages 18–27.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2015. Improving neural machine translation models with monolingual data. *arXiv preprint arXiv:1511.06709*.

Elaheh ShafieiBavani, Mohammad Ebrahimi, Raymond K Wong, and Fang Chen. 2016. An efficient approach for multi-sentence compression. In *Asian Conference on Machine Learning*, pages 414–429.

Xiaojun Wan and Jianguo Xiao. 2008. Collabrank: towards a collaborative approach to single-document keyphrase extraction. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 969–976. Association for Computational Linguistics.