# TOPIC-FOCUSED EXTRACTIVE SUMMARIZATION

AKSHAY KUMAR GUPTA

Supervised by DR. GREG DURRETT

A thesis submitted to
The University of Texas at Austin
for the distinction of Turing Scholar
in Partial Fulfillment
of the Requirements
for the Degree of
**BACHELOR OF SCIENCE**


Department of Computer Science
College of Natural Sciences
The University of Texas at Austin
May 2020

# Topic-Focused Extractive Summarization

**Akshay Kumar Gupta**
Department of Computer Science
The University of Texas at Austin
akshaykg@cs.utexas.edu

## Abstract

Extractive single-document summarization is the task of condensing a source document into a shorter form while retaining its information content and meaning, by identifying important sections of content in the document and generating them verbatim. However, such systems tend to produce summaries that are often overly generic, with little controllability over the output, and cannot cater to many individuals' unique information needs in a real-world setting. In this paper, we focus on the task of topic-focused extractive summarization, the task of extractively summarizing a document from some domain with a focus on some particular topic in that domain, with the goal of producing tailored summaries as per user-controlled specifications. We propose a new BERT-based neural model to learn this task, and build a system which can generate topic-focused summaries for unseen documents in some domain as per the user's requirements after being trained on a small number of document-summary pairs per domain. We run our system on the CNN/DM and CourtListener datasets, and evaluate it against three baselines (LEAD, KEYWORD and BERTSUM). Our experiments show that when evaluated by humans, we are able to match the coherence of the LEAD baseline while extracting content and consistently outperform ad hoc keyword-based methods; and that when evaluated automatically, we outperform keyword-based methods while providing greater controllability to users.

## 1 Introduction

Automatic single-document summarization is an important and long-studied (Nenkova and McKeown, 2011) problem in natural language processing; with numerous potential downstream applications in information retrieval and question answering tasks, among others. One approach to this task is extractive summarization, where important subsequences of text from a source document are lifted verbatim and concatenated to form a shorter document, whilst preserving the original document's information content. However, general-purpose extractive summarization systems lack controllability and thus are unable to condition their output on a user's unique information needs, which limits their utility in real-world settings. To address this shortcoming, we explore the idea of topic-focused extractive summarization – the task of generating an extractive summary for a document within some domain that is "focused" on a particular topic in that domain, where topics for a domain are user-specified. An example use case of such a system is provided in Figure 1.

Advances in computational resources and increased representational power provided by new, deeper architectures for models, along with the availability of large-scale datasets with hundreds of thousands of document-summary examples (Sandhaus, 2008; Hermann et al., 2015; Grusky et al., 2018), mean that neural and data-driven approaches to the single-document summarization problem have become more and more prominent over the past few years (Nallapati et al., 2016; Paulus et al., 2018; Li et al., 2017; See et al., 2017; Narayan et al., 2018; Gehrmann et al., 2018; Liu and Lapata, 2019). Furthermore, powerful pre-trained encoders like BERT (Devlin et al., 2019) have made significant impact in the area of transfer learning, affording us the ability to fine-tune them to various NLP tasks with relative ease and obtain state-of-the-art results on tasks like question answering and summarization. Furthermore, unlike most neural summarization models, the pre-trained BERT model does not require a great deal of data to be fine-tuned on various NLP tasks – this allows our system to outperform all three baselines on human evaluation after being trained on just a

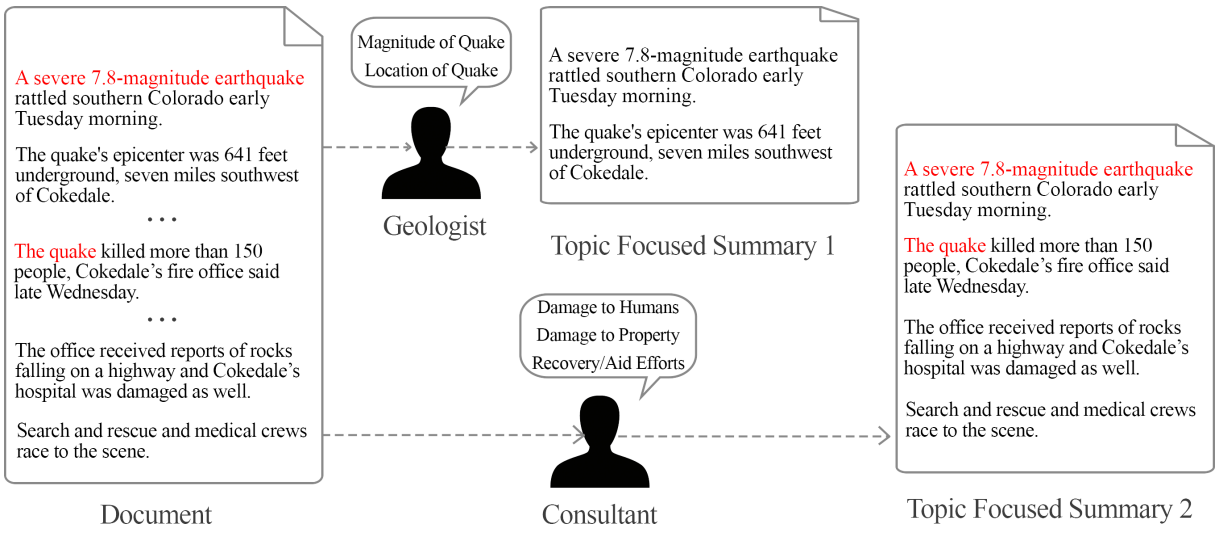


Figure 1: Example of two users who might care about different aspects of earthquake-related articles, and the divergent summaries our system could generate for them.

few hundred documents.

