NEWSROOM: A Dataset of 1.3 Million Summaries with Diverse Extractive Strategies

Max Grusky\textsuperscript{1,2}, Mor Naaman\textsuperscript{2}, Yoav Artzi\textsuperscript{1,2}
\textsuperscript{1}Department of Computer Science, \textsuperscript{2}Cornell Tech
Cornell University, New York, NY 10044
\{grusky@cs, mor@jacobs, yoav@cs\}.cornell.edu

Abstract

We present NEWSROOM, a summarization dataset of 1.3 million articles and summaries written by authors and editors in newsrooms of 38 major news publications. Extracted from search and social media metadata between 1998 and 2017, these high-quality summaries demonstrate high diversity of summarization styles. In particular, the summaries combine abstractive and extractive strategies, borrowing words and phrases from articles at varying rates. We analyze the extraction strategies used in NEWSROOM summaries against other datasets to quantify the diversity and difficulty of our new data, and train existing methods on the data to evaluate its utility and challenges. The dataset is available online at summari.es.

1 Introduction

The development of learning methods for automatic summarization is constrained by the limited high-quality data available for training and evaluation. Large datasets have driven rapid improvement in other natural language generation tasks, such as machine translation, where data size and diversity have proven critical for modeling the alignment between source and target texts (Tiedemann, 2012). Similar challenges exist in summarization, with the additional complications introduced by the length of source texts and the diversity of summarization strategies used by writers. Access to large-scale high-quality data is an essential prerequisite for making substantial progress in summarization. In this paper, we present NEWSROOM, a dataset with 1.3 million news articles and human-written summaries.

NEWSROOM’s summaries were written by authors and editors in the newsrooms of news, sports, entertainment, financial, and other publications. The summaries were published with articles as HTML metadata for social media services and search engines page descriptions. NEWSROOM summaries are written by humans, for common readers, and with the explicit purpose of summarization. As a result, NEWSROOM is a nearly two decade-long snapshot representing how single-document summarization is used in practice across a variety of sources, writers, and topics.

Identifying large, high-quality resources for summarization has called for creative solutions in the past. This includes using news headlines as summaries of article prefixes (Napoles et al., 2012; Rush et al., 2015), concatenating bullet points as summaries (Hermann et al., 2015; See et al., 2017), or using librarian archival summaries (Sandhaus, 2008). While these solutions provide large scale data, it comes at the cost of how well they reflect the summarization problem or their focus on very specific styles of summarizations, as we discuss in Section 4. NEWSROOM is distinguished from these resources in its combination of size and diversity. The summaries were written with the explicit goal of concisely summarizing news articles over almost two decades.

Abstractive Summary: South African photographer Anton Hammerl, missing in Libya since April 4th, was killed in Libya more than a month ago.

Mixed Summary: A major climate protest in New York on Sunday could mark a seminal shift in the politics of global warming, just ahead of the U.N. Climate Summit.

Extractive Summary: A person familiar with the search tells The Associated Press that Texas has offered its head coaching job to Louvilles Charlie Strong and he is expected to accept.

Figure 1: NEWSROOM summaries showing different extraction strategies, from time.com, mashable.com, and foxsports.com. Multi-word phrases shared between article and summary are underlined. Novel words used only in the summary are italicized.
Rather than rely on a single source, the dataset includes summaries from 38 major publishers. This diversity of sources and time span translate into a diversity of summarization styles.

We explore NEWSROOM to better understand the dataset and how summarization is used in practice by newsrooms. Our analysis focuses on a key dimension, extractiveness and abstractive: extractive summaries frequently borrow words and phrases from their source text, while abstractive summaries describe the contents of articles primarily using new language. We develop measures designed to quantify extractiveness and use these measures to subdivide the data into extractive, mixed, and abstractive subsets, as shown in Figure 1, displaying the broad set of summarization techniques practiced by different publishers.

Finally, we analyze the performance of three summarization models as baselines for NEWSROOM to better understand the challenges the dataset poses. In addition to automated ROUGE evaluation (Lin, 2004a,b), we design and execute a benchmark human evaluation protocol to quantify the output summaries relevance and quality. Our experiments demonstrate that NEWSROOM presents an open challenge for summarization systems, while providing a large resource to enable data-intensive learning methods. The dataset and evaluation protocol are available online at summarises.

