Multi-News: a Large-Scale Multi-Document Summarization Dataset and Abstractive Hierarchical Model

Abstract

Automatic generation of summaries from multiple news articles is a valuable tool as the number of online publications grows rapidly. Single document summarization (SDS) systems have benefited from advances in neural encoder-decoder model thanks to the availability of large datasets. However, multi-document summarization (MDS) of news articles has been limited to datasets of a couple of hundred examples. In this paper, we introduce Multi-News, the first large-scale MDS news dataset. Additionally, we propose an end-to-end model which incorporates a traditional extractive summarization model with a standard SDS model and achieves competitive results on MDS datasets. We benchmark several methods on Multi-News and release our data and code in hope that this work will promote advances in summarization in the multi-document setting.

Introduction

Summarization is a central problem in Natural Language Processing with increasing applications as the desire to receive content in a concise and easily-understood format increases. Recent advances in neural methods for text summarization have largely been applied in the setting of single-document news summarization and headline generation BIBREF0 , BIBREF1 , BIBREF2 . These works take advantage of large datasets such as the Gigaword Corpus BIBREF3 , the CNN/Daily Mail (CNNDM) dataset BIBREF4 , the New York Times dataset BIBREF5 and the Newsroom corpus BIBREF6 , which contain on the order of hundreds of thousands to millions of article-summary pairs. However, multi-document summarization (MDS), which aims to output summaries from document clusters on the same topic, has largely been performed on datasets with less than 100 document clusters such as the DUC 2004 BIBREF7 and TAC 2011 BIBREF8 datasets, and has benefited less from advances in deep learning methods.

Multi-document summarization of news events offers the challenge of outputting a well-organized summary which covers an event comprehensively while simultaneously avoiding redundancy. The input documents may differ in focus and point of view for an event. We present an example of multiple input news documents and their summary in Figure TABREF2 . The three source documents discuss the same event and contain overlaps in content: the fact that Meng Wanzhou was arrested is stated explicitly in Source 1 and 3 and indirectly in Source 2. However, some sources contain information not mentioned in the others which should be included in the summary: Source 3 states that (Wanzhou) is being sought for extradition by the US while only Source 2 mentioned the attitude of the Chinese side.

Recent work in tackling this problem with neural models has attempted to exploit the graph structure among discourse relations in text clusters BIBREF9 or through an auxiliary text classification task BIBREF10 . Additionally, a couple of recent papers have attempted to adapt neural encoder decoder models trained on single document summarization datasets to MDS BIBREF11 , BIBREF12 , BIBREF13 .

However, data sparsity has largely been the bottleneck of the development of neural MDS systems. The creation of large-scale multi-document summarization dataset for training has been restricted due to the sparsity and cost of human-written summaries. liu18wikisum trains abstractive sequence-to-sequence models on a large corpus of Wikipedia text with citations and search engine results as input documents. However, no analogous dataset exists in the news domain. To bridge the gap, we introduce Multi-News, the first large-scale MDS news dataset, which contains 56,216 articles-summary pairs. We also propose a hierarchical model for neural abstractive multi-document summarization, which consists of a pointer-generator network BIBREF1 and an additional Maximal Marginal Relevance (MMR) BIBREF14 module that calculates sentence ranking scores based on relevancy and redundancy. We integrate sentence-level MMR scores into the pointer-generator model to adapt the attention weights on a word-level. Our model performs competitively on both our Multi-News dataset and the DUC 2004 dataset on ROUGE scores. We additionally perform human evaluation on several system outputs.

Our contributions are as follows: We introduce the first large-scale multi-document summarization datasets in the news domain. We propose an end-to-end method to incorporate MMR into pointer-generator networks. Finally, we benchmark various methods on our dataset to lay the foundations for future work on large-scale MDS.

Related Work

Traditional non-neural approaches to multi-document summarization have been both extractive BIBREF14 , BIBREF15 , BIBREF16 , BIBREF17 , BIBREF18 as well as abstractive BIBREF19 , BIBREF20 , BIBREF21 , BIBREF22 . Recently, neural methods have shown great promise in text summarization, although largely in the single-document setting, with both extractive BIBREF23 , BIBREF24 , BIBREF25 and abstractive methods BIBREF26 , BIBREF27 , BIBREF1 , BIBREF28 , BIBREF29 , BIBREF30 , BIBREF2

In addition to the multi-document methods described above which address data sparsity, recent work has attempted unsupervised and weakly supervised methods in non-news domains BIBREF31 , BIBREF32 . The methods most related to this work are SDS adapted for MDS data. zhang18mds adopts a hierarchical encoding framework trained on SDS data to MDS data by adding an additional document-level encoding. baumel18mds incorporates query relevance into standard sequence-to-sequence models. lebanoff18mds adapts encoder-decoder models trained on single-document datasets to the MDS case by introducing an external MMR module which does not require training on the MDS dataset. In our work, we incorporate the MMR module directly into our model, learning weights for the similarity functions simultaneously with the rest of the model.

