Generating Query Focused Summaries from Query-Free Resources

Yumo Xu and **Mirella Lapata**
Institute for Language, Cognition and Computation
School of Informatics, University of Edinburgh
10 Crichton Street, Edinburgh EH8 9AB
yumo.xu@ed.ac.uk   mlap@inf.ed.ac.uk

Abstract

The availability of large-scale datasets has driven the development of neural models that create *generic* summaries from single or multiple documents. In this work we consider *query focused* summarization (QFS), a task for which training data in the form of queries, documents, and summaries is not readily available. We propose to decompose QFS into (1) query modeling (i.e., finding supportive evidence within a set of documents for a query) and (2) conditional language modeling (i.e., summary generation). We introduce MARGE, a **Ma**sked **ROUGE** Regression framework for evidence estimation and ranking which relies on a unified representation for summaries and queries, so that summaries in generic data can be converted into *proxy queries* for learning a query model. Experiments across QFS benchmarks and query types show that our model achieves state-of-the-art performance despite learning from weak supervision.[1]

1 Introduction

The neural encoder-decoder framework has become increasingly popular in *generic* summarization (See et al. 2017; Gehrmann et al. 2018; Liu and Lapata 2019a; Fabbri et al. 2019, *inter alia*) thanks to the availability of large-scale datasets containing hundreds of thousands of document-summary pairs. Training data of this magnitude is not readily available for *query focused* summarization (QFS; Dang 2005) which aims to create a short summary from a set of documents that answers a specific query. Existing corpora (Nema et al., 2017; Dang, 2005; Hoa, 2006; Baumel et al., 2016) are relatively small for modern data-hungry neural architectures and have been mostly used for evaluation purposes.

A major bottleneck in leveraging generic summarization data for QFS is the absence of queries (Nema et al., 2017); the majority of existing datasets consist of document-summary pairs, while QFS summaries are expected to answer specific queries. Recent work (Xu and Lapata, 2020; Su et al., 2020; Laskar et al., 2020) sidesteps this problem by resorting to distant supervision from query-relevant NLP resources including question answering (Rajpurkar et al., 2016; Chakraborty et al., 2020) and paraphrase identification (Dolan and Brockett, 2005). Such approaches incorporate query modeling in the summarization process but are even more data hungry compared to generic summarization ones, since they additionally require access to QA datasets which can be extremely costly to create (Bajaj et al., 2016; Kwiatkowski et al., 2019). Moreover, there is often a mismatch between queries in QA datasets and those in QFS scenarios (Xu and Lapata, 2020); the two types of queries are not identically distributed and it is practically infeasible to find appropriate query-related resources for all domains and topics.

In this work we do not assume access to any resources other than those available for generic summarization. We further decompose abstractive QFS into two subtasks: (1) *query modeling* (i.e., finding supportive evidence within a set of documents for a query) and (2) *conditional language modeling* (i.e., generating an abstractive summary based on found evidence). Under this formulation, we use generic summarization data not only for conditional language modeling, but also for learning an evidence ranking model. Inspired by the Cloze task and its applications in NLP (Taylor, 1953; Lewis et al., 2019; Lee et al., 2019), we propose MARGE, a **Ma**sked **ROUGE** regression framework for evidence estimation and ranking. MARGE intro-

---

[1] Our code and data is available at https://github.com/yumoxu/marge.

(a) Unified Masked Representaion

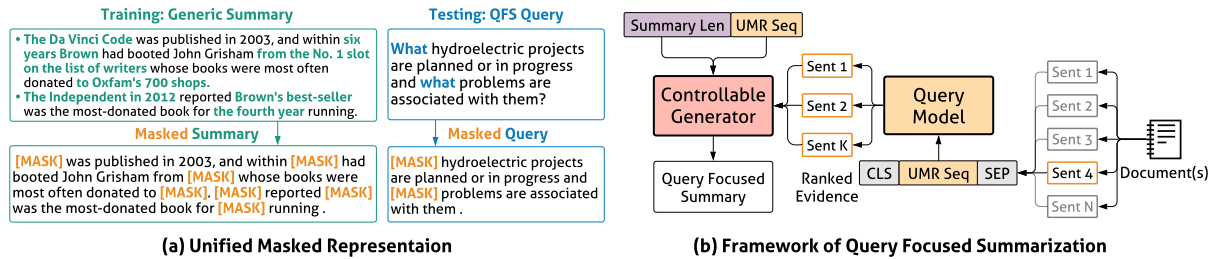**(b) Framework of Query Focused Summarization**

Figure 1: Overview of our abstractive QFS approach. Summaries and queries are rendered with Unified Masked Representation (UMR) for training and testing, respectively. The summarization framework consists of a query model and a controllable generator. The query model ranks sentences in the input document(s) which provide evidence to answer the query; the generator operates over evidence bearing sentences to generate the final summary.

duces a unified representation for *summaries* and *queries*, so that summaries in generic data can be converted into *proxy queries* for learning a query model. Based on the evidence selected by MARGE, we generate abstractive summaries whilst controlling their length and the extent to which the query influences their content.

Our contributions in this work are threefold: we propose a weakly supervised system for abstractive QFS where no query-related resources are required; we discover a new type of connection between generic summaries and QFS queries, and provide a universal representation for them which allows generic summarization data to be exploited for QFS; we provide experimental results on QFS benchmarks, and show that across query types and domains our system achieves state-of-the-art results on both evidence ranking and abstractive QFS.

2   Related Work

The majority of previous QFS approaches have been extractive, operating over queries and document clusters from which they select query-relevant sentences to compose a summary. They mostly differ in the way centrality and relevance are estimated and incorporated, e.g., via manifold ranking (Wan et al., 2007), using a look-ahead strategy (Badrinath et al., 2011), uncertainty prediction (Wan and Zhang, 2014), or attention mechanisms (Li et al., 2017a,b). More recently Xu and Lapata (2020) propose a coarse-to-fine framework that leverages distant supervision from question answering to extract summary-worthy content.

Abstractive QFS has received significantly less attention. This is due to generation models being particularly data-hungry (Lebanoff et al., 2018; Liu and Lapata, 2019a) and the scarcity of QFS training data. The increasing availability of pre-trained models has prompted the development of pipeline-style frameworks for QFS which use resources from a wider range of NLP tasks. For example, Su et al. (2020) fine-tune BART (Lewis et al., 2020) on CNN/DailyMail (Hermann et al., 2015), a single-document summarization dataset, and generate abstracts for QFS by iteratively summarizing paragraphs to a budget. They learn a query model for paragraph selection based on a plethora of QA and machine reading datasets (Su et al., 2019; Rajpurkar et al., 2016). Similarly, Laskar et al. (2020) fine-tune BERTSUM on CNN/DailyMail, and propose a three-stage system which uses supervision from QFS data (typically reserved for evaluation) and related QA and paraphrase identification tasks.

