Transforming Wikipedia into Augmented Data for Query-Focused Summarization

Haichao Zhu[†]*, **Li Dong**[‡], **Furu Wei**[‡], **Bing Qin**[†♮], **Ting Liu**[†♮]
[†]Harbin Institute of Technology, Harbin, China
[‡]Microsoft Research, Beijing, China
[♮]Peng Cheng Laboratory, Shenzhen, China
{hczhu,qinb,tliu}@ir.hit.edu.cn
{lidong1,fuwei}@microsoft.com

Abstract

The manual construction of a query-focused summarization corpus is costly and time-consuming. The limited size of existing datasets renders training data-driven summarization models challenging. In this paper, we use Wikipedia to automatically collect a large query-focused summarization dataset (named as WIKIREF) of more than 280,000 examples, which can serve as a means of data augmentation. Moreover, we develop a query-focused summarization model based on BERT to extract summaries from the documents. Experimental results on three DUC benchmarks show that the model pre-trained on WIKIREF has already achieved reasonable performance. After fine-tuning on the specific datasets, the model with data augmentation outperforms the state of the art on the benchmarks.

1 Introduction

Query-focused summarization aims to create a brief, well-organized and informative summary for a document with specifications described in the query. Various unsupervised methods (Carbonell and Goldstein, 1998; Erkan and Radev, 2004; McDonald, 2007; Wan and Xiao, 2009; Feigenblat et al., 2017; Baumel et al., 2018) and supervised methods (Galley, 2006; Ouyang et al., 2011; Li et al., 2013; Cao et al., 2016; Ren et al., 2017) have been proposed for the purpose. The task is first introduced in DUC 2005 (Dang, 2005), with human annotated data released until 2007. The DUC benchmark datasets are of high quality. But the limited size renders training query-focused summarization models challenging, especially for the data-driven methods. Meanwhile, manually constructing a large-scale query-focused summarization dataset is quite costly and time-consuming.

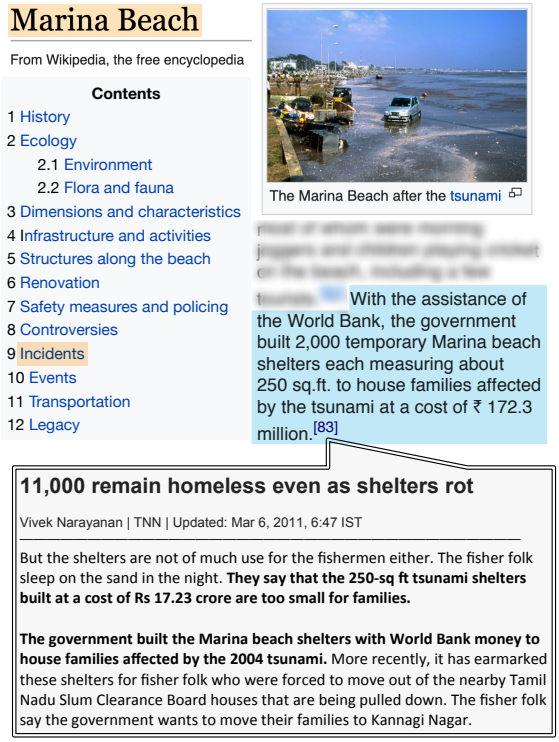

Figure 1: An example of the automatic query-focused summarization example construction. Given a statement in Wikipedia article "*Marina Beach*", we take the body text of citation as the document, use the article title along with section titles to form a query (i.e., "*Marina Beach, Incidents*"), and the statement is the summary.

In order to advance query-focused summarization with limited data, we improve the summarization model with data augmentation. Specifically, we transform Wikipedia into a large-scale query-focused summarization dataset (named as WIKIREF). To automatically construct query-focused summarization examples using Wikipedia, the statements' citations in Wikipedia articles as pivots to align the queries and documents. Figure 1 shows an example that is constructed by the proposed method. We first take the highlighted

* Contribution during internship at Microsoft Research Asia.

statement as the summary. Its supporting citation is expected to provide an adequate context to derive the statement, thus can serve as the source document. On the other hand, the section titles give a hint about which aspect of the document is the summary's focus. Therefore, we use the article title and the section titles of the statement to form the query. Given that Wikipedia is the largest online encyclopedia, we can automatically construct massive query-focused summarization examples.