Multi-News Dataset

Our dataset, which we call Multi-News, consists of news articles and human-written summaries of these articles from the site newser.com. Each summary is professionally written by editors and includes links to the original articles cited. We will release stable Wayback-archived links, and scripts to reproduce the dataset from these links. Our dataset is notably the first large-scale dataset for MDS on news articles. Our dataset also comes from a diverse set of news sources; over 1,500 sites appear as source documents 5 times or greater, as opposed to previous news datasets (DUC comes from 2 sources, CNNDM comes from CNN and Daily Mail respectively, and even the Newsroom dataset BIBREF6 covers only 38 news sources). A total of 20 editors contribute to 85% of the total summaries on newser.com. Thus we believe that this dataset allows for the summarization of diverse source documents and summaries.

Statistics and Analysis

The number of collected Wayback links for summaries and their corresponding cited articles totals over 250,000. We only include examples with between 2 and 10 source documents per summary, as our goal is MDS, and the number of examples with more than 10 sources was minimal. The number of source articles per summary present, after downloading and processing the text to obtain the original article text, varies across the dataset, as shown in Table TABREF4 . We believe this setting reflects real-world situations; often for a new or specialized event there may be only a few news articles. Nonetheless, we would like to summarize these events in addition to others with greater news coverage.

We split our dataset into training (80%, 44,972), validation (10%, 5,622), and test (10%, 5,622) sets. Table TABREF5 compares Multi-News to other news datasets used in experiments below. We choose to compare Multi-News with DUC data from 2003 and 2004 and TAC 2011 data, which are typically used in multi-document settings. Additionally, we compare to the single-document CNNDM dataset, as this has been recently used in work which adapts SDS to MDS BIBREF11 . The number of examples in our Multi-News dataset is two orders of magnitude larger than previous MDS news data. The total number of words in the concatenated inputs is shorter than other MDS datasets, as those consist of 10 input documents, but larger than SDS datasets, as expected. Our summaries are notably longer than in other works, about 260 words on average. While compressing information into a shorter text is the goal of summarization, our dataset tests the ability of abstractive models to generate fluent text concise in meaning while also coherent in the entirety of its generally longer output, which we consider an interesting challenge.

Diversity

We report the percentage of n-grams in the gold summaries which do not appear in the input documents as a measure of how abstractive our summaries are in Table TABREF6 . As the table shows, the smaller MDS datasets tend to be more abstractive, but Multi-News is comparable and similar to the abstractiveness of SDS datasets. Grusky:18 additionally define three measures of the extractive nature of a dataset, which we use here for a comparison. We extend these notions to the multi-document setting by concatenating the source documents and treating them as a single input. Extractive fragment coverage is the percentage of words in the summary that are from the source article, measuring the extent to which a summary is derivative of a text: DISPLAYFORM0

where A is the article, S the summary, and INLINEFORM0 the set of all token sequences identified as extractive in a greedy manner; if there is a sequence of source tokens that is a prefix of the remainder of the summary, that is marked as extractive. Similarly, density is defined as the average length of the extractive fragment to which each summary word belongs: DISPLAYFORM0

Finally, compression ratio is defined as the word ratio between the articles and its summaries: DISPLAYFORM0

These numbers are plotted using kernel density estimation in Figure FIGREF11 . As explained above, our summaries are larger on average, which corresponds to a lower compression rate. The variability along the x-axis (fragment coverage), suggests variability in the percentage of copied words, with the DUC data varying the most. In terms of y-axis (fragment density), our dataset shows variability in the average length of copied sequence, suggesting varying styles of word sequence arrangement. Our dataset exhibits extractive characteristics similar to the CNNDM dataset.

Other Datasets

As discussed above, large scale datasets for multi-document news summarization are lacking. There have been several attempts to create MDS datasets in other domains. zopf18mds introduce a multi-lingual MDS dataset based on English and German Wikipedia articles as summaries to create a set of about 7,000 examples. liu18wikisum use Wikipedia as well, creating a dataset of over two million examples. That paper uses Wikipedia references as input documents but largely relies on Google search to increase topic coverage. We, however, are focused on the news domain, and the source articles in our dataset are specifically cited by the corresponding summaries. Related work has also focused on opinion summarization in the multi-document setting; angelidis18opinions introduces a dataset of 600 Amazon product reviews.