We also focus on abstractive QFS, however, we do not assume access to any additional training resources over and above generic summarization datasets, even for query modeling. Moreover, our system is able to generate long QFS abstracts *all at once*, instead of *iteratively* creating bullet-style summaries which often lack coherence.

3   Problem Formulation

Let $\{(S, \mathcal{D})\}$ denote a generic summarization dataset where $\mathcal{D} = \{d_1, d_2, \ldots, d_M\}$ is a collection of documents with corresponding summaries S. $|\mathcal{D}| = 1$ for single-document summarization (SDS) and $|\mathcal{D}| > 1$ for multi-document summarization (MDS). In QFS, a query Q additionally specifies an information request, $\{(S, \mathcal{D}, Q)\}$. It is often assumed (e.g., in DUC benchmarks) that Q consists of a short title (e.g., *Amnesty International*), and a query narrative which is longer and more detailed (e.g., *What is the scope of operations of Amnesty International and what are the international reactions to its activities?*).

In this work, we propose to decompose QFS

6097

into two sub-tasks, namely *query modeling* and *conditional language modeling*. The query model $q_\theta(D|Q; \theta)$ estimates whether textual units (e.g., sentences) within document cluster D are relevant to query Q, while $p_\phi(S|D, Q; \phi)$ generates summary S conditioned on evidence provided by the query model and (optionally) the query itself (see Figure 1(b) for an illustration). When $S \perp\!\!\!\perp Q$, we have a *query-agnostic* conditional language model $p_\phi(S|D; \phi)$. Otherwise, the conditional language model is *query-guided*. Our query model is trained with distant supervision derived from *generic* summarization data which is easier to obtain (e.g., from online sources) compared to QA datasets which must be annotated from scratch (e.g., for different types of questions and domains). Although queries are not verbalized in generic summarization, we hypothesize that the summaries themselves constitute a response to *latent* queries.

So, how can we reverse-engineer the queries from the summaries? Inspired by the standard Cloze task (Taylor, 1953) and its recent variants (Lewis et al., 2019; Lee et al., 2019), we render queries and summaries in a *Unified Masked Representation* (UMR) which enables summaries to serve as *proxy* queries for model training, as shown in Figure 1(a). We further assume that the answer to these queries can be found in sentences which form part of the document collection $\mathcal{D}$. Although we do not know for certain what these sentences are we can assume that if they have a high ROUGE score against the reference summary they are likely to contain an answer. We therefore use ROUGE as a distant supervision signal, and train a model that takes a query and document sentence as input and estimates their relevance. At inference time, we also render actual queries in UMR and rank all sentences in the document collection with our trained model. The most relevant sentences serve as input to a conditional language model to generate query focused abstractive summaries.

4 Query Modeling

As explained earlier, we train a query model $q_\theta(D|Q; \theta)$ on summary-sentence pairs via distant supervision. We use a summary-based proxy query UMR_S during training and an actual query UMR_Q during testing. In the following, we first describe how UMRs are obtained and then discuss how the query model is trained.

Algorithm 1 Generate Masked Summary

1: **function** MASKSUMMARY($\mathcal{S}, \gamma$) ▷ *Summary sentences and mask ratio*
2:      Parse each $s \in S$ with OpenIE to extract information slots $\mathcal{I}$
3:      Reveal budget $B = |\mathcal{I}| * \gamma$ ▷ *Reveal information partially*
4:      Initialize revealed word number $b = 0$
5:      Initialize masked summary $\mathcal{M}$ to S and fill with [MASK]
6:      Initialize EOM = false   ▷ *End of Masking*
7:      **while** true **do**
8:          $\mathcal{S}_a =$ GETAVAIABLE($\mathcal{S}$)   ▷ *Sentences with masked slots*
9:          **for** $s \leftarrow \mathcal{S}_a$ **do**
10:              $b = b+$ REVEAL(s)   ▷ *Reveal a randomly sampled slot and record its length, i.e., #tokens*
11:              **if** $b \geq B$ **then** EOM = true
12:          **if** EOM **then**   ▷ *Start post-process*
13:              **for** $m \leftarrow \mathcal{M}$ **do**
14:                  MERGE(m)   ▷ *Merge adjacent* [MASK]
15:              **return** $\mathcal{M}$
16: **end function**

Unified Masked Representation The intuition behind UMR is that a summary will encapsulate most salient information a user needs, while a query typically covers only a small fraction. We thus add one or more "placeholders" to the query to represent missing information the user actually seeks. We also identify such information in generic summaries for selective masking, to reduce the distributional shift during training.

The UMR for a summary is the concatenation of its sentential UMRs. To convert a sentence from natural language to UMR, we parse it with Open Information Extraction (Open IE; Stanovsky et al. 2018) to a set of propositions consisting of verbs and their arguments. The latter are considered candidate *information slots* $\mathcal{I}$. We initialize Algorithm 1, by replacing all such slots with a [MASK] token. We subsequently sample and reveal a set of slots subject to a budget constraint. We define the budget as $B = \gamma * |\mathcal{I}|$ where $\gamma \in [0, 1]$ modulates the proportion of tokens to be revealed within $\mathcal{I}$ slots (and is optimized on the development set). Finally, in order to keep the representation of UMR_S and UMR_Q consistent (see next paragraph), we merge adjacent [MASK] tokens to one [MASK] resulting in a partially masked summary.

We mask QFS queries by considering their structure and lexical makeup. Queries in DUC benchmarks often contain *interrogative* words (e.g., *how is A* and *what is B*) and *request* words (e.g., *describe A* and *tell me B*). Following this observation, we manually collect a small set of such query words and replace them with [MASK]. For queries with a title and a narrative, we first mask the narrative and then prepend "[MASK] $\mathcal{T}$.", where $\mathcal{T}$ is a sequence of title tokens. Figure 1(a) shows examples of a masked query and summary.

Evidence Ranking We represent sentences in a document collection and UMR queries with a pretrained BERT model (Devlin et al., 2019). Specifically, we concatenate a UMR query and a candidate sentence to sequence "`[CLS]` $\mathcal{U}$ `[SEP]` $\mathcal{C}$ `[SEP]`" where $\mathcal{U}$ is a sequence of tokens within a UMR query and $\mathcal{C}$ a sequence of tokens in a document sentence (we pad each sequence in a minibatch of L tokens). The `[CLS]` vector serves as input to a single layer neural network which estimates whether the sentence contains sufficient evidence to answer the query (see Figure 1(b) right). We use the mean-square error to compute the loss and update the encoding parameters in BERT via standard backpropagation:

$$\mathcal{L}(\theta) = \frac{1}{|\mathcal{D}|} \sum_{(S,C) \sim \mathcal{D}} \left[(y - \hat{y}(S, C; \theta))^2 \right]. \quad (1)$$

where S, C is a summary-sentence pair sampled from collection D and y the training signal. Recall the summary is rendered as UMR_S.