In this paper we build a topic-focused extractive summarization system that distills existing summaries of a small number of documents within a domain into several "topics" with minimal user supervision, and then allows the user to request tailored summaries for new documents in that domain based on their information needs and the topics for that domain. We accomplish this using a user-seeded clustering procedure on pre-existing summaries to determine topics, a beam search procedure to generate oracle extractive summaries from provided human-abstractive summaries for training, and a new topic-focused extractive summarization model built on top of the pre-trained BERT model.

The task worked on by (Stewart, 2009) is related to ours, although it relies on handcrafted features and nonparametric models such as SVMs and Random Forests for classification rather than neural methods, in addition to focusing more on the multi-document summarization problem. The work of (Baumel et al., 2018) is also similar to ours in that they decompose the QFS task into two parts – scoring sentences by relevance, and generating a summary using these scores over various queries – although they approach this as an abstractive rather than an extractive summarization problem.

We evaluate the performance of our model both automatically, and using human judgement, on two datasets – a small single-domain subset of the the widely popular single-document news summarization CNN/DM dataset (Hermann et al., 2015), as well as a single-domain subset of the CourtListener dataset (Lerman et al., 2017), a dataset containing legal opinions in federal courts of the USA. The domain we chose from CNN/DM was "earthquakes" – with topics like "magnitude of quake", "epicenter of quake", "damage to property", "damage to human life" and "recovery/aid efforts" – all of which are topics that a user might want summarized for any given article about an earthquake. The domain we use from Courtlistener is "denial of post-conviction relief" – with topics like "facts and background", "appellant's claims", "court's findings" and "final judgement" – all components of a court opinion that a user might be particularly interested in learning about when reading an opinion about such a case.

Across these datasets, we experimentally show that our system is consistently evaluated by humans as at par with or better than several baselines including a ad hoc keyword-based method, and when evaluated automatically, our system outperforms all baselines on the Courtlistener dataset and a keyword-based method on the "earthquakes" domain, while being able to generate tailored summaries that cater to specific user needs.

Our main contributions through this work are as follows: we propose a new system for topic-focused extractive summarization that allows for greater controllability in output to meet users' information needs; evaluation and analysis showing that our system matches or outperforms several baselines including an ad hoc keyword-based

method and the BERTSUM system on several qualitative metrics; we present a novel algorithm for training our system, leveraging a pre-trained Transformer architecture and fine-tuning it for our task.

## 2 System Overview

A user may provide our system with an unseen $n$-sentence document $D = \{s_1, \cdots, s_n\}$ from some domain, specifying some $T \subseteq \mathcal{T}$ (where $\mathcal{T}$ is a set of user-specified topics for the domain) that they would like the summary to be focused on. For each $t \in T$, the corresponding trained model for $t$ is fed $D$ and returns some $1 \leq i \leq n$ – the index of the extractive summary sentence most relevant to $t$. This process leaves us with a base summary $\hat{S} \subseteq D$ where $|\hat{S}| = |T|$. Finally, we run a procedure that extends $\hat{S}$ with additional pieces of missing context based on coreference resolution using AllenNLP (Gardner et al., 2018), inserting additional sentences from $D$ where necessary to produce a final summary. This is illustrated in Figure 2.

The models we train for each topic are trained on a set $\mathcal{D}$ of documents from some domain (e.g. earthquakes, court cases about post-conviction relief), along with their respective gold-standard summaries $\mathcal{S}$. We ask the user to specify certain "topics" $\mathcal{T}$ for the domain's summaries, on the sentence level (e.g., earthquake magnitude, recovery/aid efforts, result of appeal). We then cluster all summary sentences (after flattening $\mathcal{S}$ i.e. putting all summary sentences in a single array) and assign to each sentence a single "topic" from among those previously specified. In the interest of performance, we allow the user to "seed" this clustering by pre-assigning some sentences to each cluster.

Next, we construct a set of oracle extractive summaries $\mathcal{O}$ for $\mathcal{D}$ using a beam search procedure and optimizing directly for ROUGE with respect to the gold-standard summary for each document – these will be used for training our models. The topic assignments from the clustering procedure are preserved, so each oracle extractive summary sentence is associated with the same topic as its reference gold-standary summary sentence. For each "topic" specified within the domain, we then train a model to identify the most relevant sentence in a document with respect to that topic.

## 3 Topic Clustering

When we consider summaries for documents at large within a certain domain, there often emerge patterns in the kind of information that is summarized, or salient topics for that domain that end-users have a high likelihood of being interested in. Examples of these for the "earthquake news" domain would be "magnitude of the quake", "damage to property as a result of the quake", "recovery/aid efforts following the quake". We argue that this notion of "topics" can be leveraged to create a system that can produce tailored summaries for an end-user, with a relatively low number of training examples per domain (just a few hundred).

