TEXT SUMMARIZATION UNDER LOW SUPERVISION

Chao Zhao

A dissertation submitted to the faculty of the University of North Carolina at Chapel Hill in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Computer Science.

Chapel Hill
2023

Approved by:

Snigdha Chaturvedi

Mohit Bansal

Kathleen McKeown

Colin Raffel

Soumyadip Sengupta

ABSTRACT

Chao Zhao: Text Summarization Under Low Supervision
(Under the direction of Prof. Snigdha Chaturvedi)

Text summarization aims to create a concise and fluent summary that captures the most salient information from a given document(s). However, most summarization methods require large-scale document-summary pairs as the training data, which is laborious to acquire for many domains. This calls for the development of summarization algorithms that can work in a low-supervision setting, which is still a challenging problem. In this dissertation, we address the problem from three perspectives.

We start by improving the summarization methods using external information. Specifically, we focus on the task of product review summarization. We utilize the feature descriptions of the product as external information to better guide the model to identify aspect-related information from reviews and create corresponding summaries.

Besides the use of external information, we also explore the use of external models, and propose a method that enables knowledge transfer from single-document summarization (SDS) to multi-document summarization (MDS). Our approach involves an efficient and effective technique of multiple document reordering, which facilitates both unsupervised and supervised MDS.

In the third part, we present novel approaches to automatically construct high-quality paired training data for summarization. In particular, we introduce two large-scale datasets: DIANA for dialogue summarization and NARRASUM for narrative summarization. We experimentally demonstrate that pre-training on these datasets significantly improves summarization quality.

Finally, given that the primary objective of summarization is to help users better grasp key information and understand the document, we investigate the potential of utilizing automatically constructed summarization datasets to enhance reading comprehension in a zero-shot manner. We propose PARROT, a zero-shot approach that leverages document-summary pairs for reading comprehension. Our results demonstrate that PARROT outperforms previous zero-shot approaches and achieves comparable performance to fully supervised models, showcasing how text summarization can facilitate reading comprehension with minimal supervision.

To my parents.

ACKNOWLEDGEMENTS

I am profoundly grateful to my PhD supervisor, Prof. Snigdha Chaturvedi, whose sagacious guidance and consistent support were indispensable in finishing this dissertation. From crafting meaningful research questions to refining the art of academic writing and presentation, her mentorship was pivotal at every stage of this endeavor. Working alongside her has been a tremendous privilege, and the knowledge gained has been truly invaluable.

I would also like to thank my esteemed thesis committee members: Profs. Mohit Bansal, Kathleen McKeown, Colin Raffel, and Soumyadip Sengupta. They provided invaluable insights during our regular discussions and their suggestions significantly enhanced the quality of this work. I also express my gratitude for the guidance and enlightening discussions with Prof. Marilyn Walker, whose course introduced me to the fascinating field of natural language generation.

My gratitude extends to all my internship supervisors. At Baidu Knowledge Graph, Min Zhao and Huapeng Qin offered me fresh perspectives on symbolic knowledge in NLP. At AWS AI, Miguel Ballesteros and Muthu Kumar Chandrasekaran offered profound insight into text summarization. At Tencent AI, Wenlin Yao, Dian Yu, Kaiqiang Song, Dong Yu, and Jianshu Chen from Tencent AI ignited my enthusiasm for building robust dialogue understanding systems. At Alexa AI, Seokhwan Kim, Spandana Gella, Yang Liu, and Dilek Hakkani-Tur guided me through real-world industry-level challenges in dialogue understanding and generation.

I would like to thank my lab peers and collaborators: Faeze Brahman, Tenghao Huang, Somnath Basu Roy Chowdhury, Anvesh Rao Vijjini, Anneliese Brei, and Haoyuan Li. Their insightful discussions, generous support, and close collaboration significantly enriched this work.