Previous work (Liu and Lapata, 2019a) has used ROUGE-2 as training signal for paragraph ranking. However, sentences are significantly shorter than paragraphs, and we observe a number of instances with a ROUGE-2 score of 0. We therefore perform label smoothing and define y as the F1 interpolation of ROUGE-2 and ROUGE-1: $y = \text{R}_2(S, C) + \lambda * \text{R}_1(S, C)$ where λ is optimized on the development set. At inference time, we use the trained model to compute the affinity score between UMR_Q and all candidate sentences in $\mathcal{D}$ and rank them accordingly. The highest ranked sentences are deemed query-relevant and passed on to our summary generation model.[2]

Query Narrative Expansion In some cases queries may be relatively short and narratives absent. This can be problematic for our setup since query proxies (in the form of summaries) are typically long and detailed. For datasets with short queries we automatically create query narratives in an unsupervised fashion. We employ LexRank (Erkan and Radev, 2004) to select a subset of representative sentences under a word budget and concatenate them to form narratives (which we append to the original queries).

---

[2]The Cloze task has been also employed in recent work in generic summarization (Huang et al., 2020). In comparison, we address a different research question (i.e., query modeling vs. summary evaluation) based on a different formulation (masked ROUGE regression vs. multiple-choice QA).

5 Query Focused Generation

We also leverage generic summarization datasets to fine-tune a pretrained language model for abstractive QFS. In experiments we employ the publicly released UNILMV2 (Bao et al., 2020) to instantiate the controllable generator shown in Figure 1(b), however any other language model could have been used instead.

With Transformer (Vaswani et al., 2017) as the backbone network, UNILMV2 is jointly pretrained for natural language understanding and generation. Specifically, a bidirectional model is employs an autoencoding objective (AE; identical to Devlin et al. 2019), while a partially autoregressive (PAR) sequence-to-sequence model decomposes the probability of masked tokens in input sequence x as:

$$p(x_M \mid x_{\setminus M}) = \prod_{i=1}^{|M|} \prod_{m \in M_i} p(x_m \mid x_{\setminus M_{\geq i}}) \quad (2)$$

where M is the uniformly-produced factorization order. The masked position set M_i at the ith factorization step can be either a token or a n-gram block. x_M is a set of x_{M_i}, and similarly, $x_{\setminus M}$ is a set of $x_{\setminus M_i}$. The pretraining loss is computed as $\mathcal{L}_{\text{AE}} + \mathcal{L}_{\text{PAR}}$.

At inference, UNILMV2 operates over sentences deemed relevant by the query model and decodes summaries autoregressively (see Figure 1(b) left).

Synthetic MDS Data The pre-trained language model can be fine-tuned on MDS datasets (e.g., Multi-News; Fabbri et al. 2019) which are perhaps better aligned with the QFS task since both MDS and QFS operate over document clusters. We additionally propose a way to create synthetic MDS datasets based on SDS data. This is advantageous for two reasons. Firstly, MDS resources are fairly limited compared to SDS data (Zhang et al., 2018; Lebanoff et al., 2018). And secondly, by construction, we can ensure various data characteristics which might be desirable (e.g., the number of topics represented in the document collection).

A challenge with leveraging SDS for QFS is the summary length (Lebanoff et al., 2018). Summaries in SDS datasets such as CNN/DailyMail (Hermann et al., 2015), are on average 30 tokens long. In contrast, query focused summaries can be as long as 250 tokens. We sidestep this problem by adopting a *retrieval*-based solution. Specifically, we first build a database with all summaries in the

original dataset. For each sample (d_i, s_i), we query the database with summary s_i. We retrieve $N_i - 1$ other summaries $\mathcal{S}_i$ with the bigram hashing and TF-IDF matching method described in Chen et al. (2017). Then, we fetch their corresponding articles $\mathcal{D}_i$, and form the ith cluster as:

$$\mathcal{D}_i^* = \{d_i\} \bigcup \mathcal{D}_i \quad (3)$$

$$\hat{s}_i^* = \text{concat}(s_i, , s_{i,1}, \ldots, s_{i,N_i}), s_{i,n} \in \mathcal{S}_i \quad (4)$$

where $\mathcal{D}_i^*$ are the source documents, and $\hat{s}_i^*$ is a potentially redundant summary of them. We set N_i to minimize the length difference between $\hat{s}_i^*$ and our summary length requirement (e.g., 250 tokens). To obtain the final summary s_i^*, we eliminate redundancy by selecting sentences from the start of $\hat{s}_i^*$, skipping sentences that have high cosine similarity with those which have already been selected.

Summarization Input In generic MDS, the input to the summarization model is a long sequence, i.e., documents within a cluster are concatenated together and sentences in each document follow their original order (Fabbri et al., 2019). In QFS, information about absolute (document) position is lost after evidence ranking. As a result, there is a discrepancy between training and testing for our generation model. To mitigate this, we collect all sentences across documents for each training sample and rank them in descending order according to their ROUGE-2 score against the reference summary. The pretrained language model is fine-tuned against this evidence-ranked list of sentences. During inference, when *actual* queries are available, we instead use the top sentences ranked by our query model as input to summary generation.

Query Guidance Given that summarization input essentially consists of sentences that are highly relevant to the query, an obvious question concerns the usefulness of explicitly modeling the query during generation. We thus instantiate two conditional language models. For a *query-guided* summarizer $p_\phi(S|D, Q; \phi)$, we prepend UMRS_S to the selected evidence during training and UMR_Q at inference. While for a *query-agnostic* summarizer $p_\phi(S|D; \phi)$, we only consider the selected evidence as input to our summarizer and this setting is identical to generic MDS.

Length Control QFS tasks usually require summaries of a fixed length budget (e.g, 250 words), whereas summary length is bound to be variable

Dataset	2005	2006	2007	TD-QFS
Domain	Cross	Cross	Cross	Medical
Query Narrative	Long	Long	Long	Short
#Clusters	50	50	45	4
#Queries/Cluster	1	1	1	10
#Documents/Cluster	32	25	25	185
#Summaries/Query	4-9	4	4	3
#Words/Summary	250	250	250	250

Table 1: Multi-Document QFS dataset statistics.

in the training data. Inspired by Fan et al. (2018), we quantize summary length into discrete bins. We augment each training instance with this information, i.e., we prepend a length token (e.g., [230]) to document sentences. At inference, we inform the model of the summary budget by prepending the expected length token (e.g., [250]) to the sentences selected by the evidence ranker (see Figure 1(b)).