Our goal, for a given domain with some topics (a relatively small number, say 5-10), is to train a model per topic to identify the most relevant sentence for that topic in a document. In order to train our models, we need to assign a topic to each summary sentence in our dataset, so that we can have some notion of a "gold-standard" for sentences of a given topic. To do so, we flatten the set of gold-standard summaries $\mathcal{S}$ (which are already sentence-tokenized), and apply a clustering algorithm on the result (call it $\mathcal{S}'$). This allows us to represent a summary $s = \{s_1, \cdots, s_m\}$ as $s_{\mathcal{T}} = \{t_1, \cdots, t_m\}$ where $t_i \in \mathcal{T}$ is the topic assigned to sentence $s_i$ by the clustering algorithm. Each document with at least one summary sentence of topic $t$ will be included in the training data for the model corresponding to topic $t$.

We use a $k$-means clustering procedure, using tf-idf (Salton and McGill, 1986) representations for each sentence with a maximum of $10,000$ features across unigrams, bigrams, and trigrams. We set the minimum document frequency threshold to 5 for the tf-idf feature-generation procedure. Running this completely unsupervised yielded inconsistent clustering results, so in the interest of cluster assignment quality as well as increased controllability on the user end, we allow minimal supervision in the form of user-provided seeds for the $k$-means procedure. Here, the user can manually specify 10-15 seed sentences per cluster, which are then used to initialize cluster centroids and seed the clustering; this approach yields a significantly cleaner separation into clusters corresponding to each topic. Examples of summary sentences with their assigned topic are shown in Table 1.

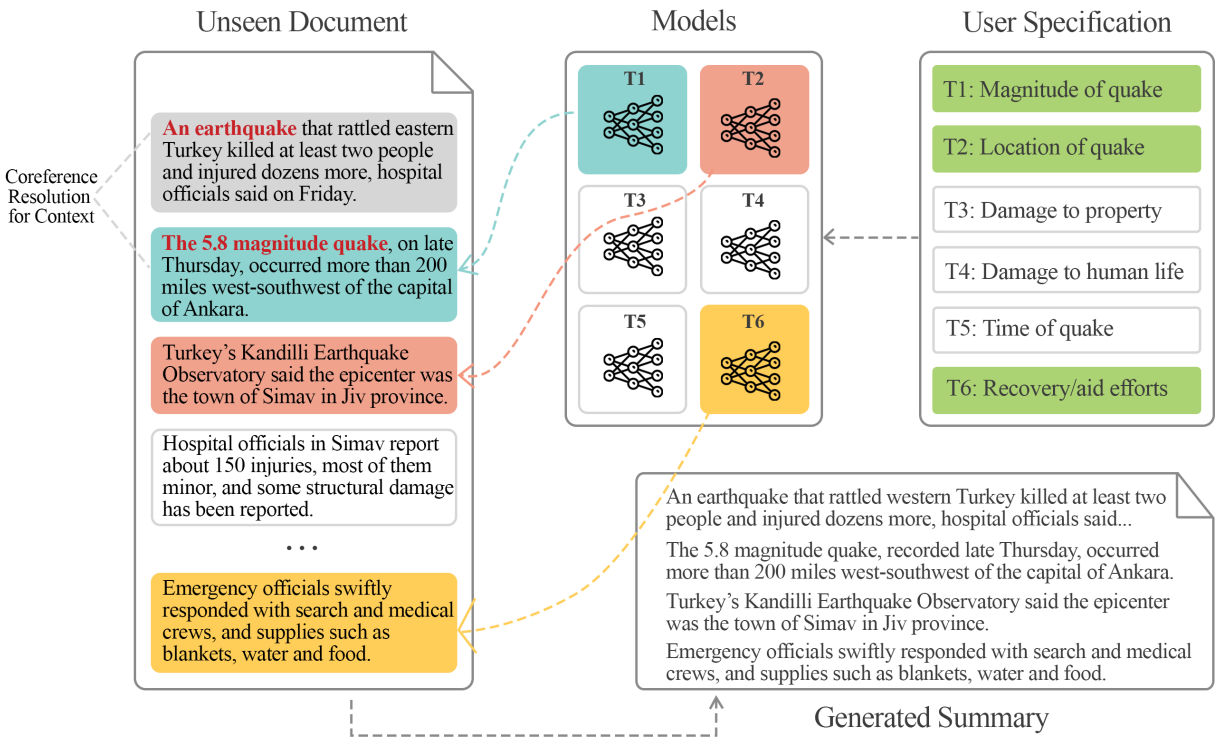


Figure 2: Diagram of the proposed system. A user can specify a subset of topics to extract. Each topic has a corresponding model which can choose a maximally relevant sentence from the unseen document; we then expand these sentences to include relevant context via coreference resolution to produce the final summary.

## 4 Models

**BERT-based Model** Our main model is a neural network model that encodes each sentence of a source document, scores each sentence for relevance to a certain topic, and returns the most relevant sentence from the document. We fine-tune a BERT model (Devlin et al., 2019) with a single linear layer on top of the `[CLS]` representation of the entire sentence in the ouput layer, for sentence classification.

We use the BERT-base model, which contains an encoder with 12 Transformer blocks, 12 self-attention heads and a hidden state size of 768. The base model takes as input a sequence of no more than 512 tokens, and returns its representation. The special token `[CLS]` is inserted at the start of each input sequence, and contains the special classification embedding in this task, and the special token `[SEP]` is appended to the end of each sequence as a separator.

For text classification tasks such as this one, BERT uses the final hidden state $\mathbf{h}$ of the `[CLS]` token to store the representation of the entire sequence, and a softmax layer is added on top of the base model to predict the probability of a particular class label $c$ (or in this case, the relevance to our single class/topic):

$$p(c|\mathbf{h}) = \text{softmax}(W\mathbf{h})$$

where $W$ is the weight matrix for our task.

