

Towards Annotating and Creating Sub-Sentence Summary Highlights

Kristjan Arumae[♣], Parminder Bhatia[◇], Fei Liu[♣]

[♣]Computer Science Department, University of Central Florida

[◇]Amazon, USA

{arumae, parmib}@amazon.com feiliu@cs.ucf.edu

Abstract

Highlighting is a powerful tool to pick out important content and emphasize. Creating summary highlights at the sub-sentence level is particularly desirable, because sub-sentences are more concise than whole sentences. They are also better suited than individual words and phrases that can potentially lead to disfluent, fragmented summaries. In this paper we seek to generate summary highlights by annotating summary-worthy sub-sentences and teaching classifiers to do the same. We frame the task as jointly selecting important sentences and identifying a single most informative textual unit from each sentence. This formulation dramatically reduces the task complexity involved in sentence compression. Our study provides new benchmarks and baselines for generating highlights at the sub-sentence level.

1 Introduction

Highlighting at an appropriate level of granularity is important to emphasize salient content in an unobtrusive manner. A small collection of keywords may be insufficient to deliver the main points of an article, while highlighting whole sentences often provide superfluous information. In domains such as newswire, scholarly publications, legal and policy documents (Kim et al., 2010; Sadeh et al., 2013; Hasan and Ng, 2014), people are tempted to write long and complicated sentences. It is particularly desirable to pick out only *important sentence parts* as opposed to whole sentences.

Generating highlights at the sub-sentence level has not been thoroughly investigated in the past. A related thread of research is extractive and compressive summarization (Daumé III and Marcu, 2002; Zajic et al., 2007; Martins and Smith, 2009; Filippova, 2010; Berg-Kirkpatrick et al., 2011; Thadani and McKeown, 2013; Wang et al., 2013; Li et al., 2013, 2014; Durrett et al., 2016).

The methods select representative sentences from source documents, then delete nonessential words and constituents to form compressed summaries. Nonetheless, making multiple interdependent decisions on word deletion can render summaries ungrammatical and fragmented. In this paper, we investigate an alternative formulation that can dramatically reduce the task complexity involved in sentence compression.

We frame the task as jointly selecting representative sentences from a document and identifying a *single* most informative textual unit from each sentence to create sub-sentence highlights. This formulation is inspired by rhetorical structure theory (RST; Mann and Thompson, 1988) where sub-sentence highlights resemble the *nuclei* which are text spans essential to express the writer’s purpose. The formulation also mimics human behavior on picking out important content. If multiple parts of a sentence are important, a human uses a single stroke to highlight them all, up to the whole sentence. If only a part of the sentence is relevant, she only picks out that particular sentence part.

Generating sub-sentence highlights is advantageous over abstraction (See et al., 2017; Chen and Bansal, 2018; Gehrmann et al., 2018; Lebanoff et al., 2018; Celikyilmaz et al., 2018) in several aspects. The highlights can be overlaid on the source document, allowing them to be interpreted in context. The number of highlights is controllable by limiting sentence selection. In contrast, adjusting summary length in an end-to-end, abstractive system can be difficult. Further, highlights are guaranteed to be true-to-the-original, while system abstracts can sometimes “hallucinate” facts and distort the original meaning. Our contributions in this work include the following:

- we introduce a new task formulation of creating sub-sentence summary highlights, then describe

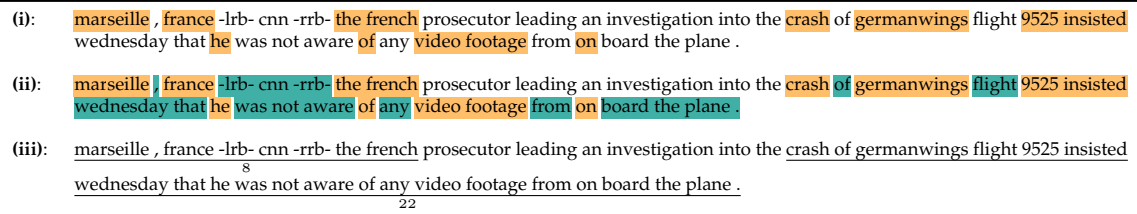


Figure 1: An illustration of label smoothing. Words aligned to the abstract are colored *orange*; gap words are colored *turquoise*.

an annotation scheme to obtain binary sentence labels for extraction, as well as start and end indices to mark the most important textual unit of a positively labeled sentence;

- we examine the feasibility of using neural extractive summarization with a multi-termed objective to identify summary sentences and their most informative sub-sentence units. Our study provides new benchmarks and baselines for highlighting at the sub-sentence level.