6 Experimental Setup

Datasets We performed experiments on the DUC 2005-2007 QFS benchmarks and TD-QFS (Baumel et al., 2016). DUC benchmarks contain long query narratives while TD-QFS focuses on medical texts with short keyword queries. Statistics for both datasets are given in Table 1. We used DUC 2005 as a development set to optimize hyperparameters and select abstractive models, and evaluated performance on the other three datasets.

We used Multi-News (Fabbri et al., 2019) and CNN/DailyMail (Hermann et al., 2015) as our generic summarization datasets to train MARGE (for evidence ranking) and to fine-tune UNILMV2 (for summary generation). Data statistics are shown in Table 2. To create the training and development sets for optimizing MARGE, we sampled sentences from each dataset. Specifically, we took the first and last 20 sentences from each cluster in Multi-News and the first and last three sentences from each article in CNN/DailyMail. For fine-tuning UNILMV2, we used the original Multi-News and the synthetic multi-document version of CNN/DailyMail described in Section 5.

Implementation Details We used the publicly released BERT model[3] and fine-tuned it for ROUGE regression with a learning rate of 3×10^{-5} and a batch size of 128 for 3 epochs on 8 GPUs (GTX 2080 Ti). We trained two summarization

---

[3]https://github.com/huggingface/pytorch-transformers

Query Modeling	Multi-News	CNN/DM
#Sentence/Doc	20	3
#Train	1,615,508	1,719,210
#Validation	200,824	80,052
#Words/Proxy Query	111.7	26.0
#Masks/Proxy Query	35.6	8.1

Summary Generation	Multi-News	CNN/DM
#Clusters	44,972	287,227
#Documents/Cluster	2.8	4.1
#Words/Summary	257.2	261.3

Table 2: Training data for query modeling and summary generation. CNN/DM statistics for summary generation refer to synthetic MDS dataset proposed in this work (based on CNN/DM).

Models	DUC 2006		DUC 2007		TD-QFS	
	R@10	R@30	R@10	R@30	R@10	R@30
ORACLE	6.7	16.2	8.4	19.1	17.2	35.6
TERMFREQ	7.2	15.1	8.5	18.5	14.2	25.9
BERTQA	8.5	16.3	10.2	20.2	9.8	21.9
BERTMRC	8.2	16.6	9.0	19.2	8.1	16.4
MARGE-MN	**11.1**	**20.2**	**13.8**	**25.3**	11.2	21.6
+EXPAND	—	—	—	—	**18.1**	**32.9**
MARGE-CD	9.1	17.4	11.1	22.1	10.0	18.7
+EXPAND	—	—	—	—	17.2	27.7

Table 3: Retrieval performance of evidence rankers. R@k is ROUGE-2 recall against top k sentences. MARGE models are trained on Multi-News (MN) and CNN/DailyMail (CD) datasets.

models on CNN/DailyMail and Multi-News, respectively, with the same hardware. For both models, we set the maximum input length to 768, and fine-tuned the publicly released UNILMV2 model[4] with a learning rate of 7×10^{-5} and a batch size of 16 for 40,000 steps with gradient accumulation every 4 steps. During decoding, we used beam search with beam size 5 and Trigram Blocking (Paulus et al., 2018) to reduce redundancy. The cosine similarity threshold for redundancy removal was set to 0.6 and summary length was discretized to 10 bins. The λ parameter for label smoothing was set to 0.15. We set γ, the parameter which modulates the proportion of information slots to reveal during masking, to 0 (see Appendix for detailed analysis of γ and its effect on model performance).

7 Results

Our experiments evaluate both components of the proposed approach, namely query modeling and summary generation. We assess the evidence ranker and the effectiveness of the unified masking. We also compare our summaries against competitive abstractive and extractive systems using automatic and human-based evaluation.

7.1 Query Modeling

Evaluation Metrics We evaluate query modeling with *retrieval* and *summarization* metrics. For the former evaluation, we follow Liu and Lapata (2019a), concatenate the top k ranked sentences, and calculate recall against gold summaries. We additionally propose to evaluate model output as

if it were an extractive summary, to better assess coverage and informativeness. We thus take the top sentences subject to a budget of 250 tokens, and remove redundancy by selecting sentences from the top and skipping sentences that have high cosine similarity (e.g., ≥ 0.6) with selected ones. We use ROUGE F1 to evaluate the resulting summaries so that *precision* is also taken into account.

Results We compare MARGE against Term Frequency, a simple but effective retrieval method that performs particularly well on DUC datasets (Katragadda and Varma, 2009). We also compare to two semantic matching models used for extractive QFS (Xu and Lapata, 2020): BERTQA which is trained on the joint set of WikiQA (Yang et al., 2015) and TrecQA (Yao et al., 2013), and BERTMRC which is fine-tuned on SQuAD 2.0 (Rajpurkar et al., 2018). ORACLE uses reference summaries as queries to retrieve summary sentences. For *summarization* evaluation, we report upper bound performance (GOLD) which we estimated by comparing a (randomly selected) reference summary against the remaining three reference summaries. In addition, we compare to LEAD which returns all lead sentences of the most recent document (up to 250 words) and LEXRANK (Erkan and Radev, 2004), a widely-used unsupervised method based on Markov random walks on sentence-similarity graphs.[5]

We summarize ranking and summarization results in Tables 3 and 4. As we can see, despite learning from weak signals, i.e., proxy queries and proxy answers, MARGE outperforms the strongest base-

---

[4]https://github.com/microsoft/unilm

[5]To examine ranking performance, we exclude multi-stage frameworks like Xu and Lapata (2020) that rerank the evidence with additional modules (e.g., centrality).

Models	DUC 2006	DUC 2007	TD-QFS
GOLD	17.0	19.1	—
ORACLE	14.8	16.0	23.0
LEAD	10.4	11.3	10.4
TERMFREQ	12.6	14.2	12.0
LEXRANK	11.4	12.7	12.2
BERTQA	13.9	14.9	16.1
BERTMRC	13.6	14.3	13.2
MARGE-MN	**14.5**	**16.6**	15.9
+EXPAND	—	—	**23.0**
MARGE-CD	13.9	15.8	16.9
+EXPAND	—	—	22.7

Table 4: Performance of evidence rankers on extractive QFS. We report the F1 score of R-SU4 (for the full set of ROUGE results, see Appendix).