We encode each sentence in each document using the BERT tokenizer and encoder – this procedure also inserts the special tokens referenced earlier into each sequence, and truncates all sequences to a maximum of 512 characters, including special tokens. We use the AdamW Optimizer (Loshchilov and Hutter, 2019) with a learning rate of $2 \times 10^{-5}$, training for 4 epochs with a batch size of 1 due to issues with BERT running out of memory when trained on multiple long documents in a single batch.

**Linear Model** In addition to our main model, we also implement and evaluate a simple linear model, consisting of a single linear layer with log softmax applied on top. Sentences are encoded using a BoW representation, with unigram and bigram features (limited to $10,000$ features) as well as a single positional feature (where in the document does the sentence occur) and a small set of binary indicator features (7 bits) used to encode the length of the sentence. We use the AdamW Optimizer for this model as well, with a learning

| Topic | Seed Sentence | Clustered Sentences |
|---|---|---|
| Magnitude of Quake | It was centered some 50 miles (80 km) south southwest of Avsallar, Turkey. | - The quake was centered 5 miles northwest of Youngstown and 1.4 miles below the surface.<br>- Epicenter was about 20 miles southwest of port city of Patras.<br>- Quake was centered 105 miles of the coast of Honshu, Japan's main island. |
| Damage to Human Life | NEW: 186 people are dead and 8,200 hospitalized, Chinese state media Xinhua reports | - Six people dead and more than a dozen hurt in Balochistan province, Pakistani official says.<br>- At least seven dead, 50 injured in northern Italy quake.<br>- More than 4,100 people were injured in the quake last Sunday |

Table 1: Examples of seed sentences from two topics in the "earthquakes" domain, along with some sentences assigned to them by the clustering algorithm.

| Domain | R-1 | R-2 | R-L |
|---|---|---|---|
| Earthquakes | 41.5 | 20.7 | 33.7 |
| Post-Conviction Relief | 50.5 | 31.6 | 46.0 |

Table 2: Oracle summary quality evaluation on both datasets. ROUGE-1, -2 and -L $F_1$ is reported.

rate of $0.01$ and a batch size of 16, training for at most 100 epochs and using early stopping with 7 patience. This model is evaluated using automatic metrics – we do not include summaries produced by this model in our human evaluations.

## 5 Training

### 5.1 Oracle Construction

In order to train our extractive models, we need ground truth extractive summaries for each document – these take the form of a subset of sentences in each document. Since the majority of summarization datasets (including the ones we use) contain human written abstractive summaries as the ground truth, we use an unsupervised approach utilizing a beam search procedure akin to Maximal Marginal Relevance (MMR) (Carbonell and Goldstein, 1998) to obtain extractive labels from the provided abstractive summaries. The idea behind our approach here is to maximize the ROUGE score of our selected sentences with respect to the reference summary.

If the reference summary consists of $n$ sentences $S = \{s_1, \cdots, s_n\}$, then we construct an oracle $O = \{o_1, \cdots, o_n\}$ with the same number of sentences, all from the document. Our heuristic cost for selecting sentence $d$ from the document to use as $o_i$ is the ROUGE score of the already-constructed $O_{i-1} = \{o_1, \cdots, o_{i-1}\}$ with $d$ appended, with respect to $S_i = \{s_1, \cdots, s_i\}$. During the state-pruning process, we use the heuristic score of the combination of sentences as calculated above and sort in descending order. We use a beam width of $\beta = 15$, which means that at any point, we have at most 15 candidate oracles being considered, and the procedure returns 15 oracle summaries, of which we pick the first (highest scoring) one.

This procedure extracts a bag of sentences from the document, but we don't know the correspondence with summary sentences, which is needed by our model. To fix this, we run a small optimization procedure which tries all permutations (where computationally feasible) of oracle summary sentences, and chooses the one where the sum of pairwise ROUGE-1 $F_1$ with respect to the already-ordered reference summary sentences over the entire summary is maximized. Our measurements of Oracle summary ROUGE with respect to gold standard summaries are provided in Table 2 and show that the oracle summaries we produce are of high quality.

### 5.2 Learning Objective

Let $T \in \mathcal{T}$ be the topic for which we are building a model. Each training example is of the form $(D, o)$ where $D = \{s_1, \cdots, s_n\}$ is a document, and $1 \leq o \leq n$ is the index of an oracle sentence of topic $T$ for $D$, corresponding to a summary sentence of topic $T$ for $D$. Our learning objective is to find the sentence that has the highest relevance to topic $T$ from $D$. We use cross-entropy as our