2 Existing Datasets

There are a several frequently used summarization datasets. Listed in Figure 2 are examples from four datasets. The examples are chosen to be representative: they have scores within 5% of their dataset average across our analysis measures (Section 4). To illustrate the extractive and abstractive nature of summaries, we underline multi-word phrases shared between the article and summary, and italicize words used only in the summary.

2.1 Document Understanding Conference

Datasets produced for the Document Understanding Conference (DUC)1 are small, high-quality datasets developed to evaluate summarization systems (Harman and Over, 2004; Dang, 2006).

DUC data consist of newswire articles paired with human summaries written specifically for DUC. One distinctive feature of the DUC datasets

---

1http://duc.nist.gov/
unsupervised methods (Dorr et al., 2003; Mihalcea and Tarau, 2004; Barrios et al., 2016).

2.2 Gigaword

The Gigaword Corpus (Napoles et al., 2012) contains nearly 10 million documents from seven newswire sources, including the Associated Press, New York Times Newswire Service, and Washington Post Newswire Service. Compared to other existing datasets used for summarization, the Gigaword corpus is the largest and most diverse in its sources. While Gigaword does not contain summaries, prior work uses Gigaword headlines as simulated summaries (Rush et al., 2015; Chopra et al., 2016). These systems are trained on Gigaword to recreate headlines given the first sentence of an article. When used this way, Gigaword’s simulated summaries are shorter than most natural summary text. Gigaword, along with similar text-headline datasets (Filippova and Altun, 2013), are also used for the related sentence compression task (Dorr et al., 2003; Filippova et al., 2015).

2.3 New York Times Corpus

The New York Times Annotated Corpus (Sandhaus, 2008) is the largest summarization dataset currently available. It consists of carefully curated articles from a single source, The New York Times. The corpus contains several hundred thousand articles written between 1987–2007 that have paired summaries. The summaries were written for the corpus by library scientists, rather than at the time of publication. Our analysis in Section 4 reveals that the data are somewhat biased toward extractive strategies, making it particularly useful as an extractive summarization dataset. Despite this, limited work has used this dataset for summarization (Hong and Nenkova, 2014; Durrett et al., 2016; Paulus et al., 2017).

2.4 CNN / Daily Mail

The CNN / Daily Mail question answering dataset (Hermann et al., 2015) is frequently used for summarization. The dataset includes CNN and Daily Mail articles, each associated with several bullet point descriptions. When used in summarization, the bullet points are typically concatenated into a single summary. The dataset has been used for summarization as is (See et al., 2017), or after pre-processing for entity anonymization (Nallapati et al., 2017). This different usage makes comparisons between systems using these data challenging. Additionally, some systems use both CNN and Daily Mail for training (Nallapati et al., 2017; Paulus et al., 2017; See et al., 2017), whereas others use only Daily Mail articles (Nallapati et al., 2016b; Cheng and Lapata, 2016). Our analysis shows that the CNN / Daily Mail summaries have strong bias toward extraction (Section 4). Similar observations about the data were made by Chen et al. (2016) with respect to the question answering task.

3 Collecting NEWSROOM Summaries

The NEWSROOM dataset was collected using social media and search engine metadata. To create the dataset, we performed a Web-scale crawling of over 100 million pages from a set of online publishers. We identify newswire articles and use the summaries provided in the HTML metadata. These summaries were created to be used in search engines and social media.

We collected HTML pages and metadata using the Internet Archive (Archive.org), accessing archived pages of a large number of popular news, sports, and entertainment sites. Using Archive.org provides two key benefits. First, the archive provides an API that allows for collection of data across time, not limited to recently available articles. Second, the archived URLs of the dataset articles are immutable, allowing distribution of this dataset using a thin, URL-only list.