2 Annotating Sub-Sentence Highlights

We propose to derive gold-standard sub-sentence highlights from human-written abstracts that often accompany the documents (Hermann et al., 2015). However, the challenge still exists, because abstracts are very loosely aligned with source documents and they contain unseen words and phrases. We define a *summary-worthy sub-sentence unit* as the longest consecutive subsequence that contains content of the abstract. We obtain gold-standard labels for sub-sentence units by first establishing word alignments between the document and abstract, then smoothing word labels to generate sub-sentence labels.

Word Alignment The attention matrix of neural sequence-to-sequence models provides a powerful and flexible mechanism for word alignment. Let $S = \{w_i\}_{i=1}^M$ be a sequence of words denoting the document, and $T = \{w_t\}_{t=1}^N$ denoting the abstract. The attention weight $\alpha_{t,i}$ indicates the amount of attention received by the i -th document word in order to generate the t -th abstract word. All attention values (α) can be automatically learned from parallel training data. After the model is trained, we identify a *single document word* that receives the most attention for generating each abstract word, as denoted in Eq. (1) and illustrated by Figure 1 (i). This step produces a set of source words containing the content of the abstract but possibly with

distinct word forms.¹

$$w_i^{(t)} = \arg \max_{i \in M} \alpha_{t,i} \quad \forall t \quad (1)$$

Smoothing Our goal is to identify sub-sentence units containing content of the abstract by smoothing word labels obtained in the previous step. We extract a single most informative textual unit from a sentence. As a first attempt, we obtain start and end indices of sub-sentence units using heuristics, which are described as follows:

- connecting two selected words if there is a small gap (< 5 words) between them. For example, in Figure 1 (ii), the gap between “*crash*” and “*germanwings*” is bridged by labelling all gap words as selected;
- the longest consecutive subsequence after filling gaps is chosen as the most important unit of the sentence. In Figure 1 (iii), we select the longest segment containing 22 words. When a tie occurs, we choose the segment appearing first;
- creating gold-standard labels for sentences and sub-sentence units. If a segment is the most informative, i.e., longest subsequence of a sentence and > 5 words, we record its start and end indices. If a segment is selected, its containing sentence is labelled as 1, otherwise 0.

2.1 Dataset and Statistics

We conduct experiments on the CNN/DM dataset released by See et al. (2017) containing news articles and human abstracts. We choose the pointer-generator networks described in the same work to obtain attention matrices used for word alignment. The model was trained on the training split of CNN/DM, then applied to all train/valid/test splits to generate gold-standard sub-sentence highlights. At test time, we compare system highlights with

¹ Aligning multiple document words with a single abstract word is possible by retrieving document words whose attention weights exceed a threshold. But the method can be data- and model-dependent, increasing the variability of alignment.

	Sentences		Gold-Standard Highlights			Human Abstracts	
	#TotalSents	%PosSents	#Sents	#Tokens	%CompR	#Sents	#Tokens
Train	5,312,010	24.42	4.51	51.46	0.47	3.68	56.47
Valid	211,022	30.85	4.87	57.11	0.47	4.00	62.73
Test	182,663	29.63	4.72	54.47	0.46	3.79	59.56

Table 1: Data statistics are broken into three categories. *Sentences* indicate the number of total sentences as well as the rate of positive labels. *Gold-Standard Highlights* reflect document-level details of our new ground truth labels. Compression rate (“CompR”) indicates the percentage of a positive labeled sentence was covered by the segment. Finally *Human Abstracts* provides a comparison against CNN/DailyMail ground truth summaries.

gold-standard highlights and human abstracts, respectively, to validate system performance.

