Improve Query-Focus Summarization using NLI

Chao Zhai, Dat Nguyen, Hitesh Manivannan, Lihan Zhu, Zhongheng He
Viterbi School of Engineering
University of Southern California
{chaozhai, ndat,
hmanivan, lihanzhu,
hezhongh}@usc.edu

[1]

1 Introduction

Query-focused summarization (QFS) is a special type of text summarization which focuses on generating summaries conditioned to a specific user's query. In this setting, totally different summaries can be gene-rated from a context given different queries. QFS makes it possible for users to explicitly specify their need and preference in the context, but also brings challenges to model the relation between context and query, as well as the evaluation of the generated summaries.

Like general summarization, the methods of QFS are divided into two different parts: extractive QFS and abstractive QFS. Due to the lack of training data, extractive QFS are first explored in this area (Litvak and Vanetik, 2017; Egonmwan et al., 2019), which make use of query-related frequent word sets, sentence ranking algorithms et al. On the other hand, abstractive QFS (Xu and Lapata, 2020; Su et al., 2020; Su et al., 2021) often introduce signals from answer relevance scores from extractive question answering model or fine-tune pretrained general summarization model. The far supervision signals or knowledge transfer are popular because large-size training data is limitedly available. However, the signals are either used as part of input in self-attention architecture or used to re-rank and filter the sentences in the source documents. Limited roles are played in the gradient propagation.

Inspired by work of Chen et al (2021), we try to introduce signals from natural language inference (NLI) task. NLI is the task of determining the inference relation between two texts. We use NLI to generate an evaluation score for the generated summary given the context and specific query, and use the score as a supervision signal when training the QFS model. In this way, we can not only 1. leverage the knowledge in NLI to model, to measure the relationship between the input (context and query) and output (summary), but also 2. make this signal participate in the loss computing and gradient propagation process, bring stronger supervision information compared to previous works.

Document: Interrogator Ali Soufan said in an April op-ed article in the New York Times: "It is inaccurate to say that Abu Zubaydah had been uncooperative [and that enhanced interrogation techniques supplies interrogators with previously unobtainable information]. Along with another f.b.i. agent and with several c.i.a. officers present I questioned him from March to June before the harsh techniques were introduced later in August. Under traditional interrogation methods he provided us with important actionable intelligence."
Query: Are traditional interrogation methods insufficient?
Summary: The same info can be obtained by traditional interrogations.

Table 1 An example of QFS

2 Related Work

2.1 Query-Focus Summarization

Compared to general summarization, Query-Focus Summarization is a more complex task that aims to extract specific content from original document conditioned on a user's query. Thus, researchers often combined the traditional summarization models with supervised signals from similar task or made use of knowledge transfer.

Recently, transformer-based (Vaswani et al., 2017) models pretrained on huge number of corpuses have dominated natural language processing tasks, both classification and generation. Among them, BART (Lewis et al., 2019) and its variants became

[1] Table is cited from Dan Su, Tiezheng Yu, and Pascale Fung. 2021. Im- prove query focused abstractive summarization by incorporating answer relevance.

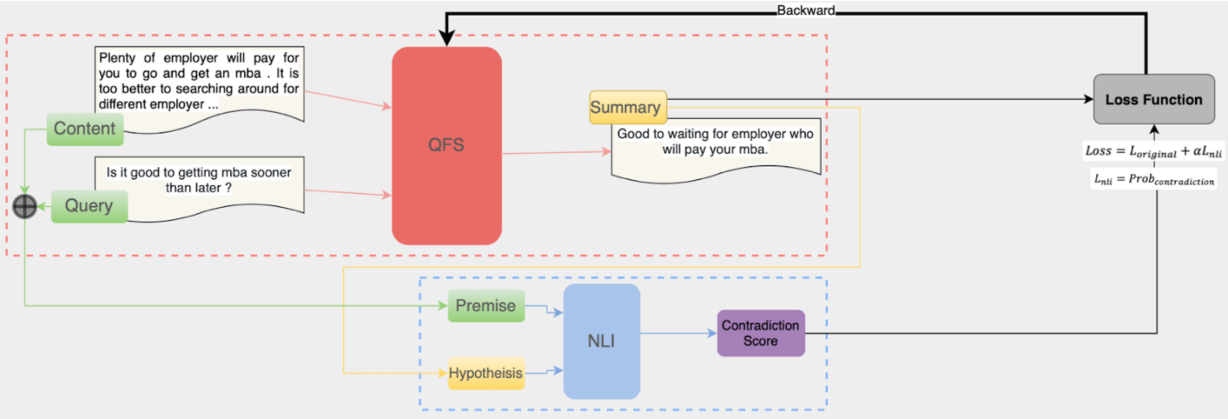
Backward
Plenty of employer will pay for you to go and get an mba . It is too better to searching around for different employer ...
Content
Is it good to getting mba sooner than later ?
Query
QFS
Summary
Good to waiting for employer who will pay your mba.
Loss Function
$Loss = L_{original} + \alpha L_{nli}$
$L_{nli} = Prob_{contradiction}$
Premise
Hypotheisis
NLI
Contradiction Score