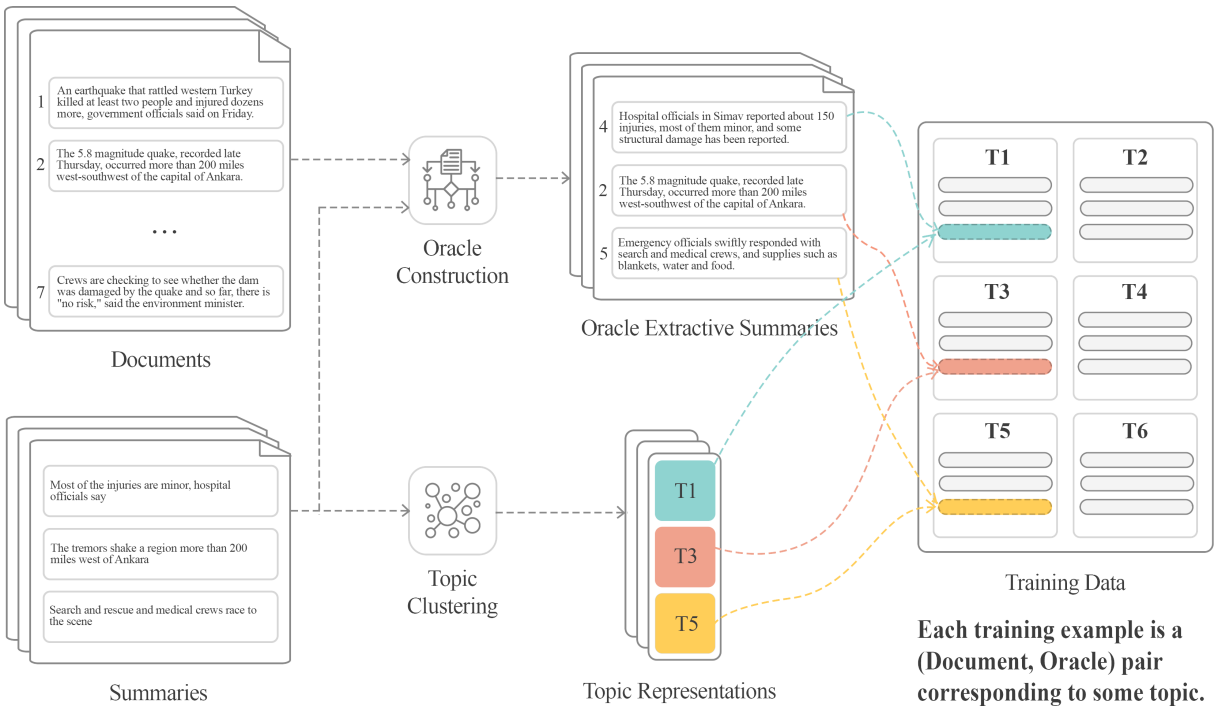


Figure 3: Pipeline from datasets to per-model training data with example document. We cluster sentences from each summary to assign each to one of several topics. We then collect training data for each topic in the form of (document, sentence index) pairs, and train each model to select the specified sentences.

loss function -

$$\mathcal{L}(\theta) = -\log p(o|D,\theta) = -\sum_{i=1}^{n} \log p(y_i|s_i, D, \theta)$$

Where $y_i = 1$ if $i = o$ and 0 otherwise. Our objective learns to discriminate among sentences with respect to how relevant they are to a given topic by optimizing the log likelihood for each sentence, maximizing the likelihood of the correct sentence being selected.

On the Courtlistener Dataset, large document sizes made it impractical to train BERT so we "reduce" each example $E = (D, o)$ to $\hat{E} = (\hat{D}, \hat{o})$ where $\hat{D}$ is a subset of $D$ of size $k = 10$ that is guaranteed to contain $s_o$ and $\hat{o}$ is the index of $s_o$ in $D$. Full documents are still used for validation and testing. An illustration of the pipeline from datasets to per-model training data is shown in figure 3.

## 6 Experimental Setup

### 6.1 Datasets

**Courtlistener** (Lerman et al., 2017) is a dataset containing legal opinions in federal courts of the United States of America. A legal opinion is a document written by a judge or a judicial panel that summarizes their decision and all relevant facts about a court case. We obtained 6,409 examples ranging over the years 2017-2019 by scraping the Courtlistener website. For the Topic-Based Extractive Summarization Task, we performed tf-idf based k-means clustering to separate the dataset into smaller domains such as opinions involving appeals for post-conviction relief (197 examples). These domains were then used independently for training and testing our system. An example document-summary pair is shown in Table 3.

**CNN/DailyMail** (Hermann et al., 2015; Nallapati et al., 2016) is an extractive summarization dataset containing news articles (781 tokens on average) paired with 3-4 sentence summaries (3.75 sentences or 56 tokens on average) that summarize the contents of the article. The dataset contains 287k examples, which we distilled down to 578 examples pertaining to earthquakes for the Topic-Based Extractive Summarization Task – this was done using a simple keyword search for "earthquake" and "magnitude". An example document-summary pair is shown in Figure 3.

| Reference Summary | Document (abridged) |
|---|---|
| Most of the injuries are minor, hospital officials say. The tremors shake a region more than 200 miles west of Ankara. Search and rescue and medical crews race to the scene. | – An earthquake that rattled western Turkey killed at least two people and injured dozens more, hospital and government officials said on Friday. The 5.8 magnitude quake, recorded late Thursday, occurred more than 200 miles west-southwest of the capital of Ankara, the U.S. Geological Survey said. Turkey's Kandilli Earthquake Observatory said the epicenter was the town of Simav in Kutahya province, where aftershocks rippled across the region overnight. Hospital officials in Simav reported about 150 injuries, most of them minor, and some structural damage has been reported. Emergency officials swiftly responded with search and medical crews, and supplies such as blankets, water and food. The Kutahya region recently had a scare of leaked cyanide-contaminated water from a wastewater dam after an embankment collapse. Crews are checking to see whether the dam was damaged by the quake and so far, there is "no risk," said Veysel Eroglu, the environment minister. |
| In 2013, the Petitioner, Julie Bauer, pleaded guilty to attempted murder with an agreed sentence of twenty-nine years of incarceration. Subsequently, the Petitioner filed a petition for post-conviction relief, which the post-conviction court denied after a hearing. On appeal, the Petitioner contends that the post-conviction court erred when it denied her petition because she received the ineffective assistance of counsel. After review, we affirm the post-conviction court's judgment. | (. . . ) Based on this incident, a Maury County grand jury indicted the Petitioner for first degree premeditated murder and conspiracy to commit first degree premeditated murder as to her mother, and attempted first degree premeditated murder and conspiracy to commit first degree premeditated murder as to her father. (. . . ) Petitioner entered a best interest plea to attempted first degree premeditated murder with an agreed-upon sentence of twenty-nine years; the remaining counts in the indictment were dismissed. (. . . ) The Petitioner filed a petition for post-conviction relief, pro se. The post- conviction court appointed an attorney, and the attorney filed an amended petition, alleging that the Petitioner had received the ineffective assistance of counsel when counsel failed to assist the Petitioner in reserving a certified question of law (. . . ) After a thorough review of the record and the applicable law, we conclude the post-conviction court properly denied the Petitioner's petition for post-conviction relief. In accordance with the foregoing reasoning and authorities, we affirm the judgment of the post-conviction court. |

