# SummScreen: A Dataset for Abstractive Screenplay Summarization

**Mingda Chen**[1]    **Zewei Chu**[*]    **Sam Wiseman**[2†]    **Kevin Gimpel**[1]

[1]Toyota Technological Institute at Chicago, IL, USA
[2]Duke University, NC, USA

{mchen,kgimpel}@ttic.edu, swiseman@cs.duke.edu, zeweichu@gmail.com

## Abstract

We introduce SUMMSCREEN, a summarization dataset comprised of pairs of TV series transcripts and human written recaps. The dataset provides a challenging testbed for abstractive summarization for several reasons. Plot details are often expressed indirectly in character dialogues and may be scattered across the entirety of the transcript. These details must be found and integrated to form the succinct plot descriptions in the recaps. Also, TV scripts contain content that does not directly pertain to the central plot but rather serves to develop characters or provide comic relief. This information is rarely contained in recaps. Since characters are fundamental to TV series, we also propose two entity-centric evaluation metrics. Empirically, we characterize the dataset by evaluating several methods, including neural models and those based on nearest neighbors. An oracle extractive approach outperforms all benchmarked models according to automatic metrics, showing that the neural models are unable to fully exploit the input transcripts. Human evaluation and qualitative analysis reveal that our non-oracle models are competitive with their oracle counterparts in terms of generating faithful plot events and can benefit from better content selectors. Both oracle and non-oracle models generate unfaithful facts, suggesting future research directions.[1]

## 1 Introduction

Abstractive summarization aims to produce a summary that concisely expresses key points of the input document rather than merely extracting pieces of it. Existing datasets are constructed from various domains, such as news (Sandhaus, 2008; Hermann

---

[1]SUMMSCREEN is available at https://github.com/mingdachen/SummScreen

**Transcript:**

[ The apartment ]

Sheldon : What color would you like to be ?
Leonard : Well , I 'd like to be green , but you know you always take it .
Sheldon : That 's not true . Any color 's fine with me . Yeah , I could be a - a combination of blue and yellow .
Leonard : Blue and yellow make green .
Sheldon : Well , then it 's settled .

Penny : Hi . Ready to go ?
Sheldon : Oh , good news , we ordered lunch , so we can all stay here and play Lord of the Rings Risk .
Amy : Sheldon , we said that we would play games with you tonight .
Sheldon : Oh , no , we 'll still be playing it tonight , this game can easily take eight hours .

Penny : Sweetie , you really thought I 'd want to do this ?
Leonard : No .
Penny : Well , did you tell him that ?
Leonard : Yes .
Penny : Did you say it out loud with words ?
Leonard : No .
Penny : I do n't want to spend the whole day playing a board game .

…

**Recap:**

Sheldon and Leonard are happy playing a board game until Amy and Penny say they are tired of doing what the guys want …

Figure 1: Excerpts from an example from SUMM-SCREEN. The transcript and recap are from the TV show "The Big Bang Theory". Generating this sentence in the recap requires discerning the characters' feelings (clues in the transcript are underlined) about playing the board game (references are shown in red). Colored boxes indicate utterances belonging to the same conversations.

et al., 2015; Rush et al., 2015; Narayan et al., 2018; Grusky et al., 2018), online forums (Völske et al., 2017), meeting dialogues (Janin et al., 2003; Carletta et al., 2005), and webpages (Chen et al., 2020). However, few datasets exist for abstractive summarization of narrative text, which focuses on entities and dialogue among entities, with plot details often communicated indirectly via dialogue. In this work, we build SUMMSCREEN, an abstractive summarization dataset combining TV series transcripts and episode recaps. Figure 1 shows an example from SUMMSCREEN.

Several aspects of SUMMSCREEN make it a challenging testbed for abstractive summarization. First, the relationship between character dialogue and plot details is not straightforward. Plot events

8602

are often expressed indirectly in dialogue, and dialogue contains other information that is not directly relevant to the plot, such as character development and humor. Also, a typical episode has multiple subplots that proceed in parallel, with consecutive scenes often describing different subplots. Solving SUMMSCREEN requires drawing information from utterances across a wide range of the input and integrating the information to form concise plot descriptions. Moreover, since actual TV episodes ground their scripts with audio-visual accompaniment, many details may be omitted from the transcript itself. This omission of details and the other challenging aspects mentioned above have inspired research into other NLP tasks on TV show transcripts, such as entity tracking (Chen and Choi, 2016; Choi and Chen, 2018) and coreference resolution (Chen et al., 2017; Zhou and Choi, 2018).

Another prominent characteristic of TV series transcripts is their focus on characters. To reflect this aspect, we propose two entity-centric metrics to evaluate the quality of generated plot summaries. One is based on bags of characters, which measures the overlap of the characters that appear in both the generated and reference recaps. The other metric measures character relations: the overlap of cooccurrences of character pairs in generations and recaps.

We empirically evaluate several types of methods on SUMMSCREEN. We consider nearest neighbor models, which look up similar transcripts or recaps, neural abstractive summarization models, and hybrid models, which use the nearest neighbor models as content selectors followed by abstractive summarization. Oracle extractive approaches outperform all models on all the automatic metrics. These results suggest that the benchmarked methods are unable to fully exploit the input transcripts and that improving content selection may be a promising research direction.

Human evaluations show that our non-oracle hybrid models are competitive with their oracle counterparts in terms of generating faithful plot events. Hybrid models may be promising approaches for future research. Qualitative analysis shows that neural models tend to generate generic summaries, hybrid models can benefit from better content selection, and hybrid models sometimes generate unfaithful details.

## 2 Related Work

There has been prior work on *extractive* screenplay summarization (Gorinski and Lapata, 2015; Papalampidi et al., 2020), and analyzing crime drama (Frermann et al., 2018). The majority of TV show transcripts are dialogues, relating our work to prior work on dialogue and meeting summarization. Relevant datasets have been studied for medical dialogues (Joshi et al., 2020; Krishna et al., 2021), chitchat (SAMSum; Gliwa et al., 2019), podcasts (Clifton et al., 2020), meetings (AMI; Carletta et al., 2005; ICSI; Janin et al., 2003; QMSum; Zhong et al., 2021), livestreams (StreamHover; Cho et al., 2021), online forums (ForumSum; Khalman et al., 2021) and news interviews (MediaSum; Zhu et al., 2021).

There have been attempts in summarizing long-form text (other than screenplays), such as books (Mihalcea and Ceylan, 2007), scientific articles (PubMed and arXiv; Cohan et al., 2018), multiple news articles (Multi-News; Fabbri et al., 2019), opinionated text (RottenTomatoes; Wang and Ling, 2016), patents (Sharma et al., 2019), TV show stories (TVRecap; Chen and Gimpel, 2021) and (extractive summarization of) chapters of novels (Ladhak et al., 2020). More detailed discussion on the differences between these datasets and SUMMSCREEN is in the next section.