Most systems on the DUC benchmark are extractive summarization models. These systems are usually decomposed into two subtasks, i.e., sentence scoring and sentence selection. Sentence scoring aims to measure query relevance and sentence salience for each sentence, which mainly adopts feature-based methods (Carbonell and Goldstein, 1998; Ouyang et al., 2011; Wan and Xiao, 2009). Sentence selection is used to generate the final summary with the minimal redundancy by selecting highest ranking sentences one by one.

In this paper, we develop a BERT-based model for query-focused extractive summarization. The model takes the concatenation of the query and the document as input. The query-sentence and sentence-sentence relationships are jointly modeled by the self-attention mechanism (Vaswani et al., 2017). The model is fine-tuned to utilize the general language representations of BERT (Devlin et al., 2019).

Experimental results on three DUC benchmarks show that the model achieves competitive performance by fine-tuning and outperforms previous state-of-the-art summarization models with data augmentation. Meanwhile, the results demonstrate that we can use WIKIREF as a large-scale dataset to advance query-focused summarization research.

2 Related Work

A wide range of unsupervised approaches have been proposed for extractive summarization. Surface features, such as n-gram overlapping, term frequency, document frequency, sentence positions (Ren et al., 2017), sentence length (Cao et al., 2016), and TF-IDF cosine similarity (Wan and Xiao, 2009). Maximum Marginal Relevance (MMR) (Carbonell and Goldstein, 1998) greedily selects sentences and considered the trade-off between saliency and redundancy. McDonald (2007) treat sentence selection as an optimization problem and solve it using Integer Linear Programming (ILP). Lin and Bilmes (2010) propose using submodular functions to maximize an objective function that considers the trade-off between coverage and redundancy terms.

Graph-based models make use of various inter-sentence and query-sentence relationships are also widely applied in the extractive summarization area. LexRank (Erkan and Radev, 2004) scores sentences in a graph of sentence similarities. Wan and Xiao (2009) apply manifold ranking to make use of the sentence-to-sentence and sentence-to-document relationships and the sentence-to-query relationships. We also model the above mentioned relationships, except for the cross-document relationships, like a graph at token level, which are aggregated into distributed representations of sentences.

Supervised methods with machine learning techniques (Galley, 2006; Ouyang et al., 2011; Li et al., 2013) are also used to better estimate sentence importance. In recent years, few deep neural networks based approaches have been used for extractive document summarization. Cao et al. (2016) propose an attention-base model which jointly handles sentence salience ranking and query relevance ranking. It automatically generates distributed representations for sentences as well as the document. To leverage contextual relations for sentence modeling, Ren et al. (2017) propose CRSum that learns sentence representations and context representations jointly with a two-level attention mechanism. The small data size is the main obstacle of developing neural models for query-focused summarization.

3 Problem Formulation

Given a query $\mathcal{Q} = (q_1, q_2, ..., q_m)$ of m token sequences and a document $\mathcal{D} = (s_1, s_2, ..., s_n)$ containing n sentences, extractive query-focused summarization aims to extract a salient subset of $\mathcal{D}$ that is related to the query as the output summary $\hat{\mathcal{S}} = \{\hat{s}_i | \hat{s}_i \in \mathcal{D}\}$. In general, the extrative summarization task can be tackled by assigning each sentence a label to indicate the inclusion in the summary or estimating scores for ranking sentences, namely sentence classification or sentence regression.

In sentence classification, the probability of putting sentence s_i in the output summary is $P(s_i|\mathcal{Q}, \mathcal{D})$. We factorize the probability of predicting $\hat{\mathcal{S}}$ as the output summary $P(\hat{\mathcal{S}}|\mathcal{Q}, \mathcal{D})$ of