Finally, I want to express my deepest gratitude to my parents for their constant support and encouragement throughout the past five years.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
BERT	Bidirectional Encoder Representations from Transformers
CNN	Convolutional Neural Network
DUC	Document Understanding Conference
GPT	Generative Pre-trained Transformer
ILP	Integer Linear Programming
IR	Information Retrieval
LLM	Large Language Model
LSTM	Long Short-Term Memory
MLM	Masked Language Modeling
MMR	Maximal Marginal Relevance
MDS	Multiple-Document Summarization
NER	Named Entity Recognition
NLG	Natural Language Generation
NLP	Natural Language Processing
NLU	Natural Language Understanding
PLM	Pre-trained Language Model
PMR	Perfect Match Ratio
RNN	Recurrent Neural Network
ROUGE	Recall-Oriented Understudy for Gisting Evaluation
RST	Rhetorical Structure Theory
SDS	Single-Document Summarization
SRL	Semantic Role Labeling
TF-IDF	Term Frequency-Inverse Document Frequency

CHAPTER 1: INTRODUCTION

Text summarization is a core problem of Artificial Intelligence (AI) and Natural Language Processing (NLP). It aims to create a concise and fluent summary of the most salient information in a given document(s). A good summary can help readers quickly understand the key information of the given document, which is especially helpful when reading the entire document is laborious and time-consuming. Given the large and growing number of documents these days which are overwhelming for users to read and process, the need for automatic text summarization is more urgent than ever.

Text summarization can be classified into different types based on different perspectives. Based on the number of the given documents, it can be classified into *single document summarization (SDS)* or *multi-document summarization (MDS)* (McKeown and Radev, 1995; Mani and Bloedorn, 1997). While SDS aims to summarize only one given document, MDS requires creating a summary for a cluster of documents on a similar topic. Based on whether the natural language generation is required, it can be classified into *extractive summarization* or *abstractive summarization* (Jing and McKeown, 2000; Cohn and Lapata, 2008). Extractive summaries are created by extracting and concatenating the salient text units (paragraphs, sentences, sub-sentences, etc) from the given document, while abstractive summaries are created by generating new text to express the salient information of the document. Summarization methods can also be categorized based on the domains they would be applied to, such as news (Nallapati et al., 2016), scientific papers (Cohan et al., 2018), online reviews (Hu and Liu, 2004), dialogues (Gliwa et al., 2019), meeting transcripts (Carletta et al., 2005), narratives (Lehnert, 1981), and so on.

Traditional summarization methods use rule- or graph-based methods to identify the salient information from the documents (Luhn, 1958; Nenkova and Vanderwende, 2005; Erkan and Radev, 2004; Mihalcea and Tarau, 2004). These methods are based on heuristics and therefore are not applicable to every domain. Later works use data-driven approaches to learn from human annotations and achieve better performance (Hovy and Lin, 1997; Cheng and Lapata, 2016; Liu and Lapata, 2019*b*). However, such methods require large-scale document-summary pairs as the training data, which is laborious to obtain for every domain. For example, in meeting summarization, the largest dataset contains only a few thousand (document, summary) pairs (Zhong

et al., 2021). In opinion summarization, it is difficult for annotators to read all reviews of a product (which may exceed 1,000) before writing a summary (Bražinskas, Lapata and Titov, 2020).

The lack of training data brings the need to develop text summarization algorithms under a low-supervision setting. However, this is still a challenging problem for the following reasons. **First**, as we mentioned before, many methods are based on heuristics such as position and frequency information, which cannot be generalized to all summarization problems. For example, opinion summarization focuses on the aspect and sentiment information in the opinions (Hu and Liu, 2004). Meeting summarization focuses on the discussions and essential points of the meeting agenda (Riedhammer, Favre and Hakkani-Tür, 2010). Heuristics-based summarization approaches usually cannot achieve satisfactory performance on these tasks. **Second**, summarization requires a deep understanding of the given documents, which can sometimes go beyond just text saliency understanding. For example, in dialogue understanding, the model needs to understand the complex dialogue discourse structures and speaker relationships (Chen and Yang, 2021*b*). In narrative understanding, the model needs to understand the causal relationships between events and the desired goals of characters (Lehnert, 1981). **Third**, different summarization tasks may have different formats of the input, making it a gap when transferring the model from data-rich tasks to data-deficient tasks. For example, SDS has more training resources compared with MDS. However, in SDS the input is a single document, but in MDS, the input is a cluster of multiple documents. Transferring knowledge from SDS to MDS is a difficult problem.