Recently there have been efforts on adapting resources for TV shows for different tasks, including question answering (Ma et al., 2018; Yang and Choi, 2019), speaker identification (Ma et al., 2017), sarcasm detection (Joshi et al., 2016), emotion detection (Zahiri and Choi, 2017; Hsu and Ku, 2018), character relation extraction (Yu et al., 2020), and story generation (Chen and Gimpel, 2021).

## 3 SUMMSCREEN

An instance in SUMMSCREEN contains a transcript from TV series and its corresponding recap. The transcripts consist of dialogue utterances with speaker names, and descriptions of scenes or character actions. The recaps are human-written summaries of the corresponding transcripts. Figure 1 shows an example in SUMMSCREEN from the TV show "The Big Bang Theory". The transcript documents a dialogue involving four characters (Sheldon, Leonard, Penny, and Amy) about playing a board game, and the recap summarizes the dialogue into sentences.

| | uni. | bi. | tri. | four. | src. | tgt. |
|---|---|---|---|---|---|---|
| SUMMSCREEN | | | | | | |
| FD | 81.6 | 29.9 | 5.6 | 1.3 | 7.6k | 113.7 |
| TMS | 86.5 | 34.1 | 6.9 | 2.1 | 6.4k | 380.6 |
| Other summarization datasets | | | | | | |
| XSum[†] | 64.2 | 16.6 | 4.5 | 1.5 | 431.1 | 23.3 |
| CNNDM[§] | 80.5 | 43.1 | 25.6 | 17.2 | 810.6 | 56.2 |
| MNews[§] | 82.2 | 42.9 | 24.3 | 17.7 | 2.1k | 264.7 |

Table 1: Fraction (%) of n-grams in the *output summaries* that also appear in the inputs, and the average numbers of tokens for the inputs and outputs. Datasets with smaller fractions of overlapping n-grams tend to favor abstractive summarization approaches. Results marked by † and § are from Narayan et al. (2018) and Fabbri et al. (2019) respectively.

## 3.1 Dataset Construction

We use two sources to construct SUMMSCREEN: The TV MegaSite, Inc. (TMS)[2] and ForeverDreaming (FD),[3] both of which provide community-contributed transcripts. As FD does not provide recaps, we obtain recaps of FD shows from Wikipedia and TVMaze.[4] To ensure dataset quality of SUMMSCREEN, we filter out instances based on two criteria. First, the overlap ratio of TV show characters appearing in the recap and its transcript should be higher than 85%. We use this criterion to ensure that the alignments between recaps and transcripts are correct. Second, the number of transcript lines that have speaker information ("character utterances") should be larger than 100. We use this criterion to eliminate transcripts that are essentially subtitles, i.e., utterances without speaker information. In practice, for each transcript line, if a colon symbol appears in the first 8 tokens and there exists at least one character name in front of the colon symbol, we will count it as a character utterance. We note that FD and TMS do not have overlapping TV series.

In Table 1, we compute n-gram overlap ratios between recaps and transcripts for measuring the abstractiveness of SUMMSCREEN. From the results, We find that despite SUMMSCREEN has longer summaries, its fraction of overlapping four-gram is comparable to XSum (Narayan et al., 2018) which is known for abstractiveness, suggesting that SUMMSCREEN favors abstractive approaches.

Table 2 shows data statistics and Figure 2 shows

| | FD | TMS |
|---|---|---|
| number of shows | 88 | 10 |
| number of episodes | 4348 | 22503 |
| min. # episodes per show | 1 | 168 |
| max. # episodes per show | 568 | 3784 |
| median # episodes per show | 9.0 | 1973.5 |
| avg. # episodes per show | 49.4 | 2250.0 |
| avg. # tokens in recaps | 113.7 | 380.6 |
| avg. # tokens in transcripts | 7605.4 | 6420.7 |
| avg. # lines in transcripts | 447.6 | 360.8 |
| avg. # char. utterances in transcripts | 330.7 | 327.0 |
| avg. # uniq. char. in recaps | 5.8 | 14.3 |
| avg. # uniq. char. in transcripts | 20.6 | 29.8 |

Table 2: Detailed dataset statistics for SUMMSCREEN.

| Genre | Count |
|---|---|
| Drama | 65 |
| Romance | 24 |
| Comedy | 23 |
| Crime | 18 |
| Action | 15 |
| Science-Fiction | 12 |
| Adventure | 9 |
| Supernatural | 9 |
| Mystery | 8 |
| Thriller | 5 |
| Family | 5 |
| Medical | 5 |
| Fantasy | 4 |
| Horror | 4 |
| History | 3 |
| Sports | 3 |
| Western | 3 |
| Children | 2 |
| Legal | 2 |
| Espionage | 1 |
| Music | 1 |

| Genre | Count |
|---|---|
| Drama | 10 |
| Romance | 6 |
| Family | 4 |
| Medical | 1 |

Figure 2: Left: TV show genres from ForeverDreaming. Right: TV show genres from TVMegaSite.

the genres of the TV shows from the two sources.[5] When computing the number of unique characters in TV shows, we first collect the character names from TVMaze and the named entities[6] preceding the colon symbols in transcripts. We then perform string matching to obtain numbers of TV show characters in recaps and transcripts. From these two tables, we observe that FD and TMS are different in many aspects. First, FD covers more diverse genres than TMS. This is partly due to the fact that TV shows from TMS are soap operas. Second, transcripts from FD are longer, which is caused by the fact that the transcripts from FD tend to have more descriptions about environments or character actions, whereas the ones from TMS are mostly

| ForeverDreaming | Train | Dev | Test |
|---|---|---|---|
| # shows | 66 | 78 | 81 |
| # episodes | 3673 | 338 | 337 |
| TVMegaSite | Train | Dev | Test |
| # shows | 10 | 10 | 10 |
| # episodes | 18915 | 1795 | 1793 |

Table 3: Statistics of train/dev/test splits for Forever-Dreaming and TVMegaSite.

made up of dialogue (see Table 2). Third, recaps from FD are shorter whereas recaps from TMS seek to cover more details. Fourth, writing styles are more diverse in FD than those in TMS. In light of these differences, we treat FD and TMS as different datasets in the following experiments.

We create train/dev/test splits for FD and TMS by ensuring the ratio to be roughly 10:1:1, and filter out instances in the dev/test splits if the reference texts are shorter than 30 word tokens. The statistics of the splits are shown in Table 3.

## 3.2 Dataset Comparison

We compare SUMMSCREEN to other abstractive dialogue summarization datasets in Table 4. SUMM-SCREEN differs from other datasets in several ways:

1. Compared to recently proposed large-scale dialogue summarization datasets (i.e., SAMsum and MediaSUM), SUMMSCREEN has longer source inputs.

