Summarizing Summarization at ACL 2019

Thomas Scialom
LIP6 - Sorbonne Universités
reciTAL
thomas@recital.ai

Jacopo Staiano
reciTAL
jacopo@recital.ai

Abstract

Several papers dealing with Summarization were presented at ACL 2019. We tried to make a summary of them. As we publish this while ACL 2019 is still on, please use this document responsibly: it might contain (slight, hopefully) imprecisions, and some papers are still in the backlog. This will turn into a markdown soon, so you can easily PR etc.

1 New Data, More Data

Big Patent Sharma et al. (2019) introduced a novel dataset, consisting of 1.3 million records of U.S. patent documents along with human written abstractive summaries.

Characteristics:

- summaries contain a richer discourse structure with more recurring entities;
- longer input sequences (avg. 3,572.8 VS 789.9 words for CNN/DM);
- salient content is evenly distributed in the input, while in popular news-based datasets it often concentrates in the first few sentences;
- fewer and shorter extractive fragments are present in the summaries.

The authors report results for various extractive and abstractive models on CNN/DM, NYT, and Big Patent. What seems really interesting is the divergence of results: PointGen (See et al., 2017) compared favorably against the extractive unsupervised model Text-Rank (Mihalcea and Tarau, 2004) on the news-based dataset, while obtaining worse results on Big Patent. This shows once more the importance of testing models on several and different datasets.

Multi-News Fabbri et al. (2019) presented the first multi-document summarization dataset, based on news. It consists of input articles from over 1,500 different websites along with 56,216 professional summaries of these articles obtained from the site newser.com. Additionally, the authors propose an end-to-end model that achieves competitive results under both automatic and human evaluation on various multi-document datasets, including Multi-News.

2 Multimodal Summarization

Talk-Summ Thanks to the recent trend of publishing videos of talks in academic conferences, Lev et al. (2019) collected 1716 pairs of papers/videos and consider the video transcripts as the summary for the related paper. The proposed method of generating training data for scientific papers summarization is fully automatic. Hence, the number of training data might benefit directly from the increasing rate of papers published in a near future without too much effort required. This would be definitely something help full in order to follow the impressive rate of publication in NLP and other scientific fields!

(Palaskar et al., 2019) The authors explored the behaviour of various models for video summarization on the How2 dataset. They proposed different text and vision modalities for the input such as the automatic transcript, the audio and the video latent representations and combinations of them through hierarchical attention. For the evaluation, in addition to ROUGE, the authors proposed a variant that doesn't take account of the stop words. Interestingly, one of the presented models include a video-only input summarization model that performs competitively with a text-only model.

3 Extractive models

(Cho et al., 2019) The authors propose to tackle multi-document summarization with Determinantal Point Processes (DPP), a learned extractive method and capsule network (Sabour et al., 2017) components. Motivation: TF-IDF vectors fall short when it comes to modeling semantic similarity, a fact that is particularly problematic for multi-document summarization. Solution: a similarity measure for pairs of sentences such that semantically similar sentences can receive high scores despite having very few words in common. The capsule network is trained under a binary classification setup, on a dataset derived from CNN/DM: the authors map abstract sentences to the most similar article sentences (label=true) and negative sampling (label=false).

(Wang et al., 2019a) A method to train an extractive model in a self-supervised fashion: the sentence encoder is first trained to learn entailment w.r.t. to the next sentence, replacement and switch of the next sentence. It allows to train faster and to obtain a slight improvement on CNN/DM. The proposed method could also lead to longer text representations in a self-supervised manner.

(Nishida et al., 2019) The study focuses on HotpotQA, a multi-hop QA explainable task: the system return the answer with the evidence sentences by reasoning and gathering disjoint pieces of the reference texts gathering. The Query Focused Extractor (QFE) is inspired by the extractive summarization model proposed in (Chen and Bansal, 2018). Instead of covering the important information in the source document with the extractive summary, the approach covers the question with the extracted evidences. The model compared favorably with the SOTA BERT-based model in HotpotQA distractor setting for retrieving the evidence while not benefiting from any pretraining. In addition, it achieves SOTA performance on FEVER dataset (Thorne et al., 2018).

(Zheng and Lapata, 2019) The authors revisited classic extractive unsupervised summarization using graph-based ranking approaches, where the nodes are the sentences of a document. They leverage BERT to encode each sentence. One of motivation is that popular supervised approaches are limited by the need of large-scale datasets and thus does not generalize well to other domain and

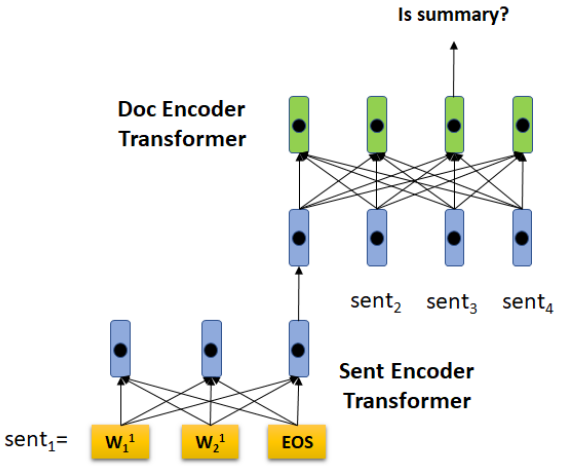

Figure 1: HIBERT architecture

languages. The model performs comparably well to SOTA approaches on the popular CNN/DM and NYT datasets, as well as for TTNews, a Chinese news summarization corpus showing its capability to adapt well to different domain. A human assessment is conducted, based on a set of question posed on the gold summary, evaluating how much relevant information is present in a generated evaluated summary. The application to sentence selection in multi-document summarization is suggested as future work.