Models	DUC 2006	DUC 2007	TD-QFS
MARGE-MN	14.5	16.6	23.0
−Verb	↓0.5	↓0.3	↓2.8
−Mask	↓0.8	↓1.2	↓1.5
−Query	↓2.9	↓2.9	↓12.6
−OpenIE	↓0.9	↓1.1	↓2.1

Table 5: Ablation results on training data (absolute performance decrease denoted by ↓).

Models	DUC 2006	DUC 2007	TD-QFS
PQSUM-WSL[†]	16.5	17.7	—
QUERYSUM*	15.3	16.8	20.7
BART-CAQ	12.9	14.4	—
PQSUM	14.8	16.0	—
UNILM-MN	11.8	12.3	12.9
UNILM-CD	13.6	14.9	16.7
MARGESUM-MN	14.3	16.5	16.5
MARGESUM-CD	**15.1**	**16.9**	**20.9**

Table 6: Abstractive summarization models with R-SU4 (full set of results in Appendix); ∗/†: extractive/supervised method.

line, BERTQA, under both evaluation tasks. Without recourse to any question/answer annotations or dataset-specific retrieval methods, our model provides more informative input to the downstream generation task. As anticipated, query expansion (+EXPAND) gives a big boost on TD-QFS (which has short queries) leading to better coverage.

Ablation Studies Table 5 shows the outcome of various ablation studies which assess the effectiveness of masking and how to best instantiate it. Specifically, −Verb additionally treats verbs as information slots for sampling and masking; −Mask removes masking entirely so that the whole summary is revealed; −Query removes the proxy query (at training time) and the actual query (at inference time); this is to investigate whether our model simply learns to judge sentence salience based on its own features, instead of performing semantic matching with the given query; −OpenIE removes the dependency on Open IE and chooses words to mask at random. Specifically, we randomly mask 15% words in summaries as in BERT (Devlin et al., 2019) and merge adjacent [MASK] tokens. Performance drops in all cases, especially when queries

are removed, underscoring the effectiveness of the proposed representation and training framework.

7.2 Abstractive Summarization

Automatic Evaluation Table 6 compares our model, which we call MARGESUM, against existing QFS systems. These include PQSUM-WSL (Laskar et al., 2020) a supervised abstractive system which represents the state of the art on DUC benchmarks. It first extracts relevant sentences for each document with a QA model, it then replaces some of these with reference summary sentences via a paraphrase model, and uses them to further fine-tune BERTSUM (Liu and Lapata, 2019b). In its supervised incarnation, two years' DUC datasets are used for training and one for testing. QUERYSUM (Xu and Lapata, 2020) is state-of-the-art extractive system which adopts a coarse-to-fine process for salience estimation.

The second block compares our model with two *distantly supervised* approaches. BART-CAQ (Su et al., 2020) uses an ensembled QA model to extract answer evidence, and fine-tuned BART (Lewis et al., 2020) to iteratively generate summaries from paragraphs. PQSUM (Laskar et al., 2020), uses fine-tuned BERTSUM to generate summaries for each document in a cluster, and a QA model to rank summary sentences against the query. Table 7 compares these models and our own in terms of their training requirements.

The third block presents the performance of UNILM fine-tuned on Multi-News and CNN/DailyMail following the standard setting in Bao et al. (2020). It uses no query guidance or length control. Documents are concatenated as input for training. During testing, sentences are selected with MARGE but ordered according to

Models	QA	PI	GS	QFS
BART-CAQ (Su et al., 2020)	✓	✗	✓	✗
PQSUM (Laskar et al., 2020)	✓	✗	✓	✗
PQSUM-WSL (Laskar et al., 2020)	✓	✓	✓	✓
UNILM (Bao et al., 2020)	✗	✗	✓	✗
MARGESUM	✗	✗	✓	✗

Table 7: Training requirements for existing QFS models (QA, PI, GS, and QFS stand for question answering, paraphrase identification, generic summarization and query focused summarization).

Models	DUC 2006	DUC 2007	TD-QFS
MARGE-CD	15.1	16.9	20.9
BERTQA	↓1.0	↓2.2	↓6.1
−Rank	↓1.7	↓3.1	↓1.3
−Length	↓0.1	↓0.5	↓0.2
−Query	↓0.5	↓0.3	↓0.4

Table 8: Ablations for MARGESUM trained on CNN/Daily Mail (performance decrease denoted by ↓).

their original document position. The last block shows two variants of MARGESUM, optimized on Multi-News and a synthetic training set built from CNN/DailyMail. Both take as input sentences selected with MARGE-MN during inference.

As we can see, without requiring expensive QA data (see Table 7), MARGESUM-CD outperforms existing distantly supervised approaches. Its performance on DUC is on par with one of the strongest extractive systems, while on TD-QFS it is superior across metrics. Also note that MARGE trained on synthetic MDS data outperforms MARGESUM-MN. Compared to Multi-News, synthetic summaries cover more topics and are less redundant, which is suited to QFS where there are usually multiple sub-queries to answer.

Ablation Studies Table 8 presents the results of several ablation studies on MARGESUM-CD. Replacing the input to the summarization component with sentences selected by BERTQA (Xu and Lapata, 2020) significantly decreases performance, demonstrating that sentences selected by MARGE are useful to downstream abstractive summarization. Removing evidence ranking altogether (−Rank) leads to a large performance drop; this is expected since sentence position information from the original documents does not transfer well to QFS settings. Removing length control (−Length) also hurts performance as does the removal of query guidance (−Query) at inference time.

DUC	Rel	Suc	Coh
GOLD	3.05	3.29	3.35
PQSUM-WSL	**2.95**	**3.27**	2.93[†][°]
QUERYSUM	2.79	3.13	2.94[†][°]
UNILM-CD	2.43[†][°]	3.09	3.27
MARGESUM-CD	2.91	3.25	**3.30**

TD-QFS	Rel	Suc	Coh
GOLD	4.70	4.23	4.60
QUERYSUM	4.32	3.90[°]	3.80[†][°]
UNILM-CD	3.63[†][°]	**4.12**	4.28
MARGESUM-CD	**4.55**	4.02	**4.37**

Table 9: Human evaluation results on DUC (above) and TD-QFS (below): average **Rel**evance, **Suc**cinctness, **Coh**erence ratings; †: sig different from MARGESUM-CD; °: sig different from Gold (at $p < 0.05$, using a pairwise t-test).