Figure 1 Our Proposed Method. On the top is the traditional QFS model, with the concatenation of content and query as inputs. At the bottom is NLI module we add, to use content and query as premise, summary as hypothesis. After computing the probability of contradiction, use it as additional loss combined with original loss.

the backbone model for many language generation tasks. Su et al. built a pipeline system containing a question answer module and a query-focus multi-document summarization module to do information extraction on Covid-19. They used QA module to extract related paragraphs to a specific query, and used those paragraphs and query as input to do query-focus summarization using BART. Xu and Lapata tried to introduce signals from QA tasks, due to limited training data in QFS. They first finetuned a QA model that can selected answer span in a given sentence based on a query and then use the model to filter the sentences in the source document. Finally, they used LEXRANK algo., combined with the scores from QA models, to determine which sentences should be kept as the summary. Furthermore, Su et al. try to incoperate the scores from QA models into the self-attention module in transformer architecture to generate query-related summary. In details, they get score for each token in source document using HLTC-MRQA (Su et al., 2019), and used the scores as input when computing attention scores.

2.2 Natural Language Inference

Natural language inference (NLI) is a popular task in NLP, aiming to determine whether a premise can entail, contradict, or know nothing about a hypothesis. As mentioned above, NLI is also dominated by pretrained language models (Devlin et al., 2018; Liu et al., 2019; Yang et al., 2019; He et al., 2020).

NLI also showed huge potential to be used as signal in other natural language tasks. Chen et al. explored to use NLI model to verify QA systems, and also use it to improve model calibration and transferring ability of QA models. Falke et al. used NLI model to verify the correctness of the sentences in generated summaries and re-rank the candidate summaries. Li et al. tried to incorporate entailment knowledge into abstractive summarization models, and propose an entailment-aware encoder and decoder.

3 Methods

The key idea of our method is adding NLI's score as a loss in our training process. It's a more effective way to cooperate with the signals from other tasks.

In previous work, some researchers used an extractive model to filter the sentences in context. Some researchers used QA models to generate a score for each word in the context, measuring its relevance with the query, and used these scores as other inputs for attention module of Transformers. Some researchers simply fine-tune the general summarization model using context + [SEP] + query as inputs. These methods used some signals from other task or do some knowledge transfer, but they are too weak to influence the original model and make good use of the relationship of the [context, query, summary] tuple.

Thus, we design a new loss using NLI to strengthen signals.

$$L_{nli} = Prob_{contradiction}$$

$$Loss = L_{original} + \alpha L_{nli}$$

The loss is computed from the NLI model using (context + query -> summary). The loss can effectively measure the relationship of the [context, query, summary] tuple. Instead of acting as other inputs, these scores can join in the gradient computation and back-propagation. So it can lead to model to coverage in a correct direction in both 1. Generating good summary (with original loss) and 2. Consider the relationship of the [context, query, summary] tuple (with NLI loss).

Dataset	Model	Rouge-1	Rouge-2	Rouge-L
Debatepedia	Baseline	57.1623	45.3881	55.8273
	NLI loss	**57.3192**	**45.6343**	**55.9098**
QMSum	Baseline	31.6814	10.4602	21.0466
	NLI loss	**31.9101**	**10.8455**	**21.6838**

Table 3 Rouge scores for Debatepedia and QMSum, from baseline model and out method

4 Experiment

4.1 Dataset

We used Debatepedia and QMSum datasets to evaluate our method. Previous work in abstractive QFS (Nema et al., 2017) introduced a new dataset Debatepedia. The dataset was constructed from an encyclopedia of pro and con arguments and quotes on critical debate topics. It contains 663 debates, from which 12695 {query, document, summary} triples are put into a dataset. And QMSum was introduced as a human-annotated benchmark for query-based multidomain meeting summarization tasks (Zhong et al., 2021). It consists of 1,808 query-summary pairs over 232 meetings in multiple domains. Table 2 shows some statistics of these two datasets.