2. Compared to other dialogue summarization datasets, SUMMSCREEN has larger numbers of speakers per instance. The TV series genre focuses on narrative, which is typically entity-centric and can include multiple parallel subplots in a single episode.

3. Compared to AMI, ICSI and QMSum, which are long-input meeting summarization datasets, SUMMSCREEN has far more instances.

4. Unlike most of the other datasets, SUMM-SCREEN contains many episodes of a single show (e.g., more than 3k episodes for TMS). This episodic structure could be used to model character arcs, the evolution of character personality traits and character relationships over episodes, among others.

Properties (1) and (2) above make extracting information from transcripts more challenging than other datasets. The third property means that

SUMMSCREEN is large enough to train and evaluate neural methods.

The Spotify Podcast Dataset (Clifton et al., 2020) and StreamHover (Cho et al., 2021) are similar to SUMMSCREEN in that they contain transcribed speech and summaries. However, the transcriptions are obtained automatically and therefore contain errors.[7] The datasets therefore involve speech processing (or at least handling speech recognition errors) compared to SUMMSCREEN, which has human-written transcripts.

Since MediaSum is constructed from news transcripts, it is the most similar dataset in Table 4 to SUMMSCREEN. However, the summaries in MediaSum are twenty times shorter than those in SUMM-SCREEN, and the average number of speakers per instance is only a quarter of that in SUMMSCREEN. Furthermore, our results in Sec. 5.2 indicate that our dataset is much harder than MediaSum as the pretrained models perform worse on our dataset than on MediaSum according to automatic metrics. More detailed analysis is in the next subsection.

## 3.3 Dataset Challenges

In this subsection, we qualitatively analyze the challenging aspects of SUMMSCREEN. Since the transcripts focus on dialogue among characters, along with limited descriptions of scenes and actions, it leads to the challenge that plot information is not stated explicitly but rather only implied in the dialogue. For example, the transcript in Figure 1 does not explicitly describe what Sheldon and Leonard are playing. However, it is implied by Sheldon when he mentions playing "Lord of the Rings Risk," and later by Penny when she says that she does not "want to spend the whole day playing a board game."

A related challenge is the need to understand the context in which characters' utterances are situated. In the example, the recap describes four characters taking sides regarding playing a board game. The transcript expresses the characters' sentiments through their interactions with one another. The conflict does not occur until Sheldon proposes to "stay here and play Lord of the Rings Risk", and it becomes more apparent when Penny mentions she does not want to play the board game. Given the context, Leonard's series of yes and no responses to Penny's questions is largely due to the awkward sit-

---

[7]For this reason, we do not include their statistics in Table 4.

| | # instances | # tokens (input) | # tokens (summary) | # speakers | Domain |
|---|---|---|---|---|---|
| Multi-News | 56.2k | 2103.5 | 264.7 | - | News |
| RottenTomatoes | 3.7k | 2124.7 | 22.2 | - | Reviews |
| arXiv | 215k | 4938.0 | 220.0 | - | Science |
| PubMed | 113k | 3016.0 | 203.0 | - | Science |
| GovReport | 19.5k | 9409.4 | 553.4 | - | Government Reports |
| TVRecap | 29.0k | 1868.7 | 221.6 | - | Television Series |
| SAMSum | 16.4k | 83.9 | 20.3 | 2.2 | Chitchat |
| ForumSum | 4.1k | 303.5 | 36.0 | 6.7 | Forum Messages |
| MediaSum | 463.6k | 1553.7 | 14.4 | 6.5 | News Interviews |
| AMI | 137 | 4757.0 | 322.0 | 4.0 | Meetings |
| ICSI | 59 | 10189.0 | 534.0 | 6.2 | Meetings |
| QMSum | 1.8k | 9069.8 | 69.6 | 9.2 | Meetings |
| SUMMSCREEN | 26.9k | 6612.5 | 337.4 | 28.3 | Television Series |

Table 4: Statistics for datasets focusing on abstractive summarization for long-form text or dialogue. The numbers are averaged over instances. We omit number of speakers for datasets that do not contain dialogue. SUMMSCREEN combines long source inputs, large numbers of speakers, and a moderate number of instances.

119 DOCTOR : Camera ! Camera ! ( takes camera from ALEC 'S unresisting hands )
…
212 The DOCTOR turns around and continues to take photos with the camera …
…
256 DOCTOR : The TARDIS is like a cat - a bit slow to trust ( runs to TARDIS ) but you 'll get there in the end . ( goes inside )
…
336 DOCTOR : Right ! Done ! That 's it … She 's not a ghost ... but she 's definitely a lost soul . ( walks over to screen ) Her name 's Hila Tacorian . She 's a pioneer , a time traveller - or at least she will be , in a few hundred years .
**Summary**: … the Doctor borrows Alec 's camera and uses the TARDIS to take pictures of the mansion 's location throughout time . Thanks to this , the Doctor learns it 's not a ghost in the pictures , but a time traveler named Hila Tacorian …
**TV show**: Doctor who

251 ( … Bohannon pulls out another nail and then another … )
252 ( The Swede is unlocking the door . )
253 ( Bohannon slips through the hole in the floor … )
254 ( The Swede pulls open the door and sees that Bohannon has escaped … )
255 ( Bohannon crouches under the train platform … )
256 ( … they search around the platform looking for Bohannon but he has already moved on . )
257 ( Bohannon blends in with a group of workers . )
258 [ Scene break ]
…
410 [ CUT TO : INT . Durant 's car ]
411 ( … Bohannon stands just behind the door of the car . Durant turns , confused but not startled to see him standing there . )
412 Bohannon : ( nods ) Mr. Durant .
**Summary**: … Cullen escapes the captivity of the Swede and goes to Durant 's office …
**TV show**: Hell on wheels

Figure 3: Two excerpts from SUMMSCREEN showing that generating summaries from TV show transcripts requires drawing information from a wide range of the input transcripts. We only show lines in the transcripts that are closely related to the shown parts of summaries. The number at the beginning of each line is the line number in the original transcript. For the first instance, we omit a few lines containing clues about the doctor taking pictures of the mansion at different times due to space constraints.

uation, and it actually shows that he is happy playing the game as he and Sheldon are doing so at the beginning of the scene. Similarly, Amy mentions their previous agreement with Sheldon as a way of politely declining Sheldon's plan. The sentiments of characters are not necessarily easily discernible from their utterances but rather must be inferred using context and knowledge about the characters.

Another challenge in SUMMSCREEN is the need to draw information from a wide range of the input transcripts, which arises for two primary reasons. First, there are many utterances that serve a purpose other than driving the plot forward. They may help to develop characters or character relationships, or to add humor or suspense. These lines enrich the narrative but their information content is often omitted from the summaries. For example, in the first instance in Figure 3, we show key lines from the transcript that pertain to the excerpt of the summary. There are many other lines between the lines shown, which are conversations between the doctor and other characters. This property necessitates the models' ability to correctly attend to major events across the transcript when generating summaries. The pattern can also be observed in Table 2 through the differences between the number of unique characters in recaps and transcripts. More than half of the characters in the transcripts are not contained in the recaps.