Table 3: Example summaries and documents from each dataset we used. The document for Courtlistener is abridged (...) in this table for the readers' convenience.

### 6.2 Evaluation Metrics

We conduct evaluation using both automatic metrics as well as human judgement. For automatic evaluation, as is standard, we use $F_1$ scores from unigram and bigram overlap (ROUGE-1 and ROUGE-2) as well as LCS (ROUGE-L) with respect to the gold standard summaries. Following standard practice, we used the $F_1$ scores from the ROUGE-1, ROUGE-2 and ROUGE-L metrics to automatically evaluate the quality of summaries produced by our system. These metrics count the number of overlapping units such as n-gram, word sequences, and word pairs between model-generated summaries and human-written gold-standard summaries. Generally, ROUGE-1 and ROUGE-2 tend to give us insight into a summary's informativeness, and ROUGE-L is more indicative of fluency. We note that these metrics do not allow us to draw conclusions with regards to our system's efficacy at the topic-focused aspect of our task.

Human evaluation was conducted on the Amazon Mechanical Turk platform. We randomly selected 50 news articles from the earthquakes domain, and each Turker was provided with an article along with summaries generated by four systems: the LEAD baseline, the KEYWORD baseline, the BERTSUM baseline and OURS. They were then asked to score each summary from 1-3 along four axes: (1) **Topic 1 Relevance**: if the summary addresses the topic and provides useful information with respect to it; (2) **Topic 2 Relevance**: same as (1); (3) **Coherence**: how well the summary "flows" and the extent to which it is understandable when read; (4) **Fluency**: if the summary is fluent and doesn't contain grammatical errors.

Each set of summaries receives three scores for each axis, the arithmetic mean of which is used during evaluation. All participants were required to have earned the Mechanical Turk Masters Qualification, and the order of summaries to rank was randomized per article.

### 6.3 Baselines

LEAD This baseline simply selects the first $k$ sentences from the document, where $k$ is the number of topics that the user is interested in summarizing.

KEYWORD MATCHING This baseline uses user-provided keywords for each information topic, calculating a score $S$ for each sentence $s = \{w_1, \cdots, w_n\}$ ($w_i$ are words in the sentence) for topic $t$ with keywords $K_t = \{k_{t1}, \cdots k_{tm}\}$ as $S = |s \cap K_t|$, and returns the highest-scoring sentence for each topic the user requests. In the

event of a tie, the sentence that occurs earliest in the document is selected for that topic. If the highest score for a topic is 0, a random sentence is selected.

BERTSUM This is the BERTSUM+Classifier model from (Liu and Lapata, 2019) – a general-purpose summarization model with no topic-focused functionality. It was trained with a learning rate of $2 \times 10^{-3}$, a dropout of $0.1$ and $50,000$ steps. We used the same train/val/test split on both datasets as was used for our main model.

## 7 Results

### 7.1 Clusters Learned

For the "earthquakes" domain, based on manual observation we split summary sentences into 6 main topics – (1) magnitude of the earthquake, (2) location of the earthquake, (3) damage to property and infrastructure, (4) damage to human life, (5) time that the earthquake occurred, (6) recovery/aid efforts after the earthquake, along with (7) a "garbage" topic used for summary sentences that may be unrelated to any of the main topics. Seeding was done with an average of $14$ sentences per topic, from a total of $1,923$ sentences across all $578$ summaries.

For the "post conviction relief" domain, we split summary sentences into 4 main topics – (1) facts and background, (2) appellant's claims, (3) court's findings, (4) final judgement. We observed that these topics tend to appear in the same order within the reference summaries, so we are able to separate sentences out simply based on their position in the document, without any need for a clustering procedure. The total number of sentences was $761$ across $197$ summaries.