The publisher sites we crawled were selected using a combination of Alexa.com top overall sites, as well as Alexa’s top news sites. We supplemented the lists with older lists published by Google of the highest-traffic sites on the Web. We excluded sites such as Reddit that primarily aggregate rather than produce content, as well as publisher sites that proved to have few or no articles with summary metadata available, or have articles primarily in languages other than English. This process resulted in a set of 38 publishers that were included in the dataset.

---


4 Google removed this list in 2013, see: https://web.archive.org/web/2012/http://www.google.com/adplanner/static/top1000

2 https://github.com/abisee/cnn-dailymail
3.1 Content Scraping

We used two techniques to identify article pages from the selected publishers on Archive.org: the search API and index-page crawl. The API allows queries using URL pattern matching, which focuses article crawling on high-precision subdomains or paths. We used the API to search for content from the publisher domains, using specific patterns or post-processing filtering to ensure article content. In addition, we used Archive.org to retrieve the historical versions of the home page for all publisher domains. The archive has content from 1998 to 2017 with varying degrees of time resolution. We obtained at least one snapshot of each page for every available day. For each snapshot, we retrieved all articles listed on the page.

For both search and crawled URLs, we performed article de-duplication using URLs to control for varying URL fragments, query parameters, protocols, and ports. When performing the merge, we retained only the earliest article version available to prevent the collection of stale summaries that are not updated when articles are changed.

3.2 Content Extraction

Following identification and de-duplication, we extracted the article texts and summaries and further cleaned and filtered the dataset.

**Article Text** We used Readability\(^5\) to extract HTML body content. Readability uses HTML heuristics to extract the main content and title of a page, producing article text without extraneous HTML markup and images. Our preliminary testing, as well as comparison by Peters (2015), found Readability to be one of the highest accuracy content extraction algorithms available. To exclude inline advertising and image captions sometimes present in extractions, we applied additional filtering of paragraphs with fewer than five words. We excluded articles with no body text extracted.

**Summary Metadata** We extracted the article summaries from the metadata available in the HTML pages of articles. These summaries are often written by newsroom editors and journalists to appear in social media distribution and search results. While there is no standard metadata format for summaries online, common fields are often present in the page’s HTML. Popular metadata field types include: \og\ description, twitter: description, and description. In cases where different metadata summaries were available, and were different, we used the first field available according to the order above. We excluded articles with no summary text of any type. We also removed article-summary pairs with a high amount of precisely-overlapping text to remove rule-based automatically-generated summaries fully copied from the article (e.g., the first paragraph).

3.3 Building the Dataset

Our scraping and extraction process resulted in a set of 1,321,995 article-summary pairs. Simple dataset statistics are shown in Table 1. The data are divided into training (76%), development (8%), test (8%), and unreleased test (8%) datasets using a hash function of the article URL. We use the articles’ Archive.org URLs for lightweight distribution of the data. Archive.org is an ideal platform for distributing the data, encouraging its users to scrape its resources. We provide the extraction and analysis scripts used during data collection for reproducing the full dataset from the URL list.

### Table 1: Dataset Statistics

<table>
<thead>
<tr>
<th>Dataset Size</th>
<th>1,321,995 articles</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training Set Size</td>
<td>995,041 articles</td>
</tr>
<tr>
<td>Mean Article Length</td>
<td>658.6 words</td>
</tr>
<tr>
<td>Mean Summary Length</td>
<td>26.7 words</td>
</tr>
<tr>
<td>Total Vocabulary Size</td>
<td>6,925,712 words</td>
</tr>
<tr>
<td>Occurring 10+ Times</td>
<td>784,884 words</td>
</tr>
</tbody>
</table>

4 Data Analysis

**NEWSROOM** contains summaries from different topic domains, written by many authors, over the span of more than two decades. This diversity is an important aspect of the dataset. We analyze the data to quantify the differences in summarization styles and techniques between the different publications to show the importance of reflecting this diversity. In Sections 6 and 7, we examine the effect of the dataset diversity on the performance of a variety of summarization systems.

4.1 Characterizing Summarization Strategies

We examine summarization strategies using three measures that capture the degree of text overlap between the summary and article, and the rate of compression of the information conveyed.