(Zhang et al., 2019) HIBERT stands for Hierarchical BERT. The idea is to use two pretrained transformers (see figure 1): the first, a standard BERT at token level used to represent the sentences; the second, working at sentence level and leveraging on the representation from the former to encode sentences of an entire document. Following the BERT masked pretraining method, the authors trained the sentence level transformer masking some sentences of the documents, and the final model achieves SOTA for summarization on CNN/DM and NY times datasets. The authors also report informative ablations, using out-of-domain, in-domain data, and a combination thereof for the pretraining. Cherry on the cake, they adapt BERT to extractive supervised summarization (i.e. finetuning BERT in a classification setup to select the sentences to extract) and report the result as a baseline.

4 Abstractive models

(Lebanoff et al., 2019) Abstractive summarizers tend to perform content selection and fusion

implicitly by learning to generate the text in an end-to-end manner. The proposed method, instead, summarizes documents in a two stages process, the first extractive and the second abstractive. The motivation is that separating the summarization process into two explicit steps could allow for more flexibility and explainability for each components. The extractive stage is done using BERT representations. The extracted sentence singletons are then fed into a sequence to sequence model to generate the summary. Evaluations are reported for both the extractive methods and the full pipeline on three datasets (CNN/DM, DUC and Xsum).

(Liu and Lapata, 2019) In the original WikiSum paper, the authors proposed a two stages process, first extracting the most important sentences from all the documents (Mihalcea and Tarau, 2004) in order to get a shorter input, then learning to generate the output with a transformer model. On top of that, (Liu and Lapata, 2019) propose to refine the extractive step with a hierarchical representation of the documents using attention instead of just concatenating the extracted sentences.

(Wang et al., 2019b) Bi-directional Selective Encoding with Template (Biset) is a new architecture for abstractive summarization tested on the Gigawords dataset. Template-based summarization relies on manual creation of templates. The advantage of such approach is that it results in concise and coherent summaries without requiring training data. However, it requires experts to build these templates. In this paper, an automatic method is proposed to retrieve high-quality templates from training corpus. Given an input article, the model first retrieves the most similar articles using a TF-IDF-based method. Further, a similarity measure is computed through a neural network in order to re-rank the retrieved articles. The summary corresponding to the most similar article to the input is then selected as the template. Finally a sequence to sequence network is trained to generate the summary: the authors propose an architecture to learn the interaction between the source summary and the selected template.

(Perez-Beltrachini et al., 2019) Most previous works on neural text generation represent the target summaries as a single long sequence. Assuming that documents are organized into topically coherent text segments, the authors propose a hierarchical model that encodes both documents and sentences guided by the topic structure of target summaries. The topic templates from summaries are obtained via a trained Latent Dirichlet Allocation model (Blei et al., 2003). WikiCat-Sum, the dataset used for evaluation is derived from WikiSum (Liu et al., 2018), and focuses on three domains: Companies, Films, and Animals. The dataset is publicly available [1].

(Makino et al., 2019) Most abstractive summarization models do not control the length of the generated summary and learn it from the distribution of the examples seen during training. The authors propose an optimization method under a length constraint. They report extensive experiments on CNN/DM using several models with different length constraints and optimization methods. In addition to ROUGE and length control, the authors report the average generation time, along with a human assessment.

5 Evaluation

(Hardy et al., 2019) Automatic summarization evaluation is an open research question and the current methods have several pitfalls. For this reason, most of the papers conduct human evaluations, a challenging and time consuming task. (Hardy et al., 2019) propose a new human evaluation methodology: first, a group of annotators highlight the salient content in the input article. Then, other annotators are asked to score for precision (i.e. only important information is present in the summary), recall (all important information is present in the summary) and linguistic metrics (clarity and fluency). Major advantages of this method:

- highlights are not dependent on the summaries being evaluated but only on the source documents, thus avoiding reference bias;
- it provides absolute instead of ranked evaluation allowing for better interpretability;
- the highlight annotation needs to happen only once per document, and it can be reused to evaluate many system summaries.

[1] https://github.com/lauhaide/WikiCatSum

Figure 2: Figure 2 of (Hardy et al., 2019) The UI for content evaluation with highlight.

Finally, the authors propose a version of ROUGE leveraging the highlight annotations. The UI (see fig 2) is open source [2].