The second reason why information needs to be combined across wide ranges of the input relates to scene breaks and multiple plots. As a TV show often narrates a few plots in parallel, scene breaks are used to separate the stories. The discontinuity sometimes requires models to connect subplots hundreds of lines apart. For example, for the second instance in Figure 3, the show uses scene breaks to express what is happening when Cullen

Bohannon escapes from the Swede, which is why there are almost two hundred lines between Cullen Bohannon's escape and his appearance at Durant's office.

# 4 Approaches

In this section, we describe modeling approaches that we benchmark on SUMMSCREEN. We note that since the meaning of sentences in transcripts is highly context-dependent, extractive summarization approaches are not expected to be useful for this dataset. We report the results from nearest neighbor-based extractive summarizers mostly for characterizing the dataset.

## 4.1 Neural Models

We use transformer based sequence-to-sequence architectures (Vaswani et al., 2017). Because transcripts are quite long, We limit the number of encoder hidden vectors that are used in the decoder's attention mechanism. To do so, when encoding transcripts, we first append a special token "[EOS]" to each line of the transcript, and then linearize the transcript. We then only feed the vectors representing these special tokens to the decoder. We use the Longformer (Beltagy et al., 2020) as our encoder architecture, and set the "[EOS]" tokens to use global attention. For our decoders, we use the standard transformer architecture.

## 4.2 Nearest Neighbor Models

We consider two metrics when finding nearest neighbors: BM25 (Robertson et al., 1995) (a popular metric for information retrieval), and ROUGE scores (Lin, 2004). We use ROUGE scores as they are used for evaluation, and we use BM25 because it is designed for retrieving long documents whereas ROUGE scores are not. When using ROUGE scores, we use the average of ROUGE-1, ROUGE-2, and ROUGE-L. We consider three types of nearest neighbor search: transcript-to-transcript, recap-to-transcript, and recap-to-recap.

**Recap-to-transcript (NNM-r2t).** We use each sentence in the recap as queries and the lines in the corresponding transcript as candidates. The generation is formed by the nearest neighbors for each sentence. We use BM25 or ROUGE scores as the metric. This method can serve as an oracle result for an extractive summarization system, showing roughly how much information can be extracted at the utterance level from the source transcript.

**Transcript-to-transcript (NNM-t2t).** We use the transcripts in the test sets as queries, the transcripts in the training sets as candidates, and then find the nearest neighbors using BM25. The generations are the corresponding recaps. This baseline measures the similarity of instances between training and test splits.

**Recap-to-recap (NNM-r2r).** This setting is similar to the "transcript-to-transcript" setting, but we use recaps for both queries and candidates, and we use ROUGE and our proposed entity-centric scores (see Sec. 5.1 for more details) as the metric. When using the entity metrics, we use the average of the 4 metric scores. This is an oracle baseline of the "transcript-to-transcript" setting and also measures the similarity of the splits.

## 4.3 Hybrid Models

As content selection has been shown to be helpful in prior work (Gehrmann et al., 2018; Liu et al., 2018), we use the "recap-to-transcript" nearest neighbor model and BM25 as the metric to select the most salient content from transcripts, and then apply neural models to the selected content when performing generation. As these methods combine nearest neighbor models and neural models, we refer to them as hybrid models.

In particular, for each sentence in the recap, we find the top three most similar lines in the transcript, include two extra lines that come before or after the selected lines as context, and also include a line that is retrieved by using the whole recap. As the selected content is significantly shorter than the original transcript, it allows us to use pretrained models.[8] Therefore, in this setting, we fine-tune a pretrained BART-large model (Lewis et al., 2020). We note that as the nearest neighbor models rely on the gold standard recaps, this hybrid model demonstrates an approximate upper bound on performance when using powerful content selectors.[9]

To establish a non-oracle baseline, we train neural models to predict the selected lines, and then fine-tune BART-large models on the predicted lines. Details of the architecture for this component, which we call our "neural content selector", are in the appendix.

---

[8] After the selection steps, the average number of tokens of the transcripts for FD and TMS reduces to 1138.9 and 3252.7 respectively.

[9] We use the maximum sequence length of 1024 (i.e., we truncate the input sequences if they are longer than 1024) for BART-large due to computational constraints.

## 5 Experiments

### 5.1 Setup

We report BLEU (Papineni et al., 2002), ROUGE-1 (R1), ROUGE-2 (R2), and ROUGE-L (RL). We report the average of these four metrics as it generally shows the semantic similarities between generations and references. We will refer to these metrics as generic metrics as they treat each word equally.

As characters are fundamental to TV show plots, we believe the quality of plot summaries also depends on including the right characters. To take this factor into account, we compute several bag of character (BoC) metrics based on the fraction of the overlapping characters between generated and gold standard recaps. Formally, we define the BoC precision to be

$$\frac{|f(\text{generation}) \& f(r)|}{|f(\text{generation})|} \quad (1)$$

where $f$ is a function that extracts the bag of characters from some text, where we perform string matching based on the character names that are automatically extracted during dataset construction (see Sec. 3.1), $\&$ computes the intersection of two bags, $|\cdot|$ returns the size of its inputs, and $r$ is the gold standard recap. Similarly, we define the BoC recall to be

$$\frac{|f(\text{generation}) \& f(r)|}{|f(r)|} \quad (2)$$

Since BoC does not consider relations between characters, we also report bag of character relations (BoR) metrics based on the cooccurrence of character pairs. We assume two characters are related when they appear in the same sentence. After obtaining the character relations from the gold standard recaps and the generations, we compute recall and precision against the recaps following the same approach as BoC. We note that the extracted relations are non-directional, and BoR does not consider frequency of the cooccurrences. We also report the averages of both precisions and recalls from both the BoC and BoR metrics.

More details about hyperparameters are in the appendix.

### 5.2 Results

We report test results for FD and TMS in Table 5. Our findings for the nearest neighbor models are as follows:

1. We find that the nearest neighbor models give strong performance on our dataset. In particular, NNM-r2t shows the best performance in general. This demonstrates that there is still room for improving the ability of our neural models to extract the most useful information from transcripts, suggesting that improved transcript modeling may be a fruitful research direction for these datasets.

2. We observe that NNM-r2r exhibits different strengths when based on different metrics, for example, using ROUGE scores will lead to results favorable to generic metrics.

As for the results involving neural models, our findings are as follows:

1. The neural model shows strong performance in generic semantic matching but it is relatively weak in entity metrics compared to the non-oracle baselines. (see the appendix for more discussion).