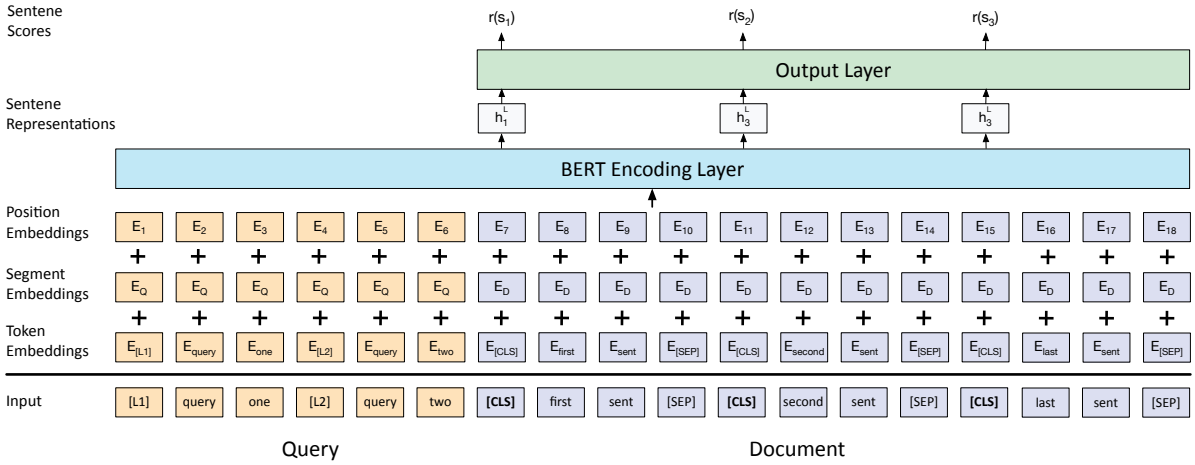

Figure 2: The overview of the proposed BERT-based extractive summarization model. We use special tokens (e.g., [L1], [L2]) to indicate hierarchial structure in queries. We surround each sentence with a [CLS] token before and a [SEP] token after. The input representations of each token are composed of three embeddings. The hidden vectors of [CLS] tokens from the last layer are used to represent and score sentences.

document $\mathcal{D}$ given query $\mathcal{Q}$ as:

$$P(\hat{\mathcal{S}}|\mathcal{Q},\mathcal{D}) = \prod_{\hat{s}_i \in \hat{\mathcal{S}}} P(\hat{s}_i|\mathcal{Q},\mathcal{D})) \quad (1)$$

In sentence regression, extractive summarization is achieved via sentence scoring and sentence selection. The former scores $\mathrm{r}(s_i|\mathcal{Q},\mathcal{D})$ a sentence s_i by considering its relevance to the query $\mathcal{Q}$ and its salience to the document $\mathcal{D}$. The latter generates a summary by ranking sentences under certain constraints, e.g., the number of sentences and the length of the summary.

4 Query-Focused Summarization Model

Figure 2 gives an overview of our BERT-based extractive query-focused summmarization model. For each sentence, we use BERT to encode its query relevance, document context and salient meanings into a vector representation. Then the vector representations are fed into a simple output layer to predict the label or estimate the score of each sentence.

4.1 Input Representation

The query $\mathcal{Q}$ and document $\mathcal{D}$ are flattened and packed as a token sequence as input. Following the standard practice of BERT, the input representation of each token is constructed by summing the corresponding token, segmentation and position embeddings. Token embeddings project the one-hot input tokens into dense vector representations. Two segment embeddings $\mathbf{E}_Q$ and $\mathbf{E}_D$ are used to indicate query and document tokens respectively. Position embeddings indicate the absolute position of each token in the input sequence. To embody the hierarchical structure of the query in a sequence, we insert a [L#] token before the #-th query token sequence. For each sentence, we insert a [CLS] token at the beginning and a [SEP] token at the end to draw a clear sentence boundary.

4.2 BERT Encoding Layer

In this layer, we use BERT (Devlin et al., 2019), a deep Transformer (Vaswani et al., 2017) consisting of stacked self-attention layers, as encoder to aggregate query, intra-sentence and inter-sentence information into sentence representations. Given the packed input embeddings $\mathbf{H}^0 = \left[\mathbf{x}_1, ..., \mathbf{x}_{|x|}\right]$, we apply an L-layer Transformer to encode the input:

$$\mathbf{H}^l = \mathrm{Transformer}_l(\mathbf{H}^{l-1}) \quad (2)$$

where $l \in [1, L]$. At last, we use the hidden vector $\mathbf{h}_i^L$ of the i-th [CLS] token as the contextualized representation of the subsequent sentence.

4.3 Output Layer

The output layer is used to score sentences for extractive query-focused summarization. Given $\mathbf{h}_i^L \in \mathbb{R}^d$ is the vector representation for the i-th sentence. When the extracive summarization is carried out through sentence classification , the output layer is a linear layer followed by a sigmoid

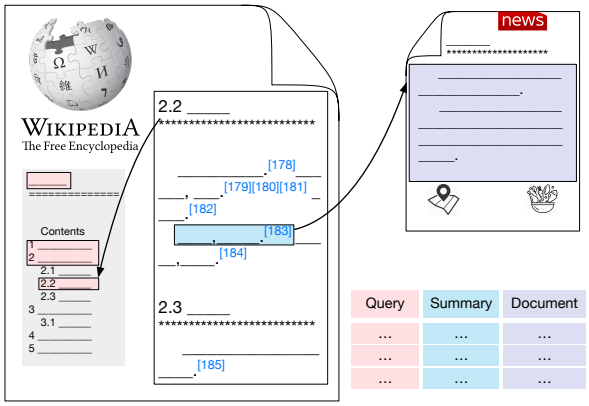

Figure 3: Illustration of WIKIREF examples creation using Wikipedia and reference pages.