Dataset	Document	Query	Summary
Debatepedia	66.4	9.97	11.16
QMSum	9590	11	70

Table 2 Average length of the documents, query and summary in Debatepedia and QMSum dataset

4.2 Baseline Model

The current state-of-the-art for the debatepedia dataset is QFS-BART (Dan su et al.) which we used as our baseline to evaluate the impact of introducing NLI into training. Using language models for domain specific tasks like summarization (Liu et al) have been extensively studied. QFS-BART improves on existing abstractive summarization models that use BART by incorporating answer relevance scores of HLTC-MRQA (Su et al., 2019). This is done by combining the relevance scores as explicit attention in the attention layers of all the Transformer decoder layers

QFS-BART has a two-stage fine-tuning process where they first directly train BART on the XSUM dataset (Shashi et al) and in the second stage, they finetune it on the QFS dataset. Using the hyperparameters specified, we achieve similar performances with the paper on the debatepedia dataset. We also finetune the model on QMSum separately to get the rogue scores on the dataset.

We adopt the Rogue score (Lin et al, 2004) which is the gold standard for evaluating the performances of summarization models

For NLI model, we used roberta-large-mnli[2] in Hugging Face[3].

5 Results

Experiment results are presented in Table 3. By comparing the performances of the Baseline versus our model with NLI-Loss incorporated into it, we notice a marginal improvement in Rogue-1, Rogue-2 and Rogue-L scores. We also look at the learning curves of rogue scores for the validation set and judge the smoothness of the curve to evaluate the changes in the consistency of the learning process for using NLI.

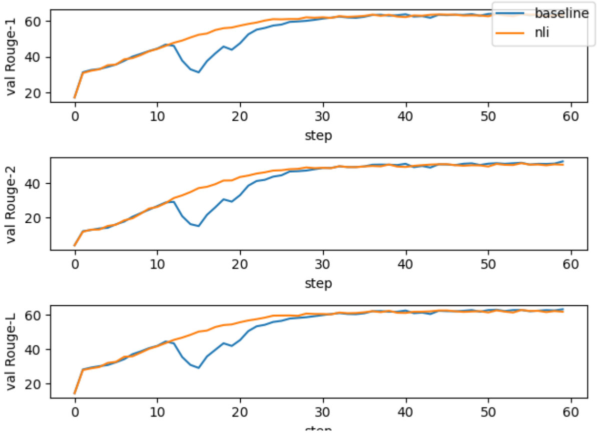

Figure 2 Rouge scores learning curve of valid set of Debatepedia.

[2] https://huggingface.co/roberta-large-mnli

[3] https://huggingface.co/

Models	**Debatepedia**			**QMSum**		
	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L
Main	57.3192	**45.6343**	**55.9098**	**31.9101**	**10.8455**	**21.6838**
C2S	**57.4680**	45.3972	55.8143	31.6213	10.6593	21.5918
Q2S	57.0414	45.2951	55.6705	31.9448	10.5836	21.6477

Table 5 Rouges scores for Debatepedia and QMSum using different way to generate NLI scores

6 Analysis

Table 4 is the average nli score in test data of Debatepedia. We compare the averaged probability distribution of baseline's outputs and our method's outputs with the ground-truth summaries. We can observe that after adding nli loss, the contradiction probability is reduced as we expect.

Average Probability	**entailment**	**neutral**	**contradiction**
Groundtruth	0.4835	0.4003	**0.1383**
Baseline	0.4895	0.3660	0.1644
NLI loss	0.4963	0.3458	0.1579

Table 4 Distribution of entailment, neutral and contradiction samples, in Debatepedia dataset

We also try to use different combinations to generate the NLI score, for example context -> summary (denoted as c2s) and query -> summary (denoted as q2c). However, the experiment's results show that both these two variants will hurt the performance, which means that the information from both query and context are necessary. Results are presented in Table 5

We also tried to fine-tune the NLI model we used to generate a more robust score, but we find it difficult to create the training data. In our design, we used the (context, query, summary) tuple in training data as positive samples and randomly selected tuple as negative samples. However, the negative samples are too easy for the models and mislead the models. Adding too many random negative samples will also make the model hard to converge. So we need to design more difficult negative samples.

7 Conclusion

We propose incorporating signals derived from Natural Language Inference tasks to train the QFS model after the success of using NLI to improve the QA system. To strengthen signals, we create a new loss using the NLI loss. The loss can effectively measure the {context, query, summary} tuple's relationship. These scores join in the gradient computation and back-propagation, rather than acting as additional inputs. Our experiment shows that by using NLI, we were able to achieve better performance than the state-of-the-art model, albeit a minor improvement, which is due to the fact that NLI's loss signal is not strong enough to have a significant impact. Based on this work, we have more faith in NLI's ability to verify and improve the QFS task. We plan to continue to work this summer and the future work will primarily focus on two areas: data augmentation and reinforcement learning attempts.