1.1 Contributions

In this dissertation, **we develop approaches to low-supervision summarization by incorporating guidance from external data sources.** External data sources are those which are absent from the document and the summary but are related to the goal of text summarization. These related data sources and supervision signals derived from them will help us narrow down the space for the summarization model in a beneficial way. More concretely, we explore three strategies:

1. Leveraging external data sources which are saliency-intensive;

2. Transferring knowledge from data-rich tasks to data-deficient tasks;

3. Constructing large-scale paired pre-training data without laborious human annotation;

We start by leveraging external data sources that exhibit high saliency. Specifically, we focus on the task of review summarization (Kim et al., 2011; Ding and Jiang, 2015), which aims to condense online product reviews into concise summaries that capture various aspects (components, attributes, or properties) of the products (Hu and Liu, 2004). One challenge in this task is to identify the aspect-related content within the reviews. Supervised approaches are not feasible due to variations in aspects across different product domains, making it difficult to transfer from one domain to another (He et al., 2017). To address this, we utilize the feature descriptions of the product as external information. Feature descriptions often discuss the various aspects of the product and therefore serve as a great resource to facilitate aspect identification. To leverage this resource, we propose ASPMEM, a generative method that can store aspect-related knowledge from the external information and judge the relevance of review sentences to the product aspects more precisely. This enhancement enables our summarizer to better extract the aspect-related review sentences to compose a more informative summary. Then, we combine the relevance with the sentiment strength to determine the salience of an opinion, and extract a subset of salient opinions to create the final summary. By formalizing the subset selection process as an Integer Linear Programming (ILP) problem, the resulting summary maximizes the collective salience scores of the selected sentences while minimizing information redundancy. Our experiments show that ASPMEM outperforms the state-of-the-art methods of review summarization without human supervision. We also find that the content in feature descriptions is more objective than that in customer reviews, making it a better source to analyze the aspect relevancy than the reviews themselves.

For some tasks, there might be no available external information. However, there might be similar tasks with abundant training data. For example, the task of single-document summarization (SDS) has more paired training data compared with the task of multi-document summarization (MDS) (Nallapati et al., 2016; Fabbri et al., 2019), making it promising to employ resources from SDS to improve MDS via transfer learning. One common approach in this direction is to make the input of MDS similar to that of SDS, and then apply an SDS model to generate the summary. Previous works have achieved this by concatenating multiple documents into a single meta-document (Cao et al., 2017; Liu et al., 2018; Lebanoff, Song and Liu, 2018; Fabbri et al., 2019). However, due to the conventions of news writing, salient information often appears at the beginning of a news article (Hong and Nenkova, 2014; Hicks et al., 2016). As a result, many summarization systems pay more attention to the beginning of the document (Kedzie, McKeown and Daumé III, 2018; Zhong et al., 2019). Therefore, in MDS, it is important to consider the order in which the documents are concatenated to form the

meta-document before applying the summarization model. Specifically, we argue that the various documents in the input are not equally important. Some documents contain more salient or detailed information and are more important. Therefore, it would be beneficial to reorder the documents such that the important ones are in the front of the meta-document and it becomes easier for the summarization model to learn the salient content. Based on this observation, we propose an economical yet effective approach to reordering the documents according to their relative importance before applying a summarization model, which can benefit summarization in both unsupervised and supervised settings. We evaluate the effectiveness of our approach on Multi-News and DUC-2004. Results show that our simple reordering approach significantly outperforms the state-of-the-art methods with more complex model architectures. We also observe that this approach brings more performance gain with the increase in the number of input documents.

In addition to leveraging external information or training data, we also propose methods to automatically construct high-quality paired pre-training data for summarization. In particular, we construct a dialogue narration dataset called DIANA. This dataset is automatically constructed by collecting and pairing subtitles and plot descriptions from movies and TV shows. We consider subtitles and plot descriptions of the same movie or TV episode as dialogues and the corresponding narratives. To create the dataset, we split both the subtitle and synopsis into smaller segments and align the related segments from each part into shorter (dialogue, narrative) pairs based on the text similarity and global optimal alignment. We further conduct quality control to filter out pairs where the dialogue and the narrative are irrelevant. Finally, we obtain 243K (dialogue, narrative) pairs as the final DIANA dataset. Our observation indicates that the narrative information can provide valuable knowledge for dialogue summarization and comprehension. It includes significant events occurring within the dialogue, visual and auditory details present in the dialogue, implicit information requiring deeper inference, causal relationships between events, and more. Building upon this dataset, we propose a novel pre-training approach that involves narrating the key information from a dialogue input. This pre-training approach helps the model learn diverse lexical, syntactic, and semantic aspects of dialogues, and enhances its ability to infer contextual information beyond the literal meaning. We experimentally show that pre-training a model on DIANA improves its capacity to comprehend and summarize dialogues. The results indicate that our narrative-guided generative pre-training objective is more effective than the de-noising objective and the discriminative objective. We also show that DIANA is a more helpful resource for dialogue comprehension and summarization compared with other non-dialogue summarization datasets.