In Table 1, we present data statistics of the gold-standard sub-sentence highlights. We observe that gold-standard highlights and human abstracts are of comparable length in terms of tokens. On average, 28% of document sentences are labelled as positive. Among these, 47% of the words belong to gold-standard sub-sentence highlights. In our processed dataset we retain important document level information such as original sentence placement and document ID. We consider each document sentence as a data instance, and introduce a neural model to predict (i) a binary sentence level label, and (ii) start and end indices of a consecutive subsequence for a positive sentence. We are particularly interested in predicting start and end indices to encourage sub-sentence segments to remain self-contained. Finally, we leverage the document ID to re-combine model output to still generate summaries at the document level.

3 Models

We provide initial modeling for our data with a single state-of-the-art architecture. The purpose is to build meaningful representations that allow for joint prediction of summary-worthy sentences and their sub-sentence units. Our model receives an input sequence as an individualized sentence denoted as $S = \{w_i^s\}_{i=1}^M$, where s denotes the sentence index in the original document. The model learns to predict the sentence label and start/end index of a sub-sentence unit based on contextualized representations.

For each token w_i^s we leverage a combined representation E_{tok} , $E_{s\text{-pos}}$, and $E_{d\text{-pos}}$, i.e., a token embedding, sentence level positional embedding, and a document level positional embedding. Here $s\text{-pos}$ denotes the token position in a sentence, $d\text{-pos}$ denotes the sentence position in a document, and $E(w_i^s) \in \mathbb{R}^d$. We justify the last embedding by

noting that the sentence position within that document plays an important role since generally there is a higher probability of positive labels towards the beginning. The final input representation is an element-wise addition of all embeddings (Eq. (2)). This input is encoded using a bi-directional transformer (Vaswani et al., 2017; Devlin et al., 2018), denoted as \mathbf{h} .

$$E(w_i^s) := E_{\text{tok}}(w_i^s) + E_{s\text{-pos}}(w_i^s) + E_{d\text{-pos}}(w_i^s) \quad (2)$$

3.1 Objectives

We use the transformer output to generate three labels: sentence, start and end positions of the sub-sentence unit. First we obtain the sequence representation via the [CLS] token.² We apply a linear transformation to this vector and a softmax layer to obtain a binary label for the entire sentence.

For the indexing objective we transform the encoder output, \mathbf{h} , to account for start and end index classification. $\mathbf{a} = \text{MLP}_{\text{start/end}}(\mathbf{h}) \in \mathbb{R}^{M \times 2}$. Again we make use of a single linear transformation, here it is applied across the encoder temporally giving each time-step two channels. The two channels are individually passed through a softmax layer to produce two distributions, for the start and end index. Finally we use a combined loss term which is trained end-to-end using a cross entropy objective:

$$\mathcal{L} = \lambda(\mathcal{L}_{\text{start}} + \mathcal{L}_{\text{end}}) + \mathcal{L}_{\text{sent}}. \quad (3)$$

For negatively labeled sentences $\mathcal{L}_{\text{start}}$ and \mathcal{L}_{end} are not utilized during training. λ is a coefficient balancing between two task objectives.

3.2 Experimental Setup

The encoder hidden state dimension is set at 768, with 12 layers and 12 attention heads (BERT_{BASE} uncased). We utilize dropout (Srivastava et al., 2014) with $p = 0.1$, and λ is empirically set to 0.1.

²[CLS] is fine-tuned as a class label for the entire sequence, and always positioned at \mathbf{h}_1

Model	ROUGE-1			ROUGE-2			ROUGE-L			
	P	R	F ₁	P	R	F ₁	P	R	F ₁	
Oracle (sent.)	36.63	69.52	46.58	20.24	37.76	25.55	25.59	47.84	32.34	
Oracle (segm.)	59.71	50.95	53.82	34.42	29.60	31.16	43.23	36.89	38.95	
Pointer Gen. (See et al., 2017)	–	–	39.53	–	–	17.28	–	–	36.38	
QASumm+NER (Arumae and Liu, 2019)	–	–	25.89	–	–	11.65	–	–	22.06	
ABSTRACT	Sent	30.91	48.61	34.84	13.31	21.40	15.09	20.14	31.44	22.55
	Sent + posit.	31.31	56.53	37.72	14.45	26.70	17.53	20.51	37.05	24.63
	Segm	32.58	44.97	34.73	13.79	19.36	14.75	21.36	29.03	22.51
	Segm + posit.	33.11	52.74	37.99	14.96	24.30	17.26	21.69	34.41	24.75
SUB-SENT	Sent	38.93	58.49	42.81	28.88	44.49	31.96	32.92	50.14	36.32
	Sent + posit.	39.97	68.59	47.02	31.38	55.31	37.19	34.58	60.30	40.86
	Segm	41.31	54.27	42.83	30.29	40.38	31.43	34.81	46.01	36.07
	Segm + posit.	42.43	64.09	47.43	32.75	50.40	36.76	36.43	55.58	40.80