function:

$$P(s_i|\mathcal{Q},\mathcal{D}) = \text{sigmoid}(\mathbf{W}_c\mathbf{h}_i^L + \mathbf{b}_c) \quad (3)$$

where $\mathbf{W}_c$ and $\mathbf{b}_c$ are trainable parameters. The output is the probability of including the i-th sentence in the summary.

In the setting of sentence regression, a linear layer without activation function is used to estimate the score of a sentence:

$$\text{r}(s_i|\mathcal{Q},\mathcal{D}) = \mathbf{W}_r\mathbf{h}_i^L + \mathbf{b}_r \quad (4)$$

where $\mathbf{W}_r$ and $\mathbf{b}_r$ are trainable parameters.

4.4 Training Objective

The training objective of sentence classification is to minimize the binary cross-entropy loss:

$$\mathcal{L} = -\sum_i^n y_i \text{log}P(s_i|\mathcal{Q},\mathcal{D})) + (1-y_i)\text{log}(1-P(s_i|\mathcal{Q},\mathcal{D})) \quad (5)$$

where $y_i \in \{0,1\}$ is the oracle label of the i-th sentence.

The training objective of sentence regression is to minimize the mean square error between the estimated score and the oracle score:

$$\mathcal{L} = \frac{1}{n}\sum_i^n(\text{r}(s_i|\mathcal{Q},\mathcal{D}) - \text{f}(s_i|\mathcal{S}^*))^2 \quad (6)$$

where $\mathcal{S}^*$ is the oracle summary and $\text{f}(s_i|\mathcal{S}^*)$ is the oracle score of the i-th sentence.

5 WIKIREF: Transforming Wikipedia into Augmented Data

We automatically construct a query-focused summarization dataset (named as WIKIREF) using Wikipedia and corresponding reference web pages. In the following sections, we will first elaborate the creation process. Then we will analyze the queries, documents and summaries quantitatively and qualitatively.

5.1 Data Creation

We follow two steps to collect and process the data: (1) we crawl English Wikipedia and the references of the Wikipedia articles and parse the HTML sources into plain text; (2) we preprocess the plain text and filter the examples through a set of fine-grained rules.

5.1.1 Raw Data Collection

To maintain the highest standards possible, most statements in Wikipedia are attributed to reliable, published sources that can be accessed through hyperlinks. In the first step, we parse the English Wikipedia database dump into plain text and save statements with citations. If a statement is attributed multiple citations, only the first citation is used. We also limit the sources of the citations to four types, namely web pages, newspaper articles, press and press release. A statement may contain more than one sentence.

The statement can be seen as a summary of the supporting citations from a certain aspect. Therefore, we can take the body of the citation as the document and treat the statement as the summary. Meanwhile, the section titles of a statement could be used as a natural coarse-grained query to specify the focused aspects. Then we can form a complete query-focused summarization example by referring to the statement, attributed citation and section titles along with the article title as summary, document and query respectively. It is worth noticing that the queries in WIKIREF dataset are thus keywords, instead of natural language as in other query-focused summarization datasets.

We show an example in Figure 3 to illustrate the raw data collection process. The associated query, summary and the document are highlighted in colors in the diagram. At last, we have collected more than 2,000,000 English examples in total after the raw data collection step.

5.1.2 Data Curation

To make sure the statement is a plausible summary of the cited document, we process and filter the examples through a set of fine-grained rules. The text is tokenized and lemmatized using Spacy. First,

	Train	Dev	Test
Total Examples	256,724	12,000	12,000
Wiki Articles	160,223	11,457	11,476
Doc. Tokens	397.7	395.4	398.7
Doc. Sents	18.8	18.7	18.8
Summary Tokens	36.1	35.9	36.2
Summary Sents	1.4	1.4	1.4
Query Depth	2.5	2.5	2.5
Query Tokens	6.7	6.8	6.7

Table 1: Statistics of the WIKIREF dataset.

we calculate the unigram recall of the document, where only the non-stop words are considered. We throw out the example whose score is lower than the threshold. Here we set the threshold to 0.5 empirically, which means at least more than half of the summary tokens should be in the document. Next, we filter the examples with multiple length and sentence number constraints. To set reasonable thresholds, we use the statistics of the examples whose documents contain no more than 1,000 tokens. The 5th and the 95th percentiles are used as low and high thresholds of each constraint. Finally, in order to ensure generating the summary with the given document is feasible, we filter the examples through extractive oracle score. The extractive oracle is obtained through a greedy search over document sentence combinations with maximum 5 sentences. Here we adopt ROUGE-2 recall as scoring metric and only the examples with an oracle score higher than 0.2 are kept. After running through the above rules, we have the WIKIREF dataset with 280,724 examples. We randomly split the data into training, development and test sets and ensure no document overlapping across splits.