Similarly, we also automatically construct a large-scale dataset for narrative summarization, called NARRASUM. Specifically, we first collect narratives from plot descriptions of movies or TV episodes through online resources. After data collection, we build an align-and-verify pipeline to automatically align plot descriptions of the same movie or TV episodes from different sources. Finally, we construct document-summary pairs by treating the long plot description as the document to be summarized and the shorter one (of the same movie or TV episode) as the corresponding summary. After filtering out low-quality document-summary pairs, we obtain around 122K narrative document-summary pairs in English as the final dataset. We observe that compared with other summarization datasets, the narratives in NARRASUM are of diverse genres, and the summaries are more abstractive and of varying lengths. Furthermore, rather than focusing on a particular part of the document (as in news summarization datasets), the summaries in NARRASUM are designed to cover the entire narratives. It brings new challenges to current summarization methods. We investigate the performance of several strong baselines and state-of-the-art summarization models on NARRASUM. Results show that models trained on NARRASUM outperform the baseline approaches on all measures by a large margin, indicating that NARRASUM can provide a strong supervision signal for identifying the salient information and creating the summary accordingly. However, there is a large gap between human and machine performance in various dimensions, demonstrating that narrative summarization is a challenging task.

Finally, given that the primary objective of summarization is to help users better grasp key information and understand the document (McKeown et al., 2005), we investigate the potential of utilizing automatically constructed summarization datasets to enhance machine reading comprehension in a zero-shot manner. In particular, we tackle the challenging problem of narrative comprehension, which entails understanding complex plot structures and character interactions (Kočiský et al., 2018). Our idea is to leverage parallel reading: reading a document-summary pair of narratives that convey the same story but differ in various aspects of story-telling style (Genette, 1983). We leverage this parallelism by asking questions based on the summary and encouraging the model to answer them based on the document. By exposing the model to narrative variations of the same story, we discourage its reliance on text-matching and enhance its ability to comprehend paraphrases, integrate information from long contexts, perform multi-hop reasoning, deduce implicit information, and ultimately understand the underlying meaning within the narrative. With this idea in mind, we propose PARROT, a zero-shot approach for narrative reading comprehension that leverages document-summary pairs. It selectively masks important narrative elements within the summary, and then

pre-trains the model to predict these masked elements by reading the original document. To encourage PARROT to learn about a wide array of narrative elements, we mask a diverse set of elements covering characters, events, time, place, environments, and more. Lastly, to enable PARROT to perform narrative reading comprehension in a zero-shot manner, we narrow the disparity between the pre-training task of span prediction and the downstream task of reading comprehension by aligning their data formats. To support the training of PARROT, we also automatically collect a large-scale narrative summarization dataset called NARRASUM. This dataset comprises 122K document-summary instances extracted from plot descriptions of movies and TV episodes. Through our experiments, we demonstrate that PARROT, pre-trained on NARRASUM, outperforms previous zero-shot approaches and achieves comparable performance to fully supervised models on narrative comprehension tasks. These results showcase how text summarization can facilitate machine reading comprehension with minimal supervision.

1.2 Chapter Organization

The rest of the dissertation is organized as follows. Chapter 2 provides an overview of the existing literature on text summarization, including single and multi-document summarization, as well as extractive and abstractive summarization. In Chapter 3, we focus on leveraging external information for product review summarization. In Chapter 4, we present our research on introducing document reordering to enhance knowledge transfer from single-document summarization to multi-document summarization. Moving on to Chapter 5 and Chapter 6, we introduce DIANA and NARRASUM, two automatically constructed datasets designed for dialogue summarization and narrative summarization. In Chapter 7, we apply summarization resources to facilitate zero-shot narrative reading comprehension. Lastly, Chapter 8 summarizes the entire dissertation and discusses possible directions for future research.