Table 2: ROUGE results on CNN/DM test set at both sentence and sub-sentence level. The top two rows test gold-standard sentences and sub-sentences against human abstracts. Additionally we show results of an abstractive (See et al., 2017) and an extractive summarizer (Arumae and Liu, 2019) whose CNN/DM results are macro-averaged. The bottom two sections showcase our models. We report results at sentence and sub-sentence level and report those with and without E_{d-pos} embeddings (+posit.). These results are further broken down to reflect evaluation against human abstracts and our own gold standard segments.

We use Adam (Kingma and Ba, 2014) as our optimizer with a learning rate of $3e^{-5}$, and implement early stopping against the validation split. Devlin et al. (2018) suggest that fine-tuning takes only a few epochs with large datasets. Training was conducted on a GeForce GTX 1080 Ti GPU, and each model took at most three days to converge with a maximum epoch time of 12 hours.

At inference time we only extract start and end indices when the sentence label is positive. Additionally if the system produced an end index occurring before the start index we ignore it and select the argmax of the distribution for end indexes which are located after the start index.

4 Results

In Table 2 we report results on the CNN/DM test set evaluated by ROUGE (Lin, 2004). We examine to what extent our summary sentences and sub-sentence highlights, annotated using the strategy presented in §2, have matched the content of human abstracts. These are the *oracle* results for sentences and segments, respectively. Despite that abstracts can contain unseen words, we observe that 70% of the abstract words are covered by gold-standard sentences, and 51% of abstract words are included in sub-sentence units, suggesting the effectiveness of our annotation method on capturing summary-worthy content.

We proceed by evaluating our method against state-of-the-art extractive and abstractive summarization systems. Arumae and Liu (2019) present an approach to extract summary segments using

question-answering as supervision signal, assuming a high quality summary can serve as document surrogate to answer questions. See et al. (2017) present pointer-generator networks, an abstractive summarization model and a reliable baseline for being both state-of-the-art, and also a vital tool for guiding our data creation. We show that the performance of oracle summaries is superior to these baselines in terms of R-2, with sub-sentence highlights achieving the highest R-2 F-score of 31%, suggesting extracting sub-sentence highlights is a promising direction moving forward.

4.1 Modeling

Our models are shown in the bottom two sections of Table 2. We obtain system-predicted whole sentences (*Sent*) and sub-sentence segments (*Segm*); then evaluate them against both human abstracts (ABSTRACT) and gold-standard highlights (SUB-SENT). We test the efficacy of document positional embeddings (Eq. (2)), denoted as +posit.

Using R-2 as a defining metric, our model outperforms or performs competitively with both the abstractive and extractive baselines. We find that the use of document level positional embeddings is beneficial and that for both summary types, models with these embeddings have a competitive edge against those without. Notably sub-sentence level ROUGE scores consistently outmatch sentence level values. These results are nontrivial, as segment level modeling is highly challenging, often resulting in increased precision but drastically reduced recall (Cheng and Lapata, 2016).

Our model (+*posit*) positively labeled 22.27% of sentences, with an average summary length of 3.54 sentences. The segment model crops selected sentences, exhibiting a compression ratio of 0.77. Comparing to gold-standard ratio of 0.47, there is a 67.4% increase, pointing to future work on highlighting sub-sentence segments.

5 Conclusion

We introduced a new task and dataset to study sub-sentence highlight extraction. We have shown the dataset provides a new upper bound for evaluation metrics, and that the use of sub-sentence segments provides more concise summaries over full sentences. Furthermore, we evaluated our data using a state-of-the-art neural architecture to show the modeling capabilities using this data.