5.2 Data Statistics

Table 1 show statistics of the WIKIREF dataset. The development set and the test set contains 12,000 examples each. The statistics across splits are evenly distributed and no bias observed. The numerous Wikipedia articles cover a wide range of topics. The average depth of the query is 2.5 with article titles are considered. Since the query are keywords in WIKIREF, it is relatively shorter than the natural language queries with an average length of 6.7 tokens. Most summaries are composed of one or two sentences. And the document contains 18.8 sentences on average.

Oracle Score	Query Relatedness	Doc Salience
[20, 30)	1.87	1.33
[30, 50)	1.80	1.40
[50, 70)	1.87	1.53
[70, 100]	1.93	1.60

Table 2: Quality rating results of human evaluation on the WIKIREF dataset. "Query Relatedness": 2 for summary completely related to the query, 1 for summary partially related to the query, 0 otherwise. "Doc Salience": 2 for summary conveys all salient document content, 1 for summary conveys partial salient document content, 0 otherwise.

5.3 Human Evaluation

We also conduct human evaluation on 60 WIKIREF samples to examine the quality of the automatically constructed data. We partition the examples into four bins according to the oracle score and then sample 15 examples from each bin. Each example is scored in two criteria: (1) "Query Relatedness" examines to what extent the summary is a good response to the query and (2) "Doc Salience" examines to what extent the summary conveys salient document content given the query.

Table 2 shows the evaluation result. We can see that most of the time the summaries are good responses to the queries across bins. Since we take section titles as query and the statement under the section as summary, the high evaluation score can be attributed to Wikipedia pages of high quality. When the oracle scores are getting higher, the summaries continue to better convey the salient document content specified by the query. On the other hand, we notice that sometimes the summaries only contain a proportion of salient document content. It is reasonable since reference articles may present several aspects related to topic. But we can see that it is mitigated when the oracle scores are high on the WIKIREF dataset.

6 Experiments

In this section, we present experimental results of the proposed model on the DUC 2005, 2006, 2007 datasets with and without data augmentation. We also carry out benchmark tests on WIKIREF as a standard query-focused summarization dataset.

6.1 Implementation Details

We use the uncased version of BERT-base for fine-tuning. The max sequence length is set to 512.

	Dev			Test		
	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-1	ROUGE-2	ROUGE-L
ALL	14.05	7.84	12.97	14.09	7.88	13.01
LEAD	26.55	10.66	21.99	26.32	10.48	21.81
TRANSFORMER	28.18	12.92	23.92	28.07	12.80	23.79
BERT (Reg)	34.72	18.06	29.37	34.62	17.99	29.29
BERT (Class)	**36.30**	**19.15**	**31.04**	**35.98**	**18.81**	**30.73**
- Query	33.81	16.75	28.71	33.42	16.32	28.38
Oracle (Reg)	51.34	35.80	45.62	51.41	35.89	45.68
Oracle (Class)	54.34	37.39	48.34	54.46	37.52	48.51

Table 3: ROUGE scores of baselines and the proposed model on WIKIREF dataset. "Class" and "Reg" represent classification and regression, which indicate the supervision type used for training. "- Query" indicates removing queries from the input.

We use Adam optimizer (Kingma and Ba, 2015) with learning rate of 3e-5, β_1 = 0.9, β_2 = 0.999, L2 weight decay of 0.01, and linear decay of the learning rate. We split long documents into multiple windows with a stride of 100. Therefore, a sentence can appear in more than one windows. To avoid making predictions on an incomplete sentence or with suboptimal context, we score a sentence only when it is completely included and its context is maximally covered. The training epoch and batch size are selected from {3, 4}, and {24, 32}, respectively.