CHAPTER 2: RELATED WORKS

In this chapter, we introduce the related works in text summarization. First, we provide a summary of existing extractive and abstractive summarization methods for both single-document summarization (SDS) and multi-document summarization (MDS). Next, we introduce the studies for summarization under low supervision, which is highly related to our work.

2.1 Single Document Summarization

2.1.1 Extractive Summarization

Extractive summarization has a long history in summarization. Compared with abstractive summarization, extractive summarization does not require generating new sentences, and therefore the approach is easier and controllable. It also makes the extracted summary more faithful and reliable. The key challenge of extractive summarization is to 1) identify the salient information from the given document, and 2) organize the salient information to compose a summary.

Common approaches identify the salient information at the word level. Early works use heuristics such as position (Luhn, 1958; Edmundson, 1969) or frequency information (Nenkova and Vanderwende, 2005) to select important words, and then select sentences which include these words to compose the summary. The motivation is that these words contain important information and are therefore more likely to be included in the summary. Later works applied more carefully designed heuristics such as TF-IDF (Filatova and Hatzivassiloglou, 2004; Galley, 2006) or latent semantics (Gong and Liu, 2001; Yeh et al., 2005). Similar approaches can also be applied at the phrase level such as n-grams, syntactic subtrees, semantic frames, and named entities (Gillick and Favre, 2009).

Another important line of research uses graph-based approaches to identify salient information, which can go beyond word-level saliency identification and consider the document as a whole (e.g., LexRank (Erkan and Radev, 2004) and TextRak (Mihalcea and Tarau, 2004)). The basic idea is to convert the document into a graph, in which the nodes represent sentences and edge weights represent the similarity between sentences.

Then a graph-based ranking algorithm (e.g., eigenvector centrality or PageRank (Brin and Page, 1998)) is applied to determine the saliency of sentences.

The above approaches use heuristics and do not require any human annotation. Taking advantage of the success of supervised machine learning, a summarization model can also be learned in a supervised manner. That is, given a training corpus where the important sentences in the document are manually annotated, the sentence selection in summarization can be regarded as a binary classification problem (Kupiec, Pedersen and Chen, 1995) or a sequential labeling problem (Conroy and O'leary, 2001; Shen et al., 2007). The input of the model is a sentence (with or without context), and the output is a binary label to indicate whether this sentence should be included in the final summary. Traditional approaches use statistical machine learning. The basic learning schema is to extract summarization-related features from sentences and then train a classifier based on these features (Hovy and Lin, 1997; Mani and Bloedorn, 1998). Features can include heuristic-based importance indicators such as sentence length and sentence position. Sentence-level features can also be aggregated from word-level features (e.g,. by averaging word-level TF-IDF scores or checking whether an important word is included in the sentence). They can also include lexical and semantic features such as n-grams (Hakkani-Tur and Tur, 2007), part-of-speech tags (Fuentes, Alfonseca and Rodríguez, 2007), named entities (Fuentes, Alfonseca and Rodríguez, 2007), syntactic features (Pollock and Zamora, 1975), and discourse features (Louis, Joshi and Nenkova, 2010).

After that, neural based approaches (Collobert et al., 2011; Mikolov et al., 2013) have shown a better capability of understanding text and achieving better performance than traditional approaches on many NLP tasks, including text summarization. Similar to the statistical approaches, neural approaches also rely on a classifier to assign a saliency score for each sentence. The difference is that instead of extracting manually designed features, neural approaches directly encode text into a distributed semantic space, in which sentences are represented as dense vectors. An effective neural approach can learn good representations of sentences, which is able to distinguish the important and unimportant sentences in the vector space (Bengio, Courville and Vincent, 2013). Sentence representations are obtained by first mapping tokens in the sentence as word embeddings (vectors) (Mikolov et al., 2013; Pennington, Socher and Manning, 2014) and then using a sentence encoder to aggregate the information from the word embeddings. Some early neural approaches use bag-of-words (Mikolov et al., 2013) and convolutional neural networks (CNNs) (LeCun et al., 1989) as encoders for summarization (Kågebäck et al., 2014; Yin and Pei, 2015; Cao et al., 2015; Kedzie, McKeown and Daumé III, 2018). Then sequential encoders such as Long Short-Term Memory