2. The hybrid model is better than the neural model in terms of generating character mentions and relations. With the help of the oracle content selector, the hybrid model improves significantly in both semantic matching and entity-related metrics, showing that future research may find improvement by designing better content selectors.

## 6 Analysis

## 7 Effect of Combining FD and TMS

We study the effect of transfer learning using these two resources. When doing so, we use the training and development sets constructed from both resources, and at test time, we evaluate models on the official test splits. We experiment with the oracle hybrid model and report results in Table 6. In general, we find that extra training data helps FD. We hypothesize that this is due to the relatively small size of FD. However, for TMS, training on FD harms performance, which is likely because of the larger training set size for TMS and the differences between the two resources.

### 7.1 Human Evaluation

We conduct human evaluations for three models: NNM-t2t, hybrid model, and hybrid model (oracle). To evaluate two key aspects of SUMMSCREEN, namely events and characters relationships, we ask human annotators two questions. The first is "Do

| | Generic Metrics | | | | | Entity Metrics | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | BLEU | R1 | R2 | RL | avg. | BoC-p | BoC-r | BoR-p | BoR-r | avg. |
| ForeverDreaming | | | | | | | | | | |
| NNM-r2t (oracle, BM25) | 3.4 | 34.3 | 6.6 | 29.6 | 18.5 | 70.5 | 61.9 | 36.4 | 16.1 | 46.2 |
| NNM-r2t (oracle, RG) | 3.9 | 34.8 | 8.5 | 31.5 | 19.7 | **76.7** | 63.3 | **46.5** | 21.3 | 52.0 |
| NNM-r2r (oracle, RG) | **9.9** | **38.8** | **11.5** | **33.9** | **23.5** | 50.6 | 51.4 | 24.6 | 26.8 | 38.4 |
| NNM-r2r (oracle, Entity Metrics) | 5.5 | 31.1 | 6.8 | 27.1 | 17.6 | 58.6 | **79.6** | 26.4 | **43.7** | **52.1** |
| NNM-t2t | 7.9 | 31.3 | 7.8 | 27.4 | 18.6 | 56.5 | 59.2 | 28.2 | 29.4 | 43.3 |
| Neural model | 2.6 | 25.9 | 4.2 | 23.8 | 14.1 | 54.7 | 38.5 | 22.8 | 15.1 | 32.8 |
| Hybrid model | 2.4 | 25.3 | 3.9 | 23.1 | 13.7 | 61.2 | 51.4 | 29.8 | 23.6 | 41.5 |
| Hybrid model (oracle) | 3.0 | 26.4 | 5.0 | 23.3 | 14.4 | 70.0 | 57.8 | 36.9 | 29.1 | 48.5 |
| TVMegaSite | | | | | | | | | | |
| NNM-r2t (oracle, BM25) | 6.7 | 45.0 | 10.2 | 43.0 | 26.2 | 82.5 | **80.4** | 57.7 | 18.1 | 59.7 |
| NNM-r2t (oracle, RG) | **8.5** | 44.1 | 11.7 | 42.4 | 26.7 | 85.2 | 76.8 | **61.2** | 16.9 | 60.0 |
| NNM-r2r (oracle, RG) | 7.9 | **49.0** | 11.6 | **46.9** | **28.9** | 59.2 | 59.0 | 29.5 | 29.9 | 44.4 |
| NNM-r2r (oracle, Entity Metrics) | 4.9 | 42.8 | 8.8 | 40.4 | 24.2 | 60.8 | 81.7 | 26.0 | **37.5** | 51.5 |
| NNM-t2t | 6.2 | 43.2 | 8.6 | 41.4 | 24.9 | 63.2 | 69.3 | 31.8 | 35.3 | 49.9 |
| Neural model | 7.9 | 42.9 | 11.9 | 41.6 | 26.1 | **86.1** | 48.7 | 48.9 | 22.3 | 51.5 |
| Hybrid model | 5.5 | 38.8 | 10.2 | 36.9 | 22.8 | 84.5 | 57.2 | 51.0 | 29.3 | 55.5 |
| Hybrid model (oracle) | 8.9 | 42.1 | **11.9** | 40.9 | 25.9 | 84.0 | 69.5 | 56.4 | 36.8 | **61.7** |

Table 5: Results on the SUMMSCREEN test sets. BLEU, R1, R2, and RL are BLEU and ROUGE scores between model generated and reference recaps. Bo{C,R}-{p,r} are precision and recall for bag of characters and bag of character relations, respectively. The highest numbers for each dataset in each column are in bold.

| | Generic | Entity |
|---|---|---|
| ForeverDreaming | | |
| FD Only | 16.5 | 47.3 |
| TMS + FD | 16.9 | 50.1 |
| TVMegaSite | | |
| TMS Only | 25.9 | 61.7 |
| TMS + FD | 23.2 | 58.0 |

Table 6: Results of the oracle hybrid model comparing training on both datasets (TMS + FD) to training on the in-domain dataset only. The metrics are averaged scores of the generic and entity metrics. Training on both datasets helps for FD but hurts for TMS.

| | Predicates | Character Relation |
|---|---|---|
| NNM-t2t | 1.6±0.8 | 2.1±1.1 |
| Hybrid model | 2.3±0.9 | 2.0±1.0 |
| Hybrid model (oracle) | 2.4±1.0 | 2.4±1.0 |

Table 7: Human evaluation results. We report the average scores and their corresponding standard deviations for questions on predicate match and character relation similarity.

the predicates in the generation match the predicates in the reference?"[10] The second is "When multiple characters are mentioned as being related in some way in the generated recap, are those same characters mentioned as being related in some way in the reference?" We disregard the subjects in the first question because the second question involves evaluating characters and we want the two questions to focus on different aspects to maximize the

efficiency of human annotations. Ratings are on a 1-5 scale with 5 indicating a perfect match. We randomly picked instances from the FD test set. We (the authors) annotated 120 instances in total for each question.

After dropping 2 invalid annotations for the second question (as there may not be multiple characters mentioned), we summarize results in Table 7. While trends for the model performance on character relations are generally similar to our observations in Table 5, the results for predicate match are very different for NNM-t2t. This is likely because the first question is about predicates disregarding the correctness of the participants. We also want to highlight that compared to the oracle hybrid model, the non-oracle one shows competitive performance on predicate match but is less close in terms of generating correct character relations, showing future opportunities for improving this model.