6.2 Evaluation Metrics

For summary evaluation, we use ROUGE (Lin, 2004) as our automatic evaluation metric. ROUGE[1] is the official metrics of the DUC benchmarks and widely used for summarization evaluation. ROUGE-N measures the summary quality by counting overlapping N-grams with respect to the reference summary. Whereas ROUGE-L measures the longest common subsequence. To compare with previous work on DUC datasets, we report the ROUGE-1 and ROUGE-2 recall computed with official parameters [2] that limits the length to 250 words. On the WIKIREF dataset, we report ROUGE-1, ROUGE-2 and ROUGE-L scores[3].

6.3 Experiments on WIKIREF

6.3.1 Settings

We first train our extractive summarization model on the WIKIREF dataset through sentence classification. And we need the ground-truth binary labels of sentences to be extracted. However, we can not find the sentences that exactly match the reference summary for most examples. To solve this problem, we use a greedy algorithm similar to Zhou et al. (2018) to find an oracle summary with document sentences that maximizes the ROUGE-2 F1 score with respect to the reference summary. Given a document of n sentences, we greedily enumerate the combination of sentences. For documents that contain numerous sentences, searching for an global optimal combination of sentences is computationally expensive. Meanwhile it is unnecessary since the reference summaries contain no more than four sentences. So we stop searching when no combination with i sentences scores higher than the best the combination with i-1 sentences.

We also train an extractive summarization model through sentence regression. For each sentence, the oracle score for training is the ROUGE-2 F1 score.

During inference, we rank sentences according to their predicted scores. Then we append the sentence one by one to form the summary if it is not redundant and scores higher than a threshold. We skip the redundant sentences that contain overlapping trigrams with respect to the current output summary as in Liu (2019). The threshold is searched on the development set to obtain the highest ROUGE-2 F1 score.

6.3.2 Baselines

We apply the proposed model and the following baselines:

[1] ROUGE-1.5.5

[2] -n 2 -x -m -2 4 -u -c 95 -r 1000 -f A -p 0.5 -t 0 -l 250

[3] -n 2 -m -c 95 -r 1000

	DUC 2005		DUC 2006		DUC 2007	
	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2	ROUGE-1	ROUGE-2
LEAD*	29.71	4.69	32.61	5.71	36.14	8.12
QUERY-SIM*	32.95	5.91	35.52	7.10	36.32	7.94
SVR*	36.91	7.04	39.24	8.87	43.42	11.10
CRSUM*	36.96	7.01	39.51	9.19	41.20	11.17
ATTSUM†	37.01	6.99	40.90	9.40	43.92	11.55
DA Pre-trained (Class)	36.19	7.00	38.67	7.88	40.08	9.19
DA Pre-trained (Reg)	36.52	7.02	38.81	8.37	41.09	10.29
BERT	38.57	7.98	41.35	9.60	43.55	11.39
BERT + DA (Class)	**38.77**	8.31	41.65	**10.24**	44.31	11.85
BERT + DA (Reg)	38.60	**8.43**	**41.67**	10.04	**44.54**	**12.01**
Oracle	43.71	13.77	48.02	17.22	49.80	19.19

Table 4: ROUGE scores on the DUC 2005, 2006 and 2007 datasets. "*" indicates results taken from Ren et al. (2017). "†" indicates results taken from Cao et al. (2016). "DA" is short for data augmentation using WIKIREF dataset. "DA Pre-trained" denotes applying the model pre-trained on augmentation data to directly extract summaries for DUC datasets. "Class" and "Reg" represent classification super and regression, which indicate the supervision type used on augmentation data.

	2005	2006	2007
Clusters	50	50	45
Documents	1,593	1,250	1,125
Sentences	46,033	34,585	24,176

Table 5: Statistics of DUC datasets.

ALL outputs all sentences of the document as summary.

LEAD is a straightforward summarization baseline that selects the leading sentences. We take the first two sentences for that the groundtruth summary contains 1.4 sentences on average.

TRANSFORMER uses the same structure as the BERT with randomly initialized parameters.

6.3.3 Results

The results are shown in Table 3. Our proposed model with classification output layer achieves 18.81 ROUGE-2 score on the WIKIREF test set. On average, the output summary consists of 1.8 sentences. LEAD is a strong unsupervised baseline that achieves comparable results with the supervised neural baseline Transformer. Even though WIKIREF is a large-scale dataset, training models with parameters initialized from BERT still significantly outperforms Transformer. The model trained using sentence regression performs worse than the one supervised by sentence classification. It is in accordance with oracle labels and scores. We observe a performance drop when generating summaries without queries (see "-Query"). It proves that the summaries in WIKIREF are indeed query-focused.