(LSTM) network (Hochreiter and Schmidhuber, 1997), Gated Recurrent Unit (GRU) (Cho et al., 2014), and Transformer Network (Vaswani et al., 2017) become more popular in summarization because they are better fits for sequential textual data (Cheng and Lapata, 2016; Nallapati, Zhai and Zhou, 2017; Zhou et al., 2018; Zhong et al., 2019). Besides that, some works also represent the document as a graph and then apply a graph encoder (Kipf and Welling, 2017; Velickovic et al., 2018) to obtain sentence representations (Yasunaga et al., 2017; Xu et al., 2020; Wang et al., 2020; Jia et al., 2020). These methods can build better connections between sentences, especially long-distance dependencies.

Recently, pre-trained language models such as BERT (Devlin et al., 2019) have been proved as more powerful neural approaches in various tasks. These models are pre-trained using a large corpus and self-supervised objectives, making them achieve a better capacity for language understanding and therefore easier to adapt to downstream tasks including summarization (Liu and Lapata, 2019*b*; Wang et al., 2019; Zhong et al., 2020; Narayan et al., 2020).

2.1.2 Abstractive Summarization

Compared with extractive summarization, abstractive summarization is more challenging because it requires not only identifying the salient information from the document but also expressing the salient information by generating new text. Therefore it requires both natural language understanding and generation.

Early approaches did not really generate the summary from scratch. The summaries are usually created by sentence compression (removing unimportant information from a sentence) (Jing and McKeown, 1999; Jing, 2000), sentence revision (substituting some part of the sentence with a more proper one) (Mani, Gates and Bloedorn, 1999; Otterbacher, Radev and Luo, 2002; Nenkova, 2008), and sentence fusion (combining multiple sentences as one concise sentence) (Jing and McKeown, 2000; Barzilay and McKeown, 2005; Filippova and Strube, 2008). When a generation module is indeed involved, a typical approach usually applies a two-step pipeline which includes a content selection step and a surface realization step (Kan and McKeown, 2002; Wang and Cardie, 2013). In content selection, a model is used to identify the salient words/phrases, which is similar to extractive summarization. In surface realization, the selected content will be used to generate a summary. Inspired by the data-driven approaches of statistical machine translation, there are also works that train a statistical abstractive summarization model using the (document, summary) pairs (Banko, Mittal and Witbrock, 2000; Wubben, van den Bosch and Krahmer, 2012).

The development of neural approaches, especially sequence-to-sequence models, brought new solutions to abstractive summarization. Rush, Chopra and Weston (2015) first propose to use an attention-based sequence-to-sequence model to generate sentence-level abstractive summaries. Later works apply more advanced seq-2-seq architectures such as Convolutional Neural Networks (CNN) (Chopra, Auli and Rush, 2016), Recurrent Neural Networks (RNN) (Hu, Chen and Zhu, 2015; Nallapati et al., 2016), and Transformers (Gehrmann, Deng and Rush, 2018). These models are further enhanced by considering the latent structure information (Li et al., 2017), applying copy mechanism (Gu et al., 2016; See, Liu and Manning, 2017; Song, Zhao and Liu, 2018), encouraging coverage and diversity (Chen et al., 2016; See, Liu and Manning, 2017), and incorporating soft templates (Cao et al., 2018). Similar to extractive summarization, reinforcement learning (Pasunuru and Bansal, 2018; Li et al., 2019) and graph structure (Tan, Wan and Xiao, 2017) are also widely explored in abstractive summarization. Pre-trained seq-2-seq models such as BART (Lewis et al., 2020) and T5 (Raffel et al., 2020) show better capability when finetuning on summarization datasets. There are also summarization-specific pre-trained models with summarization-related training objectives such as gap sentences generation (PEGASUS (Zhang et al., 2020*a*)), sentence reordering (Zou et al., 2020), and lead sentence prediction (Zhu et al., 2021).

2.2 Multi Document Summarization

Compared with SDS, MDS (McKeown and Radev, 1995) requires creating a summary for multiple input documents, which brings new challenges to this task. First, the model needs to understand salient information by considering all documents. Second, the model needs to carefully avoid repeated or even contradictory information from multiple documents. Similar to the previous section, we will first introduce the extractive approaches and then the abstractive approaches.