Acknowledgments

We thank the anonymous reviewers for their valuable suggestions. This research was supported in part by the National Science Foundation grant IIS-1909603.

References

- Kristjan Arumae and Fei Liu. 2019. [Guiding extractive summarization with question-answering rewards](#). In *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL)*.
- Taylor Berg-Kirkpatrick, Dan Gillick, and Dan Klein. 2011. [Jointly learning to extract and compress](#). In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Asli Celikyilmaz, Antoine Bosselut, Xiaodong He, and Yejin Choi. 2018. [Deep communicating agents for abstractive summarization](#). In *Proceedings of the North American Chapter of the Association for Computational Linguistics (NAACL)*.
- Yen-Chun Chen and Mohit Bansal. 2018. [Fast abstractive summarization with reinforce-selected sentence rewriting](#). In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Jianpeng Cheng and Mirella Lapata. 2016. [Neural summarization by extracting sentences and words](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 484–494.
- Hal Daumé III and Daniel Marcu. 2002. [A noisy-channel model for document compression](#). In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.
- 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*.
- Greg Durrett, Taylor Berg-Kirkpatrick, and Dan Klein. 2016. [Learning-based single-document summarization with compression and anaphoricity constraints](#). In *Proceedings of the Association for Computational Linguistics (ACL)*.
- Katja Filippova. 2010. [Multi-sentence compression: Finding shortest paths in word graphs](#). In *Proceedings of the International Conference on Computational Linguistics (COLING)*.
- Sebastian Gehrmann, Yuntian Deng, and Alexander M. Rush. 2018. [Bottom-up abstractive summarization](#). In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- Kazi Saidul Hasan and Vincent Ng. 2014. [Automatic keyphrase extraction: A survey of the state of the art](#). In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. [Teaching machines to read and comprehend](#). In *Proceedings of Neural Information Processing Systems (NIPS)*.
- Su Nam Kim, Olena Medelyan, Min-Yen Kan, and Timothy Baldwin. 2010. [SemEval-2010 task 5: Automatic keyphrase extraction from scientific articles](#). In *Proceedings of the 5th International Workshop on Semantic Evaluation*.
- Diederik P Kingma and Jimmy Ba. 2014. [Adam: A method for stochastic optimization](#). *arXiv preprint arXiv:1412.6980*.
- Logan Lebanoff, Kaiqiang Song, and Fei Liu. 2018. [Adapting the neural encoder-decoder framework from single to multi-document summarization](#). In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- Chen Li, Fei Liu, Fuliang Weng, and Yang Liu. 2013. [Document summarization via guided sentence compression](#). In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- Chen Li, Yang Liu, Fei Liu, Lin Zhao, and Fuliang Weng. 2014. [Improving multi-document summarization by sentence compression based on expanded constituent parse tree](#). In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.
- Chin-Yew Lin. 2004. [Rouge: A package for automatic evaluation of summaries](#). *Text Summarization Branches Out*.

- William C. Mann and Sandra A. Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.
- Andre F. T. Martins and Noah A. Smith. 2009. Summarization with a joint model for sentence extraction and compression. In *Proceedings of the ACL Workshop on Integer Linear Programming for Natural Language Processing*.
- Norman Sadeh, Alessandro Acquisti, Travis Breaux, Lorrie Cranor, Aleecia McDonald, Joel Reidenberg, Noah Smith, Fei Liu, Cameron Russel, Florian Schaub, and Shomir Wilson. 2013. The usable privacy policy project: Combining crowdsourcing, machine learning and natural language processing to semi-automatically answer those privacy questions users care about. Technical Report CMU-ISR-13-119, Carnegie Mellon University.
- Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.
- Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.
- Kapil Thadani and Kathleen McKeown. 2013. Sentence compression with joint structural inference. In *Proceedings of CoNLL*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.
- Lu Wang, Hema Raghavan, Vittorio Castelli, Radu Florian, and Claire Cardie. 2013. A sentence compression based framework to query-focused multi-document summarization. In *Proceedings of ACL*.
- David Zajic, Bonnie J. Dorr, Jimmy Lin, and Richard Schwartz. 2007. Multi-candidate reduction: Sentence compression as a tool for document summarization tasks. *Information Processing and Management*.