## 7.2 Generation Samples

In Table 8, we show generation samples for the following models: the neural model, the hybrid model, and the oracle hybrid model. The neural model manages to fit most of the character names from the reference into the generation. The generation shares similar topics with the reference, but compared to the hybrid models it lacks specifics. This matches our observations from the automatics metrics where the neural model performs better on the generic metrics but worse on the entity metrics on the non-anonymized datasets. We hypothesize

---

[10] By "predicate" here we mean the part of a sentence or clause containing a verb and stating something about the subject (e.g., "went home" in "John went home").

| Reference | The remains of two witches , one of which is from the Salem witch trials from the 1600s and the other a modern day Wiccan , are discovered in the remains of a burnt out cabin . Booth and Brennan investigate the world of Wicca , including discovering the Wiccan group of which the victim was a part . Hodgins and Angela wind up in jail after some reckless driving and have to work the case from the jail cell . After spending quality time together , they realize they are still in love . Hodgins subsequently proposes to Angela and they are married by the judge who hears their case . |
|---|---|
| Neural model | Booth and Brennan are called to investigate when they are found dead in the death of a young woman who is found in to investigate . Meanwhile , Brennan and Booth are found at the victim 's death of an old friend , but the team must find out to investigate the team up with the case . The team investigates a young man who was killed when they discover that the victim was killed . The victim was not have been retrieve her father , Booth and Angela and Booth 's father 's death . |
| Hybrid model | While the team investigates the death of a 40-year - old woman , who was found buried in a rock quarry . They discover that the woman 's feet were curled after she was buried , and that the bones were de - fleshed prior to her death . Meanwhile , Hodgins and Angela are in jail . Hodgins tells Angela that he 's a witch , but Angela tells Hodgins that she 's not a witch . The team finds out that the victim 's sister , Cheri , was also buried in the quarry . |
| Hybrid model (oracle) | While the team investigates the death of a woman found buried in the woods . They discover that the victim was a Wiccan , and that she may have been a victim of a ritual that was used during the Salem Witch Trials . They also find that the woman was wearing red slippers and that her feet curled up after she was dead . Meanwhile , Hodgins and Angela are in a jail cell , and they are having a hard time adjusting to their new life in the city . The case is complicated by the fact that the body of the woman who was found is a young girl . |

Table 8: Generation samples from ForeverDreaming. The instance is from the TV show "Bones".

that this is caused by the difficulty of modeling long-form text.

In the output of the non-oracle hybrid model, many facts that are not mentioned in the reference are actually from the transcript. For example, "40-year-old woman" and "de-fleshed prior to her death" are in the transcript. Despite containing many specifics, the generation misses a few important details, such as the absence of mentioning main characters involved (i.e., Brennan and Booth). It also has incorrect facts. For example, according to the transcript, there are rocks at the scene, but the model describes the setting as a rock quarry. Compared to the other three models, the generation from the oracle hybrid model is the most faithful, although there are still incorrect facts (e.g., "... and they are having a hard time adjusting to their new life in the city."). The differences between the oracle and non-oracle hybrid model suggest that future research can focus on improving models' capabilities of doing content selection. As both oracle and non-oracle hybrid models suffer from generating incorrect facts, faithfulness in generation is also an important future research direction.

## 8 Conclusion

We construct SUMMSCREEN, which contains pairs of TV show transcripts and recaps. We qualitatively analyze the challenging aspects of our dataset. We propose two entity-centric metrics to evaluate generated summaries with one focusing on character overlap and the other focusing on overlap of charac-ter pairs. Empirically, we benchmark several neural models and nearest neighbor models for characterizing our datasets, finding that an oracle extractive summarizer gives the strongest performance according to the automatic metrics. Human evaluations show that the non-oracle hybrid model is competitive at generating faithful topics. Qualitative analysis shows that the hybrid model can benefit from better content selectors and both oracle and non-oracle hybrid models suffer from generating inaccurate details, highlighting several directions for future research.

## References

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Jean Carletta, Simone Ashby, Sebastien Bourban, Mike Flynn, Mael Guillemot, Thomas Hain, Jaroslav Kadlec, Vasilis Karaiskos, Wessel Kraaij, Melissa Kronenthal, et al. 2005. The AMI meeting corpus: A pre-announcement. In *International workshop on machine learning for multimodal interaction*, pages 28–39. Springer.

Henry Y. Chen, Ethan Zhou, and Jinho D. Choi. 2017.

Robust coreference resolution and entity linking on dialogues: Character identification on TV show transcripts. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 216–225, Vancouver, Canada. Association for Computational Linguistics.

Mingda Chen and Kevin Gimpel. 2021. TVRecap: A dataset for generating stories with character descriptions. *arXiv preprint arXiv:2109.08833*.

Wei-Fan Chen, Shahbaz Syed, Benno Stein, Matthias Hagen, and Martin Potthast. 2020. Abstractive snippet generation. In *Proceedings of The Web Conference 2020*, WWW '20, page 1309–1319, New York, NY, USA. Association for Computing Machinery.

Yu-Hsin Chen and Jinho D. Choi. 2016. Character identification on multiparty conversation: Identifying mentions of characters in TV shows. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 90–100, Los Angeles. Association for Computational Linguistics.

Sangwoo Cho, Franck Dernoncourt, Tim Ganter, Trung Bui, Nedim Lipka, Walter Chang, Hailin Jin, Jonathan Brandt, Hassan Foroosh, and Fei Liu. 2021. StreamHover: Livestream transcript summarization and annotation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6457–6474, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jinho D. Choi and Henry Y. Chen. 2018. SemEval 2018 task 4: Character identification on multiparty dialogues. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 57–64, New Orleans, Louisiana. Association for Computational Linguistics.

Ann Clifton, Sravana Reddy, Yongze Yu, Aasish Pappu, Rezvaneh Rezapour, Hamed Bonab, Maria Eskevich, Gareth Jones, Jussi Karlgren, Ben Carterette, and Rosie Jones. 2020. 100,000 podcasts: A spoken English document corpus. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5903–5917, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 615–621, New Orleans, Louisiana. Association for Computational Linguistics.

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

Lea Frermann, Shay B. Cohen, and Mirella Lapata. 2018. Whodunnit? crime drama as a case for natural language understanding. *Transactions of the Association for Computational Linguistics*, 6:1–15.

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.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. SAMSum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 70–79, Hong Kong, China. Association for Computational Linguistics.

Philip John Gorinski and Mirella Lapata. 2015. Movie script summarization as graph-based scene extraction. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1066–1076, Denver, Colorado. Association for Computational Linguistics.

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.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian Error Linear Units (GELUs). *arXiv preprint arXiv:1606.08415*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, volume 28, pages 1693–1701. Curran Associates, Inc.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Chao-Chun Hsu and Lun-Wei Ku. 2018. SocialNLP 2018 EmotionX challenge overview: Recognizing emotions in dialogues. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*, pages 27–31, Melbourne, Australia. Association for Computational Linguistics.

A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke, and C. Wooters. 2003. The ICSI meeting corpus. In *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03).*, volume 1, pages I–I.