8 Future Work

As we mentioned in the analysis section, we believe that more high-quality negative samples would improve the performance of our model. As a result, one of the next steps will be to expand the dataset with more high-quality negative samples. One option is to generate on our own, which we haven't yet figured out how to do. Another option, which appears to be simpler, is to select negative samples from other datasets. Additionally, data augmentation will aid in the testing of our model's performance across a variety of datasets.

Another future direction is to use reinforcement learning to help tune the QFS model's parameters. One option is to use the NLI scores as a reward for the policy gradient when tuning the QFS model's parameters. This could make it easier to get outputs that are more closely related to the query and the content.

Work Distribution

Chao Zhai: Data pre-processing and data-to-input pipeline building for Debatepedia and QMSum. Experiment results Analysis.

Dat Nguyen: Data pre-processing and shorter source document extraction for QMSum. Experiment results Analysis.

Hitesh Manivannan: Literature review, implementation of the NLI loss and fine-tuning the NLI model

Lihan Zhu: Literature review, experiment running and hyperparameters tuning.

Zhongheng He: Literature review, implementation of the QFS model and NLI loss, and experiment running.

References

Marina Litvak and Natalia Vanetik. 2017. Query-based summarization using MDL principle. In Proceedings of the MultiLing 2017 Workshop on Summariza- tion and Summary Evaluation Across Source Types and Genres, pages 22–31, Valencia, Spain. Association for Computational Linguistics.

Elozino Egonmwan, Vittorio Castelli, and Md Arafat Sultan. 2019. Cross-task knowledge transfer for query-based text summarization. In Proceedings of the 2nd Workshop on Machine Reading for Question Answering, pages 72–77, Hong Kong, China. Association for Computational Linguistics.

Yumo Xu and Mirella Lapata. 2020. Coarse-to-fine query focused multi-document summarization. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 3632–3645, Online. Association for Computa- tional Linguistics.

Dan Su, Yan Xu, Tiezheng Yu, Farhad Bin Siddique, Elham Barezi, and Pascale Fung. 2020. CAiRE-COVID: A question answering and query-focused multi-document summarization system for COVID-19 scholarly information management. In Proceedings of the 1st Workshop on NLP for COVID-19 (Part 2) at EMNLP 2020, Online. Association for Computational Linguistics.

Dan Su, Tiezheng Yu, and Pascale Fung. 2021. Improve query focused abstractive summarization by incorporating answer relevance. In Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 3124–3131, Online. Association for Computational Linguistics.

Jifan Chen, Eunsol Choi, and Greg Durrett. 2021. Can nli models verify qa systems' predictions? In EMNLP Findings. Jifan Chen, Shih-ting Li

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Vaswani, Ashish, et al. "Attention is all you need." Advances in neural information processing systems 30 (2017).

Lewis, Mike, et al. "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension." arXiv preprint arXiv:1910.13461 (2019).

Dan Su, Yan Xu, Genta Indra Winata, Peng Xu, Hyeondey Kim, Zihan Liu, and Pascale Fung. 2019. Generalizing question answering system with pre-trained language model fine-tuning. In Proceedings of the 2nd Workshop on Machine Reading for Question Answering, pages 203–211.

Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).

Liu, Yinhan, et al. "Roberta: A robustly optimized bert pretraining approach." arXiv preprint arXiv:1907.11692 (2019).

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. In Advances in neural information processing systems, pp. 5754–5764, 2019.

He, Pengcheng, et al. "Deberta: Decoding-enhanced bert with disentangled attention." arXiv preprint arXiv:2006.03654 (2020).

Falke T, Ribeiro L F R, Utama P A, et al. Ranking generated summaries by correctness: An interesting but challenging application for natural language inference[C]//Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. 2019: 2214-2220.

Li H, Zhu J, Zhang J, et al. Ensure the correctness of the summary: Incorporate entailment knowledge into abstractive sentence summarization[C] //Proceedings of the 27th International Conference on Computational Linguistics. 2018: 1430-1441.

Don't Give Me the Details, Just the Summary! Topic-Aware Convolutional Neural Networks for Extreme Summarization Shashi Narayan, Shay B. Cohen, Mirella Lapata. arXiv preprint arXiv:1808.08745

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. arXiv preprint arXiv:1908.08345.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In Text summarization branches out, pages 74–81.

Nema, Preksha, et al. "Diversity driven attention model for query-based abstractive summarization." arXiv preprint arXiv:1704.08300 (2017).

Zhong, Ming, et al. "QMSum: A new benchmark for query-based multi-domain meeting summarization." arXiv preprint arXiv:2104.05938 (2021).