Preliminaries

We introduce several common methods for summarization.

Pointer-generator Network

The pointer-generator network BIBREF1 is a commonly-used encoder-decoder summarization model with attention BIBREF33 which combines copying words from source documents and outputting words from a vocabulary. The encoder converts each token INLINEFORM0 in the document into the hidden state INLINEFORM1 . At each decoding step INLINEFORM2 , the decoder has a hidden state INLINEFORM3 . An attention distribution INLINEFORM4 is calculated as in BIBREF33 and is used to get the context vector INLINEFORM5 , which is a weighted sum of the encoder hidden states, representing the semantic meaning of the related document content for this decoding time step:

DISPLAYFORM0

The context vector INLINEFORM0 and the decoder hidden state INLINEFORM1 are then passed to two linear layers to produce the vocabulary distribution INLINEFORM2 . For each word, there is also a copy probability INLINEFORM3 . It is the sum of the attention weights over all the word occurrences:

DISPLAYFORM0

The pointer-generator network has a soft switch INLINEFORM0 , which indicates whether to generate a word from vocabulary by sampling from INLINEFORM1 , or to copy a word from the source sequence by sampling from the copy probability INLINEFORM2 .

DISPLAYFORM0

where INLINEFORM0 is the decoder input. The final probability distribution is a weighted sum of the vocabulary distribution and copy probability:

$$P(w) = pgenPvocab(w) + (1-pgen)Pcopy(w)$$

Transformer

The Transformer model replaces recurrent layers with self-attention in an encoder-decoder framework and has achieved state-of-the-art results in machine translation BIBREF34 and language modeling BIBREF35 , BIBREF36 . The Transformer has also been successfully applied to SDS BIBREF2 . More specifically, for each word during encoding, the multi-head self-attention sub-layer allows the encoder to directly attend to all other words in a sentence in one step. Decoding contains the typical encoder-decoder attention mechanisms as well as self-attention to all previous generated output. The Transformer motivates the elimination of recurrence to allow more direct interaction among words in a sequence.

MMR

Maximal Marginal Relevance (MMR) is an approach for combining query-relevance with information-novelty in the context of summarization BIBREF14 . MMR produces a ranked list of the candidate sentences based on the relevance and redundancy to the query, which can be used to extract sentences. The score is calculated as follows:

MMR=*argmax D i RS [Sim 1 (D i ,Q)-(1-) D j S Sim2 (D i ,D j)] where INLINEFORM0 is the collection of all candidate sentences, INLINEFORM1 is the query, INLINEFORM2 is the set of sentences that have been selected, and INLINEFORM3 is set of the un-selected ones. In general, each time we want to select a sentence, we have a ranking score for all the candidates that considers relevance and redundancy. A recent work BIBREF11 applied MMR for multi-document summarization by creating an external module and a supervised regression model for sentence importance. Our proposed method, however, incorporates MMR with the pointer-generator network in an end-to-end manner that learns parameters for similarity and redundancy.

Hi-MAP Model

In this section, we provide the details of our Hierarchical MMR-Attention Pointer-generator (Hi-MAP) model for multi-document neural abstractive summarization. We expand the existing pointer-generator network model into a hierarchical network, which allows us to calculate sentence-level MMR scores. Our model consists of a pointer-generator network and an integrated MMR module, as shown in Figure FIGREF19 .

Sentence representations

To expand our model into a hierarchical one, we compute sentence representations on both the encoder and decoder. The input is a collection of sentences INLINEFORM0 from all the source documents, where a given sentence INLINEFORM1 is made up of input word tokens. Word tokens from the whole document are treated as a single sequential input to a Bi-LSTM encoder as in the original encoder of the pointer-generator network from see2017ptrgen (see bottom of Figure FIGREF19). For each time step, the output of an input word token INLINEFORM2 is INLINEFORM3 (we use superscript INLINEFORM4 to indicate word-level LSTM cells, INLINEFORM5 for sentence-level).

To obtain a representation for each sentence INLINEFORM0 , we take the encoder output of the last token for that sentence. If that token has an index of INLINEFORM1 in the whole document INLINEFORM2 , then the sentence representation is marked as INLINEFORM3 . The word-level sentence embeddings of the document INLINEFORM4 will be a sequence which is fed into a sentence-level LSTM network. Thus, for each input sentence INLINEFORM5 , we obtain an output hidden state INLINEFORM6 . We then get the final sentence-level embeddings INLINEFORM7 (we omit the subscript for sentences INLINEFORM8). To obtain a summary representation, we simply treat the current decoded summary as a single sentence and take the output of the last step of the decoder:

INLINEFORM9 . We plan to investigate alternative methods for input and output sentence embeddings, such as separate LSTMs for each sentence, in future work.