6.4 Experiments on DUC Datasets

DUC 2005-2007 are query-focused multi-document summarization benchmarks. The documents are from the news domain and grouped into clusters according to their topics. And the summary is required to be no longer than 250 tokens. Table 5 shows statistics of the DUC datasets. Each document cluster has several reference summaries generated by humans and a query that specifies the focused aspects and desired information. We show an example query from the DUC 2006 dataset below:

> EgyptAir Flight 990?
> What caused the crash of EgyptAir Flight 990?
> Include evidence, theories and speculation.

The first narrative is usually a title and followed by several natural language questions or narratives.

6.4.1 Settings

We follow standard practice to alternately train our model on two years of data and test on the third. The oracle scores used in model training are ROUGE-2 recall of sentences. In this paper, we score a sentence by only considering the query and the its document. Then we rank sentences according to the estimated scores across documents within

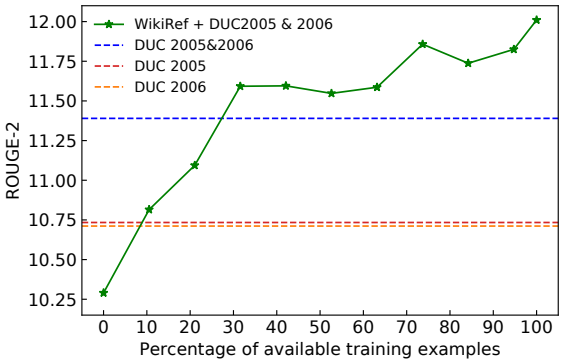

Figure 4: ROUGE-2 score on the DUC 2007 evaluation set with various training data. The horizontal lines indicate training on DUC 2005, on DUC 2006, and on both. The green line indicates training on only a proportion of DUC 2005 and DUC 2006 examples with the WIKIREF data augmentation. The x-axis indicates the number of used training examples, along with data augmentation.

a cluster. For each cluster, we fetch the top-ranked sentences iteratively into the output summary with redundancy constraint met. A sentence is redundant if more than half of its bigrams appear in the current output summary.

The WIKIREF dataset is used as augmentation data for DUC datasets in two steps. We first fine-tune BERT on the WIKIREF dataset. Subsequently, we use the DUC datasets to further fine-tune parameters of the best pre-trained model.

6.4.2 Baselines

We compare our method with several previous query-focused summarization models, of which the ATTSUM is the state-of-the-art model:

LEAD is a simple baseline that selects leading sentences to form a summary.

QUERY-SIM is an unsupervised method that ranks sentences according to its TF-IDF cosine similarity to the query.

SVR (Ouyang et al., 2011) is a supervised baseline that extracts both query-dependent and query-independent features and then using Support Vector Regression to learn the weights of features.

ATTSUM (Cao et al., 2016) is a neural attention summarization system that tackles query relevance ranking and sentence salience ranking jointly.

CRSUM (Ren et al., 2017) is the contextual relation-based neural summarization system that improves sentence scoring by utilizing contextual relations among sentences.

6.4.3 Results

Table 4 shows the ROUGE scores of comparison methods and our proposed method. Fine-tuning BERT on DUC datasets alone outperforms previous best performing summarization systems on DUC 2005 and 2006 and obtains comparable results on DUC 2007. Our data augmentation method further advances the model to a new state of the art on all DUC benchmarks. We also notice that models pre-trained on the augmentation data achieve reasonable performance without further fine-tuning model parameters. It implies the WIKIREF dataset reveals useful knowledge shared by the DUC datatset. We pre-train models on augmentation data under both sentence classification and sentence regression supervision. The experimental results show that both supervision types yield similar performance.

6.4.4 Human Evaluation

To better understand the improvement brought by augmentation data, we conduct a human evaluation of the output summaries before and after data augmentation. We sample 30 output summaries of the DUC 2006 dataset for analysis. And we find that the model augmented by the WIKIREF dataset produces more query-related summaries on 23 examples. Meanwhile,the extracted sentences are usually less redundant. We attribute these benefits to the improved coverage and query-focused extraction brought by the large-scale augmentation data.

6.4.5 Ablation Study

To further verify the effectiveness of our data augmentation method, we first pre-train models on the WIKIREF dataset and then we vary the number of golden examples for fine-tuning. Here we take the DUC 2007 dataset as test set and use DUC 2005 and 2006 as training set. In Figure 4, we present ROUGE-2 scores of fine-tuning BERT on DUC datasets for comparison. Either using DUC 2005 alone or DUC 2006 alone yields inferior performance than using both. Our proposed data augmentation method can obtain competitive results using only no more than 30 golden examples and outperform BERT fine-tuning thereafter.