Given an article text \(A = (a_1, a_2, \ldots, a_n)\) consisting of a sequence of tokens \(a_i\) and the corresponding article summary \(S = (s_1, s_2, \ldots, s_m)\) consisting of tokens \(s_i\), the set of extractive frag-
Figure 3: Procedure to compute the set \( F(A, S) \) of extractive phrases in summary \( S \) extracted from article \( A \). For each sequential token of the summary, \( s_i \), the procedure iterates through tokens of the text, \( a_j \). If tokens \( s_i \) and \( a_j \) match, the longest shared token sequence after \( s_i \) and \( a_j \) is marked as the extraction starting at \( s_i \).

\[
\text{function } F(A, S) \\
F \leftarrow \emptyset, (i, j) \leftarrow (1, 1) \\
\text{while } i \leq |S| \text{ do} \\
\quad f \leftarrow \emptyset \\
\quad \text{while } j \leq |A| \text{ do} \\
\quad \quad \text{if } s_i = a_j \text{ then} \\
\quad \quad \quad (i', j') \leftarrow (i, j) \\
\quad \quad \quad \text{while } s_{i'} = a_{j'} \text{ do} \\
\quad \quad \quad \quad (i', j') \leftarrow (i' + 1, j' + 1) \\
\quad \quad \quad \text{if } |f| < (i' - i - 1) \text{ then} \\
\quad \quad \quad \quad f \leftarrow (s_i \cdots s_{i' - 1}) \\
\quad \quad \quad j \leftarrow j' \\
\quad \quad \text{else} \\
\quad \quad \quad j \leftarrow j + 1 \\
\quad \quad (i, j) \leftarrow (i + \max\{|f|, 1|, 1) \\
\quad F \leftarrow F \cup \{f\} \\
\text{return } F
\]

Figure 3: Procedure to compute the set \( F(A, S) \) of extractive phrases in summary \( S \) extracted from article \( A \). For each sequential token of the summary, \( s_i \), the procedure iterates through tokens of the text, \( a_j \). If tokens \( s_i \) and \( a_j \) match, the longest shared token sequence after \( s_i \) and \( a_j \) is marked as the extraction starting at \( s_i \).

The density \( D \) of the fragment length: \[ D(f) = \frac{1}{|f|} \sum_{f \in F(A, S)} |f|^2. \]
For example, an article with a 10-word summary made of two extractive fragments of lengths 3 and 4 would have \( \text{Coverage}(A, S) = 0.7 \) and \( \text{Density}(A, S) = 2.5 \).

**Compression Ratio** We use a simple dimension of summarization, compression ratio, to further characterize summarization strategies. We define compression as the word ratio between the article and summary:

\[ \text{Compression}(A, S) = \frac{|A|}{|S|}. \]

Summarizing with higher compression is challenging as it requires capturing more precisely the critical aspects of the article text.

### 4.2 Analysis of Dataset Diversity

We use density, coverage, and compression to understand the distribution of human summarization techniques across different sources. Figure 4 shows the distributions of summaries for different domains in the Newsroom dataset, along with three major existing summarization datasets: DUC 2003-2004 (combined), CNN / Daily Mail, and the New York Times Corpus.

**Publication Diversity** Each Newsroom publication shows a unique distribution of summaries mixing extractive and abstractive strategies in varying amounts. For example, the third entry on the top row shows the summarization strategy used by BuzzFeed. The density (y-axis) is relatively low, meaning BuzzFeed summaries are unlikely to include long extractive fragments. While the coverage (x-axis) is more varied, BuzzFeed’s coverage tends to be lower, indicating that it frequently uses novel words in summaries. The publication plots in the figure are sorted by median compression ratio. We observe that publications with lower compression ratio (top-left of the figure) exhibit higher diversity along both dimensions of extractiveness. However, as the median compression ratio increases, the distributions become more con-
Figure 4: Density and coverage distributions across the different domains and existing datasets. NEWSROOM contains diverse summaries that exhibit a variety of summarization strategies. Each box is a normalized bivariate density plot of extractive fragment coverage (x-axis) and density (y-axis), the two measures of extraction described in Section 4.1. The top left corner of each plot shows the number of training set articles \( n \) and the median compression ratio \( c \) of the articles. For DUC and New York Times, which have no standard data splits, \( n \) is the total number of articles. Above, \textit{top left to bottom right}: Plots for each publication in the NEWSROOM dataset. We omit TMZ, Economist, and ABC for presentation. Below, \textit{left to right}: Plots for each summarization dataset showing increasing diversity of summaries along both dimensions of extraction in NEWSROOM.
centrated, indicating that summarization strategies become more rigid.