Human Evaluation We also evaluated model summaries in a judgment elicitation study via Amazon Mechanical Turk. Native English speakers (self-reported) were asked to rate query-summary pairs on two dimensions: Succinctness (does the summary avoid unnecessary detail and redundant information?) and Coherence (does the summary make logical sense?). The ratings were obtained using a fivepoint Likert scale. In addition, participants were asked to assess the Relevance of the summary to the query. Crowdworkers read a summary and for each sentence decided whether it is relevant (i.e., it provides an answer to the query), irrelevant (i.e., it does not answer the query), and partially relevant (i.e., it is not clear it directly answers the query). Relevant sentences were awarded a score of 5, partially relevant ones a score of 2.5, and 0 otherwise. Sentence scores were averaged to obtain a relevance score for the whole summary.

Participants assessed summaries created by PQSUM-WSL, the state-of-the-art abstractive system, QUERYSUM, a state-of-the-art extractive system, UNILM-CD, and MARGESUM-CD.[6] We also randomly selected GOLD standard summaries to include as an upper bound. We sampled 20 query-cluster pairs from DUC (2006, 2007; 10 from each set), and 20 pairs from TD-QFS (5 from each cluster) and collected three responses per pair.

---

[6]We are grateful to Md Tahmid Rahman Laskar for providing us with the output of their PQSUM-WSL system. We include PQSUM-WSL only for human evaluation on DUC since it was not evaluated on TD-QFS (Laskar et al., 2020) and system output is not available.

Table 9 shows the human ratings for each system (we provide examples of summary output in Appendix C). Participants perceive MARGESUM-CD on par with PQSUM-WSL in terms of query relevance and summary succinctness, while significantly better than PQSUM-WSL and QUERY-SUM in terms of coherence. In fact, participants find summaries PQSUM-WSL summaries as incoherent as those created by the extractive QUERY-SUM; this is probably due to the fact that PQSUM-WSL first generates an abstractive summary for each document and then re-ranks the generated sentences. Therefore, final summary sentences are less related to each other. Summaries from our system are also considered significantly more relevant than UNILM-CD. Compared to PQSUM-WSL, although UNILM-CD is not good at producing relevant content, it maintains relatively higher coherence, demonstrating the effectiveness of training abstractive systems with synthetic data from SDS and generating long summaries at once.

8 Conclusions

In this work we proposed an abstractive framework for query focused summarization. We provided a unified mask representation for summaries and queries, which enables summaries to serve as proxy queries for model training. As a result, a query model can be trained with generic summarization data without relying on additional question-answering resources. Experimental results across datasets show that the proposed system yields state-of-the-art performance despite the weakly supervised setting, and produces more relevant and coherent summaries compared to existing approaches. In the future, we would like to push this low-resource approach even further and attempt to generate abstractive summaries without access to any summarization datasets.

References

Rama Badrinath, Suresh Venkatasubramaniyan, and CE Veni Madhavan. 2011. Improving query focused summarization using look-ahead strategy. In *Proceedings of the 33rd European Conference on Advances in Information Retrieval*, pages 641–652, Dublin, Ireland.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. MS MARCO: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Jianfeng Gao, Songhao Piao, Ming Zhou, et al. 2020. UniLMv2: Pseudo-masked language models for unified language model pre-training. In *Proceedings of the 37rd International Conference on Machine Learning*, pages 642–652, Online.

Tal Baumel, Raphael Cohen, and Michael Elhadad. 2016. Topic concentration in query focused summarization datasets. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, pages 2573–2579, Phoenix, Arizona.

Souradip Chakraborty, Ekaba Bisong, Shweta Bhatt, Thomas Wagner, Riley Elliott, and Francesco Mosconi. 2020. BioMedBERT: A pre-trained biomedical language model for qa and ir. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 669–679, Online.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading Wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1870–1879, Vancouver, Canada.

Hoa Trang Dang. 2005. Overview of duc 2005. In *Proceedings of the 2005 Document Understanding Conference*, pages 1–12, Vancouver, Canada.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, Minneapolis, Minnesota.

William B Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing*, pages 9–16, Jeju Island, Korea.

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.

Alexander Richard Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir Radev. 2019. Multi-News: A large-scale multi-document summarization dataset and abstractive hierarchical model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1074–1084, Florence, Italy.

Angela Fan, David Grangier, and Michael Auli. 2018. Controllable abstractive summarization. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 45–54, Melbourne, Australia.

Sebastian Gehrmann, Yuntian Deng, and Alexander Rush. 2018. Bottom-up abstractive summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4098–4109, Brussels, Belgium.

Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, pages 55–64, Indianapolis, Indiana.

Karl Moritz Hermann, Tomáš Kočiský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, page 1693–1701, Cambridge, MA, USA.

TD Hoa. 2006. Overview of duc 2006. In *Proceedings of the 2006 Document Understanding Conference*, New York, USA.

Luyang Huang, Lingfei Wu, and Lu Wang. 2020. Knowledge graph-augmented abstractive summarization with semantic-driven cloze reward. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5094–5107, Online.

Rahul Katragadda and Vasudeva Varma. 2009. Query-focused summaries or query-biased summaries? In *Proceedings of the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing*, pages 105–108, Suntec, Singapore.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin,

Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Md Tahmid Rahman Laskar, Enamul Hoque, and Jimmy Xiangji Huang. 2020. WSL-DS: Weakly supervised learning with distant supervision for query focused multi-document abstractive summarization. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5647–5654, Online.

Logan Lebanoff, Kaiqiang Song, and Fei Liu. 2018. Adapting the neural encoder-decoder framework from single to multi-document summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4131–4141, Brussels, Belgium.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online.

Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel. 2019. Unsupervised question answering by cloze translation. *arXiv preprint arXiv:1906.04980*.

Piji Li, Wai Lam, Lidong Bing, Weiwei Guo, and Hang Li. 2017a. Cascaded attention based unsupervised information distillation for compressive summarization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2081–2090, Brussells, Belgium.

Piji Li, Zihao Wang, Wai Lam, Zhaochun Ren, and Lidong Bing. 2017b. Salience estimation via variational auto-encoders for multi-document summarization. In *Proceedings of the 31th AAAI Conference on Artificial Intelligence*, pages 3497–3503, San Francisco, California, USA.

Yang Liu and Mirella Lapata. 2019a. Hierarchical transformers for multi-document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5070–5081, Florence, Italy.

Yang Liu and Mirella Lapata. 2019b. Text summarization with pretrained encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3730–3740, Hong Kong, China.

Preksha Nema, Mitesh M. Khapra, Anirban Laha, and Balaraman Ravindran. 2017. Diversity driven attention model for query-based abstractive summarization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1063–1072, Vancouver, Canada.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *Proceedings of the 6th International Conference on Learning Representations*, Vancouver, Canada.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 784–789, Austin, Texas.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Sydney, Australia.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pages 1073–1083, Vancouver, Canada.