Aditya Joshi, Vaibhav Tripathi, Pushpak Bhattacharyya, and Mark J. Carman. 2016. Harnessing sequence labeling for sarcasm detection in dialogue from TV series 'Friends'. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 146–155, Berlin, Germany. Association for Computational Linguistics.

Anirudh Joshi, Namit Katariya, Xavier Amatriain, and Anitha Kannan. 2020. Dr. summarize: Global summarization of medical dialogue by exploiting local structures. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3755–3763, Online. Association for Computational Linguistics.

Misha Khalman, Yao Zhao, and Mohammad Saleh. 2021. ForumSum: A multi-speaker conversation summarization dataset. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4592–4599, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kundan Krishna, Sopan Khosla, Jeffrey Bigham, and Zachary C. Lipton. 2021. Generating SOAP notes from doctor-patient conversations using modular summarization techniques. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4958–4972, Online. Association for Computational Linguistics.

Faisal Ladhak, Bryan Li, Yaser Al-Onaizan, and Kathleen McKeown. 2020. Exploring content selection in summarization of novel chapters. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5043–5054, Online. Association for Computational Linguistics.

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

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

Peter J Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. 2018. Generating wikipedia by summarizing long sequences. In *International Conference on Learning Representations*.

Kaixin Ma, Tomasz Jurczyk, and Jinho D. Choi. 2018. Challenging reading comprehension on daily conversation: Passage completion on multiparty dialog. 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 2039–2048, New Orleans, Louisiana. Association for Computational Linguistics.

Kaixin Ma, Catherine Xiao, and Jinho D. Choi. 2017. Text-based speaker identification on multiparty dialogues using multi-document convolutional neural networks. In *Proceedings of ACL 2017, Student Research Workshop*, pages 49–55, Vancouver, Canada. Association for Computational Linguistics.

Rada Mihalcea and Hakan Ceylan. 2007. Explorations in automatic book summarization. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 380–389, Prague, Czech Republic. Association for Computational Linguistics.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Pinelopi Papalampidi, Frank Keller, Lea Frermann, and Mirella Lapata. 2020. Screenplay summarization using latent narrative structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1920–1933, Online. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

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

Stephen Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, and M. Gatford. 1995. Okapi at trec-3. In *Overview of the Third Text REtrieval Conference (TREC-3)*, pages 109–126. Gaithersburg, MD: NIST.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015*

Conference on Empirical Methods in Natural Language Processing, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Evan Sandhaus. 2008. *The New York Times Annotated Corpus*. LDC corpora. Linguistic Data Consortium.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Eva Sharma, Chen Li, and Lu Wang. 2019. BIGPATENT: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2204–2213, Florence, Italy. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pages 5998–6008. Curran Associates, Inc.

Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. 2017. TL;DR: Mining Reddit to learn automatic summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, pages 59–63, Copenhagen, Denmark. Association for Computational Linguistics.

Lu Wang and Wang Ling. 2016. Neural network-based abstract generation for opinions and arguments. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 47–57, San Diego, California. Association for Computational Linguistics.

Zhengzhe Yang and Jinho D. Choi. 2019. FriendsQA: Open-domain question answering on TV show transcripts. In *Proceedings of the 20th Annual SIGdial Meeting on Discourse and Dialogue*, pages 188–197, Stockholm, Sweden. Association for Computational Linguistics.

Dian Yu, Kai Sun, Claire Cardie, and Dong Yu. 2020. Dialogue-based relation extraction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4927–4940, Online. Association for Computational Linguistics.

Sayyed M. Zahiri and Jinho D. Choi. 2017. Emotion detection on tv show transcripts with sequence-based convolutional neural networks.

Ming Zhong, Da Yin, Tao Yu, Ahmad Zaidi, Mutethia Mutuma, Rahul Jha, Ahmed Hassan Awadallah, Asli Celikyilmaz, Yang Liu, Xipeng Qiu, and Dragomir Radev. 2021. QMSum: A new benchmark for query-based multi-domain meeting summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5905–5921, Online. Association for Computational Linguistics.

Ethan Zhou and Jinho D. Choi. 2018. They exist! introducing plural mentions to coreference resolution and entity linking. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 24–34, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Chenguang Zhu, Yang Liu, Jie Mei, and Michael Zeng. 2021. MediaSum: A large-scale media interview dataset for dialogue summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5927–5934, Online. Association for Computational Linguistics.

# A  Hyperparameters

We set the maximum sequence length to be 14336 for the encoder and 1024 for the decoder. We use byte-pair encoding (Sennrich et al., 2016) with approximately 10k vocabulary size. We use a 1-layer encoder and a 12-layer decoder with 1024 hidden units unless otherwise specified. We use an effective batch size of 200, and train the models for 50 epochs. During training, we perform early stopping on the development sets based on perplexities. During testing, we use beam search with trigram blocking (Paulus et al., 2018) and a beam size of 5.

For the neural content selector, we use a 3-layer longformer encoder followed by a 2-layer feedforward network with GELU activations (Hendrycks and Gimpel, 2016). We perform early stopping based on F1 scores on the development sets, where the threshold is chosen by averaging over the oracle thresholds for each instance. When selecting content, we use the threshold chosen based on the development set and ensure that no less than 10% of lines for each transcript are selected. The model achieves test performance (F1 scores) of 19.0 on FD, 19.2 on anonymized FD, 41.5 on TMS, and 40.1 on anonymized TMS.

# B  Anonymized SUMMSCREEN

As plots for TV shows are typically about a limited number of characters, models trained on SUMMSCREEN may focus on those characters and their typical behaviors rather than the actual actions taking place in the input transcripts. To eliminate

**Anonymized Transcript:**

[ The apartment ]
ENTITY90 : What color would you like to be ?
ENTITY74 : Well , I 'd like to be green , but you know you always take it .
ENTITY90 : That 's not true . Any color 's fine with me . Yeah , I could be a - a combination of blue and yellow .
ENTITY74 : Blue and yellow make green .
ENTITY90 : Well , then it 's settled .
ENTITY77 : Hi . Ready to go ?
ENTITY90 : Oh , good news , we ordered lunch , so we can all stay here and play Lord of the Rings Risk .
ENTITY99 : ENTITY90 , we said that we would play games with you tonight .
ENTITY90 : Oh , no , we 'll still be playing it tonight , this game can easily take eight hours .
ENTITY77 : Sweetie , you really thought I 'd want to do this ?
ENTITY74 : No .
ENTITY77 : Well , did you tell him that ?
ENTITY74 : Yes .
ENTITY77 : Did you say it out loud with words ?
ENTITY74 : No .
ENTITY77 : I do n't want to spend the whole day playing a board game .
...

**Anonymized Recap:**

ENTITY90 and ENTITY74 are happy playing a board game until ENTITY99 and ENTITY77 say they are tired of doing what the guys want ...