**Dataset Diversity**  Figure 4 demonstrates how DUC, CNN / Daily Mail, and the New York Times exhibit different human summarization strategies. DUC summarization is fairly similar to the high-compression newsrooms shown in the lower publication plots in Figure 4. However, DUC’s median compression ratio is much higher than all other datasets and NEWSROOM publications. The figure shows that CNN / Daily Mail and New York Times are skewed toward extractive summaries with lower compression ratios. CNN / Daily Mail shows higher coverage and density than all other datasets and publishers in our data. Compared to existing datasets, NEWSROOM covers a much larger range of summarization styles, ranging from both highly extractive to highly abstractive.

5 Performance of Existing Systems

We train and evaluate several summarization systems to understand the challenges of NEWSROOM and its usefulness for training systems. We evaluate three systems, each using a different summarization strategy with respect to extractiveness: fully extractive (TextRank), fully abstractive (Seq2Seq), and mixed (pointer-generator). We further study the performance of the pointer-generator model on NEWSROOM by training three systems using different dataset configurations. We compare these systems to two rule-based systems that provide baseline (Lede-3) and an extractive oracle (Fragments).

**Extractive: TextRank**  TextRank is a sentence-level extractive summarization system. The system was originally developed by Mihalcea and Tarau (2004) and was later further developed and improved by Barrios et al. (2016). TextRank uses an unsupervised sentence-ranking approach similar to Google PageRank (Page et al., 1999). TextRank picks a sequence of sentences from a text for the summary up to a maximum allowable length. While this maximum length is typically preset by the user, in order to optimize ROUGE scoring, we tune this parameter to optimize ROUGE-1 $F_1$-score on the NEWSROOM training data. We experimented with values between 1–200, and found the optimal value to be 50 words. We use tuned TextRank of in Tables 2, 3, and in the supplementary material.

**Abstractive: Seq2Seq / Attention**  Sequence-to-sequence models with attention (Cho et al., 2014; Sutskever et al., 2014; Bahdanau et al., 2014) have been applied to various language tasks, including summarization (Chopra et al., 2016; Nallapati et al., 2016a). The process by which the model produces tokens is abstractive, as there is no explicit mechanism to copy tokens from the input text. We train a TensorFlow implementation of the Rush et al. (2015) model using NEWSROOM.

**Mixed: Pointer-Generator**  The pointer-generator model (See et al., 2017) uses abstractive token generation and extractive token copying using a pointer mechanism (Vinyals et al., 2015; Gülçehre et al., 2016), keeping track of extractions using coverage (Tu et al., 2016). We evaluate three instances of this model by varying the training data: (1) Pointer-C: trained on the CNN / Daily Mail dataset; (2) Pointer-N: trained on the NEWSROOM dataset; and (3) Pointer-S: trained on a random subset of NEWSROOM training data the same size as the CNN / Daily Mail training. The last instance aims to understand the effects of dataset size and summary diversity.

**Lower Bound: Lede-3**  A common automatic summarization strategy of online publications is to copy the first sentence, first paragraph, or first $k$ words of the text and treat this as the summary. Following prior work (See et al., 2017; Nallapati et al., 2017), we use the Lede-3 baseline, in which the first three sentences of the text are returned as the summary. Though simple, this baseline is competitive with state-of-the-art systems.

**Extractive Oracle: Fragments**  This system has access to the reference summary. Given an article $A$ and its summary $S$, the system computes $F(A, S)$ (Section 4). Fragments concatenates the fragments in $F(A, S)$ in the order they appear in the summary, representing the best possible performance of an ideal extractive system. Only systems that are capable of abstractive reasoning can outperform the ROUGE scores of Fragments.