(Peyrard, 2019) In this work, the author formalized several summary-related such as redundancy, relevance, and informativeness and propose a notion unifying these concepts called *importance*. Beyond theory, the author report experiments where the proposed definition are applied showing a correlation with human judgments. It will be interesting in the future to see models leveraging this framework.

[2] https://github.com/sheffieldnlp/highres

References

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Yen-Chun Chen and Mohit Bansal. 2018. Fast Abstractive Summarization with Reinforce-Selected Sentence Rewriting. *arXiv:1805.11080 [cs]*. ArXiv: 1805.11080.

Sangwoo Cho, Logan Lebanoff, Hassan Foroosh, and Fei Liu. 2019. Improving the Similarity Measure of Determinantal Point Processes for Extractive Multi-Document Summarization. *arXiv:1906.00072 [cs]*. ArXiv: 1906.00072.

Alexander R. Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir R. Radev. 2019. Multi-News: a Large-Scale Multi-Document Summarization Dataset and Abstractive Hierarchical Model. *arXiv:1906.01749 [cs]*. ArXiv: 1906.01749.

Hardy, Shashi Narayan, and Andreas Vlachos. 2019. HighRES: Highlight-based Reference-less Evaluation of Summarization. *arXiv:1906.01361 [cs]*. ArXiv: 1906.01361.

Logan Lebanoff, Kaiqiang Song, Franck Dernoncourt, Doo Soon Kim, Seokhwan Kim, Walter Chang, and Fei Liu. 2019. Scoring Sentence Singletons and Pairs for Abstractive Summarization. *arXiv:1906.00077 [cs]*. ArXiv: 1906.00077.

Guy Lev, Michal Shmueli-Scheuer, Jonathan Herzig, Achiya Jerbi, and David Konopnicki. 2019. TalkSumm: A Dataset and Scalable Annotation Method for Scientific Paper Summarization Based on Conference Talks. *arXiv:1906.01351 [cs]*. ArXiv: 1906.01351.

P. J. Liu, M. Saleh, E. Pot, B. Goodrich, R. Sepassi, L. Kaiser, and N. Shazeer. 2018. Generating Wikipedia by Summarizing Long Sequences. *arXiv*.

Yang Liu and Mirella Lapata. 2019. Hierarchical Transformers for Multi-Document Summarization. *arXiv:1905.13164 [cs]*. ArXiv: 1905.13164.

Takuya Makino, Tomoya Iwakura, Hiroya Takamura, and Manabu Okumura. 2019. Global Optimization under Length Constraint for Neural Text Summarization. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, pages 1039–1048, Florence, Italy. Association for Computational Linguistics.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*.

Kosuke Nishida, Kyosuke Nishida, Masaaki Nagata, Atsushi Otsuka, Itsumi Saito, Hisako Asano, and Junji Tomita. 2019. Answering while Summarizing: Multi-task Learning for Multi-hop QA with Evidence Extraction. *arXiv:1905.08511 [cs]*. ArXiv: 1905.08511.

Shruti Palaskar, Jindrich Libovick, Spandana Gella, and Florian Metze. 2019. Multimodal Abstractive Summarization for How2 Videos. *arXiv:1906.07901 [cs]*. ArXiv: 1906.07901.

Laura Perez-Beltrachini, Yang Liu, and Mirella Lapata. 2019. Generating Summaries with Topic Templates and Structured Convolutional Decoders. *arXiv:1906.04687 [cs]*. ArXiv: 1906.04687.

Maxime Peyrard. 2019. A Simple Theoretical Model of Importance for Summarization. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, pages 1059–1073, Florence, Italy. Association for Computational Linguistics.

Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. 2017. Dynamic routing between capsules. In *Advances in neural information processing systems*, pages 3856–3866.

A. See, P. J. Liu, and C. D. Manning. 2017. Get To The Point: Summarization with Pointer-Generator Networks. *arXiv:1704.04368*.

Eva Sharma, Chen Li, and Lu Wang. 2019. BIGPATENT: A Large-Scale Dataset for Abstractive and Coherent Summarization. *arXiv:1906.03741 [cs]*. ArXiv: 1906.03741.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for Fact Extraction and VERification. *arXiv:1803.05355 [cs]*. ArXiv: 1803.05355.

Hong Wang, Xin Wang, Wenhan Xiong, Mo Yu, Xiaoxiao Guo, Shiyu Chang, and William Yang Wang. 2019a. Self-Supervised Learning for Contextualized Extractive Summarization. *arXiv:1906.04466 [cs]*. ArXiv: 1906.04466.

Kai Wang, Xiaojun Quan, and Rui Wang. 2019b. BiSET: Bi-directional Selective Encoding with Template for Abstractive Summarization. *arXiv:1906.05012 [cs]*. ArXiv: 1906.05012.

Xingxing Zhang, Furu Wei, and Ming Zhou. 2019. HIBERT: Document Level Pre-training of Hierarchical Bidirectional Transformers for Document Summarization. *arXiv:1905.06566 [cs]*. ArXiv: 1905.06566.

Hao Zheng and Mirella Lapata. 2019. Sentence Centrality Revisited for Unsupervised Summarization. *arXiv:1906.03508 [cs]*. ArXiv: 1906.03508.