6.5 Discussion

The improvement introduced by using the WIKIREF dataset as augmentation data is trace-

able. At first, the document in the DUC datasets are news articles and we crawl newspaper webpages as one source of the WIKIREF documents. Secondly, queries in the WIKIREF dataset are hierarchical that specify the aspects it focuses on gradually. This is similar to the DUC datasets that queries are composed of several narratives to specify the desired information. The key difference is that queries in the WIKIREF dataset are composed of key words, while the ones in the DUC datasets are mostly natural language. At last, we construct the WIKIREF dataset to be a large-scale query-focused summarization dataset that contains more than 280,000 examples. In comparison, the DUC datasets contain only 145 clusters with around 10,000 documents. Therefore, query relevance and sentence context can be better modeled using data-driven neural methods with WIKIREF. And it provides a better starting point for fine-tuning on the DUC datasets.

7 Conclusions

In this paper, we propose to automatically construct a large-scale query-focused summarization dataset WIKIREF using Wikipedia articles and the corresponding references. The statements, supporting citations and article title along with section titles of the statements are used as summaries, documents and queries respectively. The WIKIREF dataset serves as a means of data augmentation on DUC benchmarks. It also is shown to be a eligible query-focused summarization benchmark. Moreover, we develop a BERT-based extractive query-focused summarization model to extract summaries from the documents. The model makes use of the query-sentence relationships and sentence-sentence relationships jointly to score sentences. The results on DUC benchmarks show that our model with data augmentation outperforms the state-of-the-art. As for future work, we would like to model relationships among documents for multi-document summarization.

References

Tal Baumel, Matan Eyal, and Michael Elhadad. 2018. Query focused abstractive summarization: Incorporating query relevance, multi-document coverage, and summary length constraints into seq2seq models. *Computing Research Repository*, arXiv:1801.07704.

Ziqiang Cao, Wenjie Li, Sujian Li, Furu Wei, and Yanran Li. 2016. AttSum: Joint learning of focusing and summarization with neural attention. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 547–556, Osaka, Japan. The COLING 2016 Organizing Committee.

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '98, pages 335–336, New York, NY, USA. ACM.

Hoa Trang Dang. 2005. Overview of duc 2005. In *Proceedings of the document understanding conference*, volume 2005, pages 1–12.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Human Language Technologies: The 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics*.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479.

Guy Feigenblat, Haggai Roitman, Odellia Boni, and David Konopnicki. 2017. Unsupervised query-focused multi-document summarization using the cross entropy method. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, pages 961–964, New York, NY, USA. ACM.

Michel Galley. 2006. A skip-chain conditional random field for ranking meeting utterances by importance. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, pages 364–372, Sydney, Australia. Association for Computational Linguistics.

Diederick P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

Chen Li, Xian Qian, and Yang Liu. 2013. Using supervised bigram-based ilp for extractive summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1004–1013.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Hui Lin and Jeff Bilmes. 2010. Multi-document summarization via budgeted maximization of submodular functions. In *Human Language Technologies: The 2010 Annual Conference of the North American*

Chapter of the Association for Computational Linguistics, pages 912–920.

Yang Liu. 2019. Fine-tune bert for extractive summarization. *Computing Research Repository*, arXiv:1903.10318. Version 1.

Ryan McDonald. 2007. A study of global inference algorithms in multi-document summarization. In *Advances in Information Retrieval*, pages 557–564, Berlin, Heidelberg. Springer Berlin Heidelberg.

You Ouyang, Wenjie Li, Sujian Li, and Qin Lu. 2011. Applying regression models to query-focused multi-document summarization. *Information Processing & Management*, 47(2):227 – 237.

Pengjie Ren, Zhumin Chen, Zhaochun Ren, Furu Wei, Jun Ma, and Maarten de Rijke. 2017. Leveraging contextual sentence relations for extractive summarization using a neural attention model. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '17, pages 95–104, New York, NY, USA. ACM.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz 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.

Xiaojun Wan and Jianguo Xiao. 2009. Graph-based multi-modality learning for topic-focused multi-document summarization. In *International Joint Conference on Artificial Intelligence*.

Qingyu Zhou, Nan Yang, Furu Wei, Shaohan Huang, Ming Zhou, and Tiejun Zhao. 2018. Neural document summarization by jointly learning to score and select sentences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 654–663, Melbourne, Australia. Association for Computational Linguistics.