6 Automatic Evaluation

We study model performance of NEWSROOM, CNN / Daily Mail, and the combined DUC 2003 and 2004 datasets. We use the five systems described in Section 5, including the extractive oracle. We also evaluate the systems using subsets of
Table 2: ROUGE-1, ROUGE-2, and ROUGE-L scores for baselines and systems on two common existing datasets, the combined DUC 2003 & 2004 datasets and CNN / Daily Mail dataset, and the released (T) and unreleased (U) test sets of NEWSROOM. The best results for non-baseline systems in the lower parts of the table are in bold.

Table 3: Performance of the baselines and systems on the three extractiveness subsets of the NEWSROOM development set, and the overall scores of systems on the full development set (D). The best results for non-baseline systems in the lower parts of the table are in bold.

Table 2 shows results for summarization systems on DUC, CNN / Daily Mail, and NEWSROOM. In nearly all cases, the fully extractive Lede-3 baseline produces the most successful summaries, with the exception of the relatively extractive DUC. Among models, NEWSROOM-trained Pointer-N performs best on all datasets other than CNN / Daily Mail, an out-of-domain dataset. Pointer-C, which has access to only a limited subset of NEWSROOM, performs worse than Pointer-N on average. However, despite not being trained on CNN / Daily Mail, Pointer-S outperforms Pointer-C on its own data under ROUGE-N and is competitive under ROUGE-L. Finally, both Pointer-N and Pointer-S outperform other systems and baselines on DUC, whereas Pointer-C does not outperform Lede-3.

Table 3 shows development results on the NEWSROOM data for different levels of extractiveness. Pointer-N outperforms the remaining models across all extractive subsets of NEWSROOM and, in the case of the abstractive subset, exceeds the performance of Lede-3. The success of Pointer-N and Pointer-S in generalizing and outperforming models on DUC and CNN / Daily Mail indicates the usefulness of NEWSROOM in generalizing to out-of-domain data. Similar subset analysis for our other two measures, coverage and compression, are included in the supplementary material.

7 Human Evaluation

ROUGE scores systems using frequencies of shared n-grams. Evaluating systems with ROUGE alone biases scoring against abstractive systems, which rely more on paraphrasing. To overcome this limitation, we provide human evaluation of the different systems on NEWSROOM. While human evaluation is still uncommon in summarization work, developing a benchmark dataset presents an opportunity for developing an accompanying protocol for human evaluation.

Our evaluation method is centered around three objectives: (1) distinguishing between syntactic and semantic summarization quality, (2) providing a reliable (consistent and replicable) measurement, and (3) allowing for portability such that the
Table 4: The prompts given to Amazon Mechanical Turk crowdworkers for evaluating each summary.

<table>
<thead>
<tr>
<th>DIMENSION</th>
<th>PROMPT</th>
</tr>
</thead>
<tbody>
<tr>
<td>Informativeness</td>
<td>How well does the summary capture the key points of the article?</td>
</tr>
<tr>
<td>Relevance</td>
<td>Are the details provided by the summary consistent with details in the article?</td>
</tr>
<tr>
<td>Fluency</td>
<td>Are the individual sentences of the summary well-written and grammatical?</td>
</tr>
<tr>
<td>Coherence</td>
<td>Do phrases and sentences of the summary fit together and make sense collectively?</td>
</tr>
</tbody>
</table>

Table 5: Average performance of systems as scored by human evaluators. Each summary was scored by three different evaluators. Dimensions, from left to right: informativeness, relevance, fluency, and coherence, and a mean of the four dimensions for each system.