### 7.2 Automatic Evaluation

Results of automatic evaluation on both datasets are presented in Table 4. LEAD is configured with $k = 3$, and the topics of interest used for KEYWORD, LINEAR and MAIN are "damage to property and infrastructure" and "recovery/aid efforts" for the "earthquakes" domain and "facts and background" and "final judgement" for the "post-conviction relief" domain. Our BERT-based model performs far better than all three baselines on the "post-conviction relief" domain in the Courtlistener dataset, followed closely by our linear model. On the "earthquakes" domain in the CNN/DM dataset, our main model performs better than KEYWORD baseline as well as our linear model, and almost as well as BERTSUM and Lead, but while providing far greater controllability to the user. Some statistics for the output of our system are also provided in Table 5.

| Model | Earthquakes | | | Post-Conviction Relief | | |
|---|---|---|---|---|---|---|
| | R-1 | R-2 | R-L | R-1 | R-2 | R-L |
| LEAD | 23.4 | 8.0 | 18.9 | 14.9 | 4.1 | 11.8 |
| KEYWORD | 19.8 | 5.6 | 15.6 | 23.2 | 9.1 | 19.1 |
| BERTSUM | **24.7** | **8.6** | **19.3** | 17.2 | 4.5 | 13.9 |
| LINEAR | 20.0 | 6.1 | 15.5 | 28.3 | 12.6 | 23.0 |
| MAIN | 22.1 | 7.6 | 16.4 | **28.6** | **12.8** | **23.5** |
| LINEAR* | 21.6 | 6.9 | 16.6 | - | - | - |
| MAIN* | 23.2 | 8.2 | 17.4 | - | - | - |

Table 4: Automatic Evaluation results on the "earthquakes" domain within the CNN/DM dataset, and the "denial of post-conviction relief" domain within the Courtlistener dataset. ROUGE-1, ROUGE-2 and ROUGE-L $F_1$ is reported.

| Model | # Tokens | # Sentences |
|---|---|---|
| LEAD | 77.52 | 3.00 |
| KEYWORD | 74.37 | 2.79 |
| BERTSUM | 72.42 | 2.93 |
| MAIN | 99.24 | 3.96 |

Table 5: Statistics for system outputs on the "earthquakes" domain within the CNN dataset, with topics "location of the earthquake", "magnitude of the earthquake" and "damage to human life". Our model produces longer summaries due to the context-adding procedure we run at the end.

We also run an alternative evaluation procedure for both of our models on the "earthquakes" domain (LINEAR* and MAIN* in Table 4), wherein for each example, we obtain the topic representation of the reference summary, discard garbage sentences, run the model for each distinct topic in the representation, selecting the top-$k$ predictions where $k$ is the number of sentences of that topic in the summary. We then run an optimization procedure similar to the one described during Oracle Construction to reorder the selected sentences and produce a final summary for ROUGE evaluation. This allows us to evaluate our model's ability to recreate the reference summary given its topic representation, whereas our original evaluation method fixes the topics beforehand, regardless of whether or not a given example has any sentences of that topic. Under this evaluation scheme, our models score higher than they do based on the

| Model | Topic 1 | Topic 2 | Coh. | Gram. | Overall |
|---|---|---|---|---|---|
| LEAD | 2.54 | 2.21 | 2.62 | **2.89** | 2.56 |
| KEYWORD | 2.65 | **2.51** | 2.69 | **2.89** | 2.68 |
| BERTSUM | 2.43 | 2.19 | 2.49 | 2.63 | 2.43 |
| MAIN | $^{*}$**2.69** | $^{*}$**2.51** | $^{\dagger *}$**2.77** | $^{*}$2.86 | $^{*}$**2.71** |

Table 6: Human evaluation results on the "earthquakes" domain within the CNN dataset, with topics "damage to property and infrastructure" and "recovery/aid efforts". We asked Turkers to score the models' output along four axes from a scale of 1 to 3. We compare results using a paired bootstrap test; † indicates better than KEYWORD and $*$ indicates better than BERTSUM with $p < 0.05$.

| Model | Topic 1 | Topic 2 | Coh. | Gram. | Overall |
|---|---|---|---|---|---|
| LEAD | 1.93 | 1.61 | **2.56** | **2.83** | 2.23 |
| KEYWORD | 1.99 | **1.76** | 2.35 | 2.79 | 2.22 |
| BERTSUM | 1.97 | 1.70 | 2.39 | 2.61 | 2.17 |
| MAIN | $^{\dagger *}$**2.11** | 1.71 | $^{\dagger *}$2.53 | $^{*}$2.78 | $^{\dagger *}$**2.28** |

Table 7: Human evaluation results on the "earthquakes" domain within the CNN dataset, with topics "location of the earthquake" and "damage to human life". We asked Turkers to score the models' output along four axes from a scale of 1 to 3. We compare results using a paired bootstrap test; † indicates better than KEYWORD and $*$ indicates better than BERTSUM with $p < 0.05$.

original scheme, and we claim that this scheme more captures our models' performance more consistently.

### 7.3 Human Evaluation

Results for human evaluation on the "earthquakes" domain within the CNN/DM dataset are presented in Table 6 and Table 7. Our model was evaluated as the best model for relevance to topic 1 as well as overall in both human judgement studies we conducted, and consistently beat BERTSUM across the board on both studies as well, outperforming both KEYWORD and BERTSUM on coherence with high statistical significance, indicating that our system is capable of producing summaries that are (1) relevant to the user's information needs and (2) easily digestible and fluent. Since all models being compared are purely extractive in nature, and operate at sentence granularity, we claim that grammar scores are influenced mostly by the quality of the data itself and not as much by the choice of model.

## 8 Related Work

**Extractive Summarization** There is a large body of existing work focusing on extractive summarization. Preliminary work on this task usually relied on human-engineered features (Filatova and Hatzivassiloglou, 2004) combined with binary classifiers (Kupiec et al., 1995), Hidden Markov models (Conroy and O'leary, 2001), graph based methods (Mihalcea and Strapparava, 2005) and integer linear programming (Woodsend and Lapata, 2011).