MMR-Attention

Now, we have all the sentence-level representation from both the articles and summary, and then we apply MMR to compute a ranking on the candidate sentences INLINEFORM0 . Intuitively, incorporating MMR will help determine salient sentences from the input at the current decoding step based on relevancy and redundancy.

We follow Section 4.3 to compute MMR scores. Here, however, our query document is represented by the summary vector INLINEFORM0 , and we want to rank the candidates in INLINEFORM1 . The MMR score for an input sentence INLINEFORM2 is then defined as:

MMR i = Sim 1 (hs i ,ssum)-(1-) sj D, j i Sim2 (hs i ,hs j) We then add a softmax function to normalize all the MMR scores of these candidates as a probability distribution. MMR i = (MMR i)i(MMR i) Now we define the similarity function between each candidate sentence INLINEFORM0 and summary sentence INLINEFORM1 to be: DISPLAYFORM0

where INLINEFORM0 is a learned parameter used to transform INLINEFORM1 and INLINEFORM2 into a common feature space.

For the second term of Equation SECREF21 , instead of choosing the maximum score from all candidates except for INLINEFORM0 , which is intended to find the candidate most similar to INLINEFORM1 , we choose to apply a self-attention model on INLINEFORM2 and all the other candidates INLINEFORM3 . We then choose the largest weight as the final score:

DISPLAYFORM0

Note that INLINEFORM0 is also a trainable parameter. Eventually, the MMR score from Equation SECREF21 becomes:

MMR i = Sim 1 (hsi,ssum)-(1-) scorei

MMR-attention Pointer-generator

After we calculate INLINEFORM0 for each sentence representation INLINEFORM1 , we use these scores to update the word-level attention weights for the pointer-generator model shown by the blue arrows in Figure FIGREF19 . Since INLINEFORM2 is a sentence weight for INLINEFORM3 , each token in the sentence will have the same value of INLINEFORM4 . The new attention for each input token from Equation EQREF14 becomes: DISPLAYFORM0

Experiments

In this section we describe additional methods we compare with and present our assumptions and experimental process.

Baseline and Extractive Methods

First We concatenate the first sentence of each article in a document cluster as the system summary. For our dataset, First- INLINEFORM0 means the first INLINEFORM1 sentences from each source article will be concatenated as the summary. Due to the difference in gold summary length, we only use First-1 for DUC, as others would exceed the average summary length.

LexRank Initially proposed by BIBREF16 , LexRank is a graph-based method for computing relative importance in extractive summarization.

TextRank Introduced by BIBREF17 , TextRank is a graph-based ranking model. Sentence importance scores are computed based on eigenvector centrality within a global graph from the corpus.

MMR In addition to incorporating MMR in our pointer generator network, we use this original method as an extractive summarization baseline. When testing on DUC data, we set these extractive methods to give an output of 100 tokens and 300 tokens for Multi-News data.

Neural Abstractive Methods

PG-Original, PG-MMR These are the original pointer-generator network models reported by BIBREF11 .

PG-BRNN The PG-BRNN model is a pointer-generator implementation from OpenNMT. As in the original paper BIBREF1 , we use a 1-layer bi-LSTM as encoder, with 128-dimensional word-embeddings and 256-dimensional hidden states for each direction. The decoder is a 512-dimensional single-layer LSTM. We include this for reference in addition to PG-Original, as our Hi-MAP code builds upon this implementation.

CopyTransformer Instead of using an LSTM, the CopyTransformer model used in Gehrmann:18 uses a 4-layer Transformer of 512 dimensions for encoder and decoder. One of the attention heads is chosen randomly as the copy distribution. This model and the PG-BRNN are run without the bottom-up masked attention for inference from Gehrmann:18 as we did not find a large improvement when reproducing the model on this data.

Experimental Setting

Following the setting from BIBREF11 , we report ROUGE BIBREF37 scores, which measure the overlap of unigrams (R-1), bigrams (R-2) and skip bigrams with a max distance of four words (R-SU). For the neural abstractive models, we truncate input articles to 500 tokens in the following way: for each example with INLINEFORM0 source input documents, we take the first 500 INLINEFORM1 tokens from each source document. As some source documents may be shorter, we iteratively determine the number of tokens to take from each document until the 500 token quota is reached. Having determined the number of tokens per source document to use, we concatenate the truncated source documents into a single mega-document. This effectively reduces MDS to SDS on longer documents, a commonly-used assumption for recent neural MDS papers BIBREF10 , BIBREF38 , BIBREF11 . We chose 500 as our truncation size as related MDS work did not find significant improvement when increasing input length from 500 to 1000 tokens BIBREF38 . We simply introduce a special token between source documents to aid our models in detecting document-to-document relationships and leave direct modeling of this relationship, as well as modeling longer input sequences, to future work. We hope that the dataset we introduce will promote such work. For our Hi-MAP model, we applied a 1-layer bidirectional LSTM network, with the hidden state dimension 256 in each direction. The sentence representation dimension is also 256. We set the INLINEFORM2 to calculate the MMR value in Equation SECREF21 .