Gabriel Stanovsky, Julian Michael, Luke Zettlemoyer, and Ido Dagan. 2018. Supervised open information extraction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 885–895, New Orleans, Louisiana.

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

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.

Wilson L Taylor. 1953. "Cloze Procedure": A new tool for measuring readability. *Journalism Quarterly*, 30(4):415–433.

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 *Advances in Neural Information Processing Systems*, pages 6000–6010.

Xiaojun Wan, Jianwu Yang, and Jianguo Xiao. 2007. Manifold-ranking based topic-focused multi-document summarization. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 2903–2908, Hyderabad, India.

Xiaojun Wan and Jianmin Zhang. 2014. CTSUM: extracting more certain summaries for news articles. In *Proceedings of the 37th international ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 787–796, New York, United States.

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*, pages 3632–3645, Online.

Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. WikiQA: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2013–2018, Lisbon, Portugal.

Xuchen Yao, Benjamin Van Durme, Chris Callison-Burch, and Peter Clark. 2013. Answer extraction as sequence tagging with tree edit distance. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 858–867, Atlanta, Georgia.

Jianmin Zhang, Jiwei Tan, and Xiaojun Wan. 2018. Neural single-document summarization model for abstractive multi-document summarization: A pilot studyadapting. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 381–390, Tilburg University, The Netherlands.

A Evidence Ranking Results

Models	DUC 2006	DUC 2007	TD-QFS
ORACLE	22.7	26.2	44.6
TERMFREQ	20.8	25.2	34.0
BERTQA	22.1	26.1	29.1
BERTMRC	22.3	25.2	23.2
MARGE-MN	**25.9**	**31.8**	29.4
+EXPAND	—	—	**39.1**
MARGE-CD	23.3	28.8	26.2
+EXPAND	—	—	26.2

Table 10: Performance of evidence rankers on top retrieval. We report the ROUGE 2 recall score for the concatenation of the top 50 retrieved sentences.

We show in the paper the top k retrieval performance of different models when $k \in \{10, 30\}$. In some cases, when top sentences are relatively short, the maximum input length to UNILM (which is set to 768) allows for more than 30 sentences to be selected. Therefore, in Table 3, we further show the top k retrieval performance of evidence rankers with larger k, set to $k = 50$. Results show that our model outperforms strong baseline systems, and we conclude that it consistently provides high quality content, under varied budgets ($k \in \{10, 30, 50\}$), to the downstream abstractive summarization task.

We report the full set of ROUGE results for evidence rankers on extractive summarization in the main paper in Table 4.

B The Effect of Reveal Ratio

We show how the mask reveal ratio γ affects model performance in Figure 2. As we can see, performance on the ROUGE regression task improves as γ increases; this is not surprising, the task becomes easier when fewer tokens are masked; when $\gamma = 1.0$, simply counting lexical overlap can solve the task perfectly. However, model performance on the QFS development set (DUC 2005) shows the opposite trend: actual queries *seek* information, instead of providing all the information needed. Therefore, the model is required to perform *semantic matching* (Guo et al., 2016) to accurately estimate evidence scores. Based on our empirical results, a simple but effective strategy is to mask all information slots (i.e., potential arguments) and reveal the rest of the words (including verbs) in the summary to construct proxy queries for training.
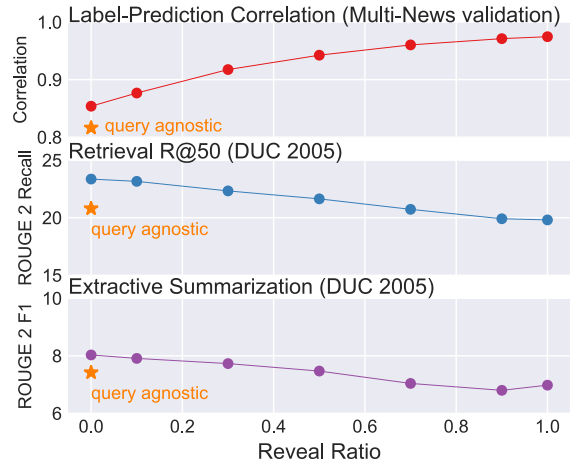

Figure 2: Model performance when reveal ratio γ is varied. Correlation refers to the average of Pearson's r correlation. The star marker denotes query-agnostic performance where all query tokens are masked, including information slots.

C Abstractive Summarization Results

We report the full set of ROUGE results for abstractive summarization models in Table 12. We also show an example of system outputs in Table 13.

D Datasets and Evaluation Package

Multi-News and CNN/Daily Mail are used to train the query model and abstractive summarization model described in this work, and they can be downloaded from `https://github.com/Alex-Fabbri/Multi-News` and `https://github.com/abisee/cnn-dailymail`, respectively.

For evaluation purposes, the TD-QFS dataset is publicly available at `https://www.cs.bgu.ac.il/~talbau/TD-QFS/dataset.html`. DUC 2005-2007 benchmarks can be requested from NIST: `https://www-nlpir.nist.gov/projects/duc/data.html`.

We computed ROUGE scores with `pyrouge`, a Python wrapper for the ROUGE summarization evaluation package: `https://github.com/bheinzerling/pyrouge`.

Models	DUC 2006			DUC 2007			TD-QFS		
	R-1	R-2	R-SU4	R-1	R-2	R-SU4	R-1	R-2	R-SU4
GOLD	45.7	11.2	17.0	47.9	14.1	19.1	-	-	-
ORACLE	40.6	9.1	14.8	41.8	10.4	16.0	44.9	18.9	23.0
LEAD	32.1	5.3	10.4	33.4	6.5	11.3	33.5	5.2	10.4
TERMFREQ	36.5	7.0	12.6	38.5	9.0	14.2	35.7	6.5	12.0
LEXRANK	34.2	6.4	11.4	35.8	7.7	12.7	35.3	7.6	12.2
BERTQA	38.6	8.4	13.9	39.8	10.0	14.9	39.5	10.5	16.1
BERTMRC	**39.6**	7.8	13.6	39.9	8.9	14.3	36.6	8.4	13.2
MARGE-MN	39.0	**9.3**	**14.5**	**41.6**	**11.6**	**16.6**	38.8	10.5	15.9
+EXPAND	—	—	—	—	—	—	**45.9**	**18.8**	**23.0**
MARGE-CD	38.4	8.6	13.9	40.7	10.8	15.8	40.1	11.6	16.9
+EXPAND	—	—	—	—	—	—	**45.9**	18.3	22.7

Table 11: Performance of evidence rankers on extractive QFS. R-1, R-2 and R-SU4 stand for the F1 score of ROUGE 1, 2, and SU4, respectively.