Neural networks have gained widespread popularity for extractive summarization tasks in recent years due to their efficacy. Previous work on this task has included a variety of data-driven approaches, including a Recurrent Neural Network based encoder for binary classification (Nallapati et al., 2017), a reinforcement-learning based system that ranks sentences (Narayan et al., 2018), a seq-to-seq decoder for index prediction (Zhou et al., 2018). A BERT-based architecture leveraging document-level encoding (Liu and Lapata, 2019) represents the state of the art, which we employ and compare to here.

**Focused Summarization** Due to an increasing emphasis on building natural language systems that are versatile and usable in real-world settings, there is a rich body of existing work on the task of producing summaries that are "focused" or "tailored" to some end-user requirement. Existing research on this task largely relies on extractive approaches, some of which includes cascaded and multitask CNNs for aspect oriented summarization of reviews (Wu et al., 2016), self-attention mechanisms for generating query-based summaries (Xie et al., 2020) and divide-and-conquer leveraging a seq-to-seq LSTM+RUM architecture (Gidiotis and Tsoumakas, 2020). Work by (Stewart, 2009) is similar to ours, albeit it does not leverage neural methods and instead relies on SVMs and Random Forests for classification, in addition to prioritizing the multi-document summarization problem. The work of (Baumel et al., 2018) is also similar to ours in that they split the query focused summarization task into two parts – first, determining the relevance per sentence to the query, and then using summarization methods to put together the actual summary – although they focus more on an abstractive summarization approach.

## 9 Conclusion

In this paper we propose and construct a topic-focused extractive summarization system that allows users to request tailored summaries for unseen documents in some domain based on their information needs and the topics for that domain, after distilling existing summaries of a small number of documents within the domain into several "topics" with minimal user supervision and training on these clusters per topic. We show experimentally that our new model outperforms several baselines on human evaluation of the topic-focused extractive summarization task, and outperforms these baselines on the Courtlistener dataset when evaluated automatically as well.

## Acknowledgments

Thanks to Greg Durrett and Jiacheng Xu for advising me on this project, supervising my research and providing me with plentiful support and guidance. I would also like to thank my parents and my peers for the immense support and belief they showed throughout the research process. Special thanks to Arushee Agrawal for help with figures! This work was supported by funding from Walmart Labs and an equipment grant from NVIDIA. Opinions expressed in this paper do not necessarily reflect the views of these sponsors.

## 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. *CoRR*, abs/1801.07704.

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, page 335–336, New York, NY, USA. Association for Computing Machinery.

John M. Conroy and Dianne P. O'leary. 2001. Text summarization via hidden markov models. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '01, page 406–407, New York, NY, USA. Association for Computing Machinery.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Elena Filatova and Vasileios Hatzivassiloglou. 2004. Event-based extractive summarization. In *Text Summarization Branches Out*, pages 104–111, Barcelona, Spain. Association for Computational Linguistics.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F. Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. AllenNLP: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6, Melbourne, Australia. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Alexios Gidiotis and Grigorios Tsoumakas. 2020. A divide-and-conquer approach to the summarization of academic articles.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 708–719, New Orleans, Louisiana. Association for Computational Linguistics.

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 - Volume 1*, NIPS'15, page 1693–1701, Cambridge, MA, USA. MIT Press.

Julian Kupiec, Jan Pedersen, and Francine Chen. 1995. A trainable document summarizer. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '95, page 68–73, New York, NY, USA. Association for Computing Machinery.

Kristina Lerman, Nathan O. Hodas, and Hao Wu. 2017. Bounded rationality in scholarly knowledge discovery. *CoRR*, abs/1710.00269.

Piji Li, Zihao Wang, Wai Lam, Zhaochun Ren, and Lidong Bing. 2017. Salience estimation via variational auto-encoders for multi-document summarization.

Yang Liu and Mirella Lapata. 2019. 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 (EMNLP-IJCNLP)*, pages 3730–3740, Hong Kong, China. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Rada Mihalcea and Carlo Strapparava. 2005. Making computers laugh: Investigations in automatic humor recognition. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 531–538, Vancouver, British Columbia, Canada. Association for Computational Linguistics.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17, page 3075–3081. AAAI Press.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar GuÌ‡lçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Ranking sentences for extractive summarization with reinforcement learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1747–1759, New Orleans, Louisiana. Association for Computational Linguistics.

Ani Nenkova and Kathleen McKeown. 2011. *Automatic Summarization*, volume 5.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *International Conference on Learning Representations*.

Gerard Salton and Michael J McGill. 1986. Introduction to modern information retrieval.

Evan Sandhaus. 2008. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6(12):e26752.

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 (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

J.G. Stewart. 2009. *Genre Oriented Summarization*. Report (Carnegie Mellon University. Language Technologies Institute). Carnegie Mellon University, Language Technologies Institute, School of Computer Science.

Kristian Woodsend and Mirella Lapata. 2011. Learning to simplify sentences with quasi-synchronous grammar and integer programming. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 409–420, Edinburgh, Scotland, UK. Association for Computational Linguistics.

H. Wu, Y. Gu, S. Sun, and X. Gu. 2016. Aspect-based opinion summarization with convolutional neural networks. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 3157–3163.

Yujia Xie, Tianyi Zhou, Yi Mao, and Weizhu Chen. 2020. Conditional self-attention for query-based summarization. *CoRR*, abs/2002.07338.

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.