Figure 4: An excerpt from anonymized SUMMSCREEN that corresponds to the instance in the Figure 1 in the main text. Character names are replaced with IDs that are permuted across episodes.

this effect, we create an anonymized version of SUMMSCREEN by replacing character names with random character IDs. We ensure that the IDs of particular characters in different episodes are randomly assigned (i.e., IDs are not consistent across episodes).

Figure 4 shows an example from anonymized SUMMSCREEN. Anonymized question answering datasets have also been created out of similar concerns to those just described (Hermann et al., 2015).

## C  Results for the Anonymized Datasets

In Table 9, it is interesting to observe the performance differences of the nearest neighbor models between the anonymized and non-anonymized datasets. The gaps show that the anonymization does not lead to much difference regarding the similarities between recaps and transcripts, but it makes correlations among recaps and transcripts much weaker especially for those entities.

## D  Effect of Anonymization

We study the effect of anonymization by investigating performance on rare entities. To do so, we first compute entity frequencies for each TV show from the training set, rank the entities by their frequencies, pick the rare entities according to the rank, and evaluate performance for the selected entities. We summarize the results in Table 10. We find that models trained on the anonymized TMS dataset give better performance on rare entities, suggesting that anonymization helps in modeling rare entities. The fact that the two models have the same performance in the "all" setting shows that anonymization also makes the learning of common entities harder, matching our expectations.

## E  Effect of Copy Mechanism

We report results on ForeverDreaming in Table 11 comparing models with and without the copy mechanism. We note that models used in this table use 6-layer decoders with 512 hidden units, so the results are not directly comparable to other results. From the results in Table 11, we find that the copy mechanism helps tremendously on the anonymized dataset, but gives mixed results on the non-anonymized dataset. This is likely due to the fact that for the anonymized dataset, there is not enough training data for the character ID embeddings, and the copy mechanism helps to reduce the required supervision. While there may be better ways of handling the character IDs that may avoid this issue (e.g., sampling IDs from exponential-like distributions rather than uniform distribution), we leave this for future research. However, this benefit does not hold for the non-anonymized dataset as the models are able to exploit more information when learning character name embeddings by having access to the character names.

## F  Effect of Combining FD and TMS

In Table 12, it is interesting to see that the anonymized ForeverDreaming benefits greatly from additional training data, supporting our previous hypothesis that the copy mechanism helps to reduce the amount of required supervision.

| | Generic Metrics | | | | | Entity Metrics | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | BLEU | R1 | R2 | RL | avg. | BoC-p | BoC-r | BoR-p | BoR-r | avg. |
| Anonymized ForeverDreaming | | | | | | | | | | |
| NNM-r2t (oracle, BM25) | 3.5 | 34.5 | 6.8 | 30.0 | 18.7 | 70.4 | 60.4 | 37.5 | 16.7 | 46.2 |
| NNM-r2t (oracle, RG) | 4.0 | **34.7** | 8.5 | 31.4 | 19.7 | **76.8** | **63.4** | **49.1** | 22.6 | **53.0** |
| NNM-r2r (oracle, RG) | **7.9** | 34.3 | **9.1** | 30.1 | **20.4** | 5.4 | 6.3 | 0.2 | 0.1 | 3.0 |
| NNM-t2t | 6.0 | 26.2 | 6.0 | 23.0 | 15.3 | 21.5 | 6.6 | 5.0 | 0.2 | 8.3 |
| Neural model | 2.6 | 28.6 | 4.6 | 25.1 | 15.2 | 65.0 | 57.7 | 27.9 | **30.6** | 45.3 |
| Hybrid model | 2.3 | 23.1 | 3.9 | 20.6 | 12.5 | 12.2 | 2.3 | 0.3 | 0.0 | 3.7 |
| Hybrid model (oracle) | 2.9 | 26.0 | 5.0 | 22.2 | 14.0 | 33.9 | 8.8 | 3.6 | 0.6 | 11.7 |
| Anonymized TVMegaSite | | | | | | | | | | |
| NNM-r2t (oracle, BM25) | 6.9 | **45.0** | 10.2 | **42.9** | 26.2 | 82.6 | **80.5** | 58.9 | 20.7 | 60.7 |
| NNM-r2t (oracle, RG) | **8.7** | 44.1 | **11.7** | 42.3 | **26.7** | 85.3 | 76.7 | **61.8** | 19.3 | 60.8 |
| NNM-r2r (oracle, RG) | 6.0 | 42.8 | 9.3 | 41.1 | 24.8 | 46.3 | 14.7 | 3.8 | 0.6 | 16.3 |
| NNM-t2t | 4.4 | 26.2 | 6.0 | 23.0 | 14.9 | 47.7 | 15.2 | 3.8 | 0.5 | 16.8 |
| Neural model | 7.1 | 41.6 | 11.6 | 40.4 | 25.2 | **86.8** | 53.6 | 32.0 | 15.2 | 46.9 |
| Hybrid model | 6.2 | 37.7 | 9.3 | 36.4 | 22.4 | 82.5 | 62.3 | 47.4 | 30.2 | 55.6 |
| Hybrid model (oracle) | 6.1 | 38.9 | 10.1 | 37.6 | 23.2 | 84.3 | 68.1 | 55.6 | **38.8** | **61.7** |

Table 9: Results on the anonymized SUMMSCREEN test sets. BLEU, R1, R2, and RL are BLEU and ROUGE scores between model generated and reference recaps. Bo{C,R}-{p,r} are precision and recall for *Bag of Characters* and *Bag of Character Relations*, respectively. The highest numbers for each dataset in each column are in bold.

| Fraction | TMS | Anonymized TMS |
|---|---|---|
| All | 61.7 | 61.7 |
| 80% | 19.1 | 25.5 |
| 60% | 11.0 | 17.0 |

Table 10: Average scores of entity metrics computed on various subsets of entities, dropping the most common entities when forming subsets. For example, the "80%" row corresponds to omitting the most frequent 20% of entities in each TV show. Results are based on the oracle hybrid model.

| | Generic | Entity |
|---|---|---|
| ForeverDreaming | | |
| w/o copy mechanism | 12.4 | 29.3 |
| w/ copy mechanism | 12.6 | 27.1 |

Table 11: Comparing models with and without the copy mechanism on ForeverDreaming.

| | Generic | Entity |
|---|---|---|
| Anonymized ForeverDreaming | | |
| Anonymized FD Only | 13.7 | 11.3 |
| Anonymized (TMS + FD) | 17.1 | 52.9 |
| Anonymized TVMegaSite | | |
| Anonymized TMS Only | 23.2 | 61.7 |
| Anonymized (TMS + FD) | 22.7 | 59.8 |

Table 12: Results of the oracle hybrid model comparing training on both datasets (TMS + FD) to training on the in-domain dataset only. The metrics are averaged scores of the generic and entity metrics. Training on both datasets helps for FD but hurts for TMS.