<table>
<thead>
<tr>
<th></th>
<th>SEMANTIC</th>
<th>SYNTACTIC</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>INF</td>
<td>REL</td>
</tr>
<tr>
<td>Lede-3</td>
<td>3.98</td>
<td>4.13</td>
</tr>
<tr>
<td>Fragments</td>
<td>2.91</td>
<td>3.26</td>
</tr>
<tr>
<td>TextRank</td>
<td>3.61</td>
<td>3.92</td>
</tr>
<tr>
<td>Abs-N</td>
<td>2.09</td>
<td>2.35</td>
</tr>
<tr>
<td>Pointer-C</td>
<td>3.55</td>
<td>3.78</td>
</tr>
<tr>
<td>Pointer-S</td>
<td>3.77</td>
<td><strong>4.02</strong></td>
</tr>
<tr>
<td>Pointer-N</td>
<td>3.36</td>
<td>3.82</td>
</tr>
</tbody>
</table>

We select two semantic and two syntactic dimensions for evaluation based on experiments with evaluation tasks by Paulus et al. (2017) and Tan et al. (2017). The two semantic dimensions, summary informativeness (INF) and relevance (REL), measure whether the system-generated text is useful as a summary, and appropriate for the source text, respectively. The two syntactic dimensions, fluency (FLU) and coherence (COH), measure whether individual sentences or phrases of the summary are well-written and whether the summary as a whole makes sense respectively. Evaluation was performed on 60 summaries, 20 from each extractive NEWSROOM subset. Each system-article pair was evaluated by three unique raters. Exact prompts given to raters for each dimension are shown in Table 4.

Table 5 shows the mean score given to each system under each of the four dimensions, as well as the mean overall score (rightmost column). No summarization system exceeded the scores given to the Lede-3 baseline. However, the extractive oracle designed to maximize $n$-gram based evaluation performed worse than the majority of systems under human evaluation. While the fully abstractive Abs-N model performed very poorly under automatic evaluation, it fared slightly better when scored by humans. TextRank received the highest overall score. TextRank generates full sentences extracted from the article, and raters preferred TextRank primarily for its fluency and coherence. The pointer-generator models do not have this advantage, and raters did not find the pointer-generator models to be as syntactically sound as TextRank. However, raters preferred the informativeness and relevance of the Pointer-S and Pointer-N models, though not the Pointer-C model, over TextRank.

8 Conclusion

We present NEWSROOM, a dataset of articles and their summaries written in the newsrooms of online publications. NEWSROOM is the largest summarization dataset available to date, and exhibits a wide variety of human summarization strategies. Our proposed measures and the analysis of strategies used by different publications and articles propose new directions for evaluating the difficulty of summarization tasks and for developing future summarization models. We show that the dataset’s diversity of summaries presents a new challenge to summarization systems. Finally, we find that using NEWSROOM to train an existing state-of-art mixed-strategy summarization model results in performance improvements on out-of-domain data. The NEWSROOM dataset is available online at summar.ies.

Acknowledgements

This work is funded by Oath as part of the Connected Experiences Laboratory and by a Google Research Award. We thank the anonymous reviewers for their feedback.

References


Additional Evaluation

In Section 4, we discuss three measures of summarization diversity: coverage, density, and compression. In addition to quantifying diversity of summarization strategies, these measures are helpful for system error analysis. We use the density measurement to understand how system performance varies when compared against references using different extractive strategies by subdividing NEWSROOM into three subsets by extractiveness and evaluating using ROUGE on each. We show here a similar analysis using the remaining two measures, coverage and compression. Results for subsets based on coverage and compression are shown in Tables 6 and 7.