Models	DUC 2006			DUC 2007			TD-QFS		
	R-1	R-2	R-SU4	R-1	R-2	R-SU4	R-1	R-2	R-SU4
PQSUM-WSL[†] (Laskar et al., 2020)	43.5	10.8	16.5	44.7	12.4	17.7	-	-	-
QUERYSUM* (Xu and Lapata, 2020)	41.6	9.5	15.3	43.3	11.6	16.8	44.3	16.1	20.7
BART-CAQ (Su et al., 2020)	38.3	7.7	12.9	40.5	9.2	14.4	-	-	-
PQSUM (Laskar et al., 2020)	**40.9**	9.4	14.8	42.2	10.8	16.0	-	-	-
UNILM-MN	34.6	6.7	11.8	35.5	7.6	12.3	36.2	8.1	12.9
UNILM-CD	37.6	8.3	13.6	39.6	10.1	14.9	40.1	11.8	16.7
MARGESUM-MN	39.1	9.1	14.3	42.1	11.7	16.5	40.8	11.6	16.5
MARGESUM-CD	40.2	**9.7**	**15.1**	**42.5**	**12.0**	**16.9**	**45.5**	**16.6**	**20.9**

Table 12: Performance of abstractive summarization systems. R-1, R-2 and R-SU4 stand for the F1 score of ROUGE 1, 2, and SU4, respectively. */†: extractive/supervised method.

Query: Steroid use among female athletes. Discuss the prevalence of steroid use among female athletes over the years. Include information regarding trends, side effects and consequences of such use.

GOLD: Increasing numbers of female athletes use anabolic steroids. Among female college athletes, an estimated 5% in swimming, 6% in basketball and 10% in track and field have used steroids. The fastest growing group of users is adolescent females. New steroids are often marketed as dietary supplements and girls are unaware of what they are taking. A young tennis player tested positive for steroids after taking an "all natural" supplement later discovered to contain a variation on the banned steroid nandrolone. A former Olympic track coach believes at least 40% of the U.S. women's team in Seoul had used steroids at some time. Because of their naturally low testosterone levels, steroids have a more dramatic effect on women, boosting their levels up to 10 times. The health effects can be dramatic, including liver damage and tumors, elevated cholesterol, heart attacks, strokes, stunted growth in adolescents, infertility, uncontrollably violent behavior, chronic depression, deeper voices, excessive facial hair, and acne. Many East German female athletes were given steroids, often unbeknownst to them or their parents and suffered serious side-effects. Athletes who have tested positive for steroids include a Bulgarian triple jumper, a Romanian hammer thrower, a Russian runner, a Dominican high, a Jamaican sprinter, a Spanish pole vaulter, a German marathon runner, two Moroccan athletes, and two Chinese Taipei weightlifters. A number of female Chinese athletes have tested positive, including several swimmers and rowers. International organizations impose bans of between two months and life on athletes found guilty of using illegal steroids.

UNILM-CD: At least 23 Chinese athletes, mostly women, have tested positive for steroid use since 1990. Such side-effects of anabolic steroids have been generally known since 1977. Some of the athletes were as young as 12 when they started taking the steroids. EPO is one of at least 20 steroids prohibited by the International Olympic Committee. The drug is legal and sold over-the-counter, but is banned by the National Olympic Committee, the National Football League and the National Collegiate Athletic Association. The National College Athletic Association (ACA) has banned EPO for life. EPO has been linked to a number of health problems, including liver damage and liver cancer. The ACA says EPO's use in the U.S. is safe, effective and safe to use. The National Institute of Drug Abuse (NIDA) is investigating the use of EPO in the United States. The NIDA says it has no evidence to support the claim that EPO can cause liver damage. The drug was approved for use by the U.S. Olympic committee in 1996. The National Association of College Athletes (NAA) is calling for the drug to be banned from all U.S. sports. The NAA says it's time for the U.S. Olympic Committee (IOC) to ban EPO. The drug has been used by more than 200,000 athletes worldwide since it was approved in 1996.

PQSUM: About 175,000 teen age girls in the United States have reported using anabolic steroids at least once within a year of the time surveyed a rise of 100 percent since 1991. In 1996 Samantha Reeves became the first female tennis player to have tested positive for steroids. The rise of steroid use in female athletes has led to a rise in women's steroid use. Over the years the number of female steroid related women in the U.S has increased There is no mistaking the dangers of steroids for women with documentation provided by trainers and coaches from the former east german sports empire. In 1996 women were banned from the olympics for a positive test of methandienone. The women involved suffered serious side effects from the drugs and that they knew of the potential danger. She was found to have steroid stanozolol in her system. In the last month Cuban sprinter Merlene Ottey withdrew from the world.championships after testing positive for nandrolone. Some were as young as 12 when they started taking the anabolic steroid pills. The drug use has been generally known since 1977. Pospelova would be the seventh athlete to test positive at the games. Such side effects of anabolic steroids are generally known as with all probability linked to doping. An over the counter supplement called andro raises testosterone and estrogen above normal levels and could be dangerous according to a harvard study by major league baseball and its players union. In 1996.

MARGESUM-CD: Penn State professor Charles Yesalis estimates the use of steroids among female athletes at 5 percent in swimming, 6 percent in basketball and 10 percent in track and field, a rise of 100 percent since 1991. The national institutes of drug abuse says 175,000 teenage girls in the united states have reported taking anabolic steroids at least once within a year of the time surveyed. The national institute on drug abuse provides information regarding trends, side effects and consequences of such use. Two Moroccan female athletes have been stripped of gold and bronze medals for using a muscle-building steroid in the first reported cases of doping at the Arab games for using the steroid nandrolone, a steroid that has been linked to liver cancer, heart disease and uncontrollable aggressiveness. Two medical experts testifying in the doping trial of a former east german sports doctor say the female swimmers they examined showed health damage linked to performance-enhancing drugs, including liver damage and excessive facial hair. The study, published in Wednesday's Journal of the American Medical Association, is the first to conclude that high doses of the steroids can elevate testosterone levels and that the hormone can be used as a performance-enhancing steroid, such as epitestosterone, as a marker the testosterone is 6 to 1 in the male sex hormone and 5 to 1 for the female steroid hormone epitestoterone - a metabolite that is used as an indicator of testosterone use - the female sex hormone.

Table 13: System outputs for cluster D0602C in DUC 2006. The gold summary answers the query covering four main aspects (denoted with different colors): (1) trend; (2) side-effects; (3) consequences of such use; (4) historical cases. Both outputs from MARGESUM-CD and PQSUM have a good coverage of the main query focuses. Compared to PQSUM, MARGESUM-CD produces a more coherent summary for the given query narrative with a more natural topic flow.