As our focus was on deep methods for MDS, we only tested several non-neural baselines. However, other classical methods deserve more attention, for which we refer the reader to Hong14 and leave the implementation of these methods on Multi-News for future work.

Analysis and Discussion

In Table TABREF30 and Table TABREF31 we report ROUGE scores on DUC 2004 and Multi-News datasets respectively. We use DUC 2004, as results on this dataset are reported in lebanoff18mds, although this dataset is not the focus of this work. For results on DUC 2004, models were trained on the CNNDM dataset, as in lebanoff18mds. PG-BRNN and CopyTransformer models, which were pretrained by OpenNMT on CNNDM, were applied to DUC without additional training, analogous to PG-Original. We also experimented with training on Multi-News and testing on DUC data, but we did not see significant improvements. We attribute the generally low performance of pointer-generator, CopyTransformer and Hi-MAP to domain differences between DUC and CNNDM as well as DUC and Multi-News. These domain differences are evident in the statistics and extractive metrics discussed in Section 3.

Additionally, for both DUC and Multi-News testing, we experimented with using the output of 500 tokens from extractive methods (LexRank, TextRank and MMR) as input to the abstractive model. However, this did not improve results. We believe this is because our truncated input mirrors the First-3 baseline, which outperforms these three extractive methods and thus may provide more information as input to the abstractive model.

Our model outperforms PG-MMR when trained and tested on the Multi-News dataset. We see much-improved model performances when trained and tested on in-domain Multi-News data. The Transformer performs best in terms of R-1 while Hi-MAP outperforms it on R-2 and R-SU. Also, we notice a drop in performance between PG-original, and PG-MMR (which takes the pre-trained PG-original and applies MMR on top of the model). Our PG-MMR results correspond to PG-MMR w Cosine reported in lebanoff18mds. We trained their sentence regression model on Multi-News data and leave the investigation of transferring regression models from SDS to Multi-News for future work.

In addition to automatic evaluation, we performed human evaluation to compare the summaries produced. We used Best-Worst Scaling BIBREF39 , BIBREF40 , which has shown to be more reliable than rating scales BIBREF41 and has been used to evaluate summaries BIBREF42 , BIBREF32 . Annotators were presented with the same input that the systems saw at testing time; input documents were truncated, and we separated input documents by visible spaces in our annotator interface. We chose three native English speakers as annotators. They were presented with input documents, and summaries generated by two out of four systems, and were asked to determine which summary was better and which was worse in terms of informativeness (is the meaning in the input text preserved in the summary?), fluency (is the summary written in well-formed and grammatical English?) and non-redundancy (does the summary avoid repeating information?). We randomly selected 50 documents from the Multi-News test set and compared all possible combinations of two out of four systems. We chose to compare PG-MMR, CopyTransformer, Hi-MAP and gold summaries. The order of summaries was randomized per example.

The results of our pairwise human-annotated comparison are shown in Table TABREF32 . Human-written summaries were easily marked as better than other systems, which, while expected, shows that there is much room for improvement in producing readable, informative summaries. We performed pairwise comparison of the models over the three metrics combined, using a one-way ANOVA with Tukey HSD tests and INLINEFORM0 value of 0.05. Overall, statistically significant differences were found between human summaries score and all other systems, CopyTransformer and the other two models, and our Hi-MAP model compared to PG-MMR. Our Hi-MAP model performs comparably to PG-MMR on informativeness and fluency but much better in terms of non-redundancy. We believe that the incorporation of learned parameters for similarity and redundancy reduces redundancy in our output summaries. In future work, we would like to incorporate MMR into Transformer models to benefit from their fluent summaries.

Conclusion

In this paper we introduce Multi-News, the first large-scale multi-document news summarization dataset. We hope that this dataset will promote work in multi-document summarization similar to the progress seen in the single-document case. Additionally, we introduce an end-to-end model which incorporates MMR into a pointer-generator network, which performs competitively compared to previous multi-document summarization models. We also benchmark methods on our dataset. In the future we plan to explore interactions among documents beyond concatenation and experiment with summarizing longer input documents.