### Low Coverage

<table>
<thead>
<tr>
<th>System</th>
<th>R-1</th>
<th>R-2</th>
<th>R-L</th>
<th>R-1</th>
<th>R-2</th>
<th>R-L</th>
<th>R-1</th>
<th>R-2</th>
<th>R-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lede-3</td>
<td>15.07</td>
<td>4.02</td>
<td>12.66</td>
<td>29.66</td>
<td>18.69</td>
<td>26.98</td>
<td>46.89</td>
<td>41.25</td>
<td>45.77</td>
</tr>
<tr>
<td>Fragments</td>
<td>72.45</td>
<td>46.16</td>
<td>72.45</td>
<td>93.41</td>
<td>83.08</td>
<td>93.41</td>
<td>99.13</td>
<td>98.16</td>
<td>99.13</td>
</tr>
<tr>
<td>TextRank</td>
<td>14.43</td>
<td>2.80</td>
<td>11.36</td>
<td>23.62</td>
<td>9.48</td>
<td>19.27</td>
<td>30.15</td>
<td>17.04</td>
<td>26.18</td>
</tr>
<tr>
<td>Abs-N</td>
<td>6.25</td>
<td>1.09</td>
<td>5.72</td>
<td>5.61</td>
<td>0.15</td>
<td>5.05</td>
<td>6.10</td>
<td>0.19</td>
<td>5.40</td>
</tr>
<tr>
<td>Pointer-C</td>
<td>13.99</td>
<td>2.46</td>
<td>11.57</td>
<td>21.70</td>
<td>8.06</td>
<td>18.47</td>
<td>25.80</td>
<td>12.06</td>
<td>22.57</td>
</tr>
<tr>
<td>Pointer-S</td>
<td>15.16</td>
<td>3.63</td>
<td>11.61</td>
<td>26.95</td>
<td>14.51</td>
<td>22.30</td>
<td>32.42</td>
<td>20.77</td>
<td>28.15</td>
</tr>
<tr>
<td>Pointer-N</td>
<td>16.07</td>
<td>3.78</td>
<td>12.85</td>
<td>28.79</td>
<td>15.31</td>
<td>24.79</td>
<td>34.03</td>
<td>21.67</td>
<td>30.62</td>
</tr>
</tbody>
</table>

Table 6: Performance of the baselines and systems on the three coverage subsets of the NEWSROOM development set. Article-summary pairs with low coverage have reference summaries that borrow words less frequently from their texts and contain more novel words and phrases. Article-summary pairs with high coverage borrow more words from their text and include fewer novel words and phrases.

### Low Compression

<table>
<thead>
<tr>
<th>System</th>
<th>R-1</th>
<th>R-2</th>
<th>R-L</th>
<th>R-1</th>
<th>R-2</th>
<th>R-L</th>
<th>R-1</th>
<th>R-2</th>
<th>R-L</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lede-3</td>
<td>42.89</td>
<td>34.91</td>
<td>41.06</td>
<td>30.62</td>
<td>20.77</td>
<td>28.30</td>
<td>18.57</td>
<td>8.83</td>
<td>16.53</td>
</tr>
<tr>
<td>Fragments</td>
<td>87.78</td>
<td>77.20</td>
<td>87.78</td>
<td>89.73</td>
<td>77.66</td>
<td>89.73</td>
<td>87.88</td>
<td>73.34</td>
<td>87.88</td>
</tr>
<tr>
<td>TextRank</td>
<td>30.35</td>
<td>17.51</td>
<td>26.67</td>
<td>22.98</td>
<td>8.69</td>
<td>18.56</td>
<td>15.07</td>
<td>3.31</td>
<td>11.78</td>
</tr>
<tr>
<td>Abs-N</td>
<td>6.27</td>
<td>0.75</td>
<td>5.65</td>
<td>6.22</td>
<td>0.52</td>
<td>5.60</td>
<td>5.48</td>
<td>0.18</td>
<td>4.93</td>
</tr>
<tr>
<td>Pointer-C</td>
<td>27.47</td>
<td>13.49</td>
<td>24.18</td>
<td>20.05</td>
<td>6.25</td>
<td>16.76</td>
<td>14.07</td>
<td>2.89</td>
<td>11.76</td>
</tr>
<tr>
<td>Pointer-S</td>
<td>35.42</td>
<td>23.43</td>
<td>30.89</td>
<td>24.11</td>
<td>11.28</td>
<td>19.45</td>
<td>15.31</td>
<td>4.46</td>
<td>11.98</td>
</tr>
<tr>
<td>Pointer-N</td>
<td>36.96</td>
<td>24.52</td>
<td>33.43</td>
<td>25.56</td>
<td>11.68</td>
<td>21.47</td>
<td>16.57</td>
<td>4.72</td>
<td>13.52</td>
</tr>
</tbody>
</table>

Table 7: Performance of the baselines and systems on the three compression subsets of the NEWSROOM development set. Article-summary pairs with low compression have longer reference summaries with respect to their texts. Article-summary pairs with high compression have shorter reference summaries with respect to their texts.