2.2.1 Extractive Summarization

In SDS, we use word-level or sentence-level features to determine sentence saliency. These approaches can be directly applied to MDS when we ignore the document assignment information of words or sentences. For example, many graph-based approaches can work on both SDS and MDS tasks (McKeown et al., 1999; Radev, Jing and Budzikowska, 2000; Radev et al., 2004; Wan and Yang, 2008; Zheng et al., 2019). Moreover,

when applied to MDS, graph-based approaches are flexible enough to build connections between documents by modifying the graph structure (Li et al., 2020; Wang et al., 2020; Pasunuru et al., 2021).

One of the core problems in extractive MDS is to avoid repetition. There are two common strategies. The first is to explicitly cluster sentences into topics and then summarize each cluster, respectively (Radev et al., 2004; Wan and Yang, 2008; Zhang et al., 2015; Ernst et al., 2022). The second is to regard MDS as a constraint optimization problem, where an ideal summary wants to achieve maximal informativeness and minimally redundancy under a pre-specified length (McDonald, 2007). It can be implemented greedily by adding sentences to the summary and requiring the new sentences to have minimal similarity to previously selected sentences. Some examples include Maximal Marginal Relevance (MMR) (Carbonell and Goldstein, 1998; Goldstein et al., 2000) and pivoted QR (Conroy, Schlesinger and O'Leary, 2006). It can also be implemented by greedily removing sentences (or other units such as syntactic constituent) from summaries until the summary has maximal similarity with the document (Marcu, 1999; Zhou, Ticrea and Hovy, 2004; Zajic et al., 2006). These solutions are further optimized by employing either better greedy algorithms (e.g., dynamic programming (McDonald, 2007) and submodular function maximization (Lin and Bilmes, 2010)) or global inference (e.g., integer linear program (McDonald, 2007; Gillick and Favre, 2009)).

2.2.2 Abstractive Summarization

Similar to extractive MDS, a popular approach for abstractive MDS is also to reformulate this problem as an SDS problem by concatenating all documents into a single *meta-document* and then using an abstractive SDS model to summarize it (Cao et al., 2017; Liu et al., 2018; Lebanoff, Song and Liu, 2018; Fabbri et al., 2019; Xiao et al., 2022). The effectiveness of this simple approach has been demonstrated in multiple datasets.

More advanced approaches, based on encoder-decoder model, improve the model in two directions, from the perspective of either the encoder or the decoder. From the encoder perspective, recent work has presented better modeling of cross-document interactions during encoding (Liu and Lapata, 2019*a*; Jin, Wang and Wan, 2020; Li et al., 2020; Pasunuru et al., 2021). For example, Liu and Lapata (2019*a*) represent a hierarchical attention encoder to encode both the paragraph-level contextual information and the interaction across multiple paragraphs. Jin, Wang and Wan (2020) add the word-level granularity as another layer of the hierarchy, and then employ cross-attention mechanisms to enable interaction between representations with different granularity.

From the decoder perspective, neural approaches naturally provide more flexibility in fusing information from multiple sources (Wang and Ling, 2016; Chu and Liu, 2019; Jin and Wan, 2020; Ernst et al., 2022). For example, Chu and Liu (2019) use a neural encoder to encode documents as hidden representations and take the mean of these representations as the input for the decoder. Jin and Wan (2020) employ a decoding controller to aggregate multiple decoders' outputs for multiple input documents.

Avoiding repetition is also a central topic in abstractive MDS. Similar to the sentence fusion in abstractive SDS, repetition in MDS can be avoided by reformulating the wording of the summary and merging salient information from multiple sentences as one concise summary (Barzilay, McKeown and Elhadad, 1999; Filippova, 2010). Similar to extractive MDS, repetition in abstractive MDS can be avoided by either sentence clustering (Nayeem, Fuad and Chali, 2018; Ernst et al., 2022) or constraint optimization (Banerjee, Mitra and Sugiyama, 2015; Nayeem, Fuad and Chali, 2018).

2.3 Summarization Under Low Supervision

Supervised summarization approaches can achieve better performance compared with unsupervised approaches, especially in domains where heuristics may not be appropriate to identify the salient information. However, supervised approaches rely heavily on availability of large training sets, which might not be easy to obtain for all domains. In fact, most of the existing large-scale datasets in summarization are limited to the domains of news and scientific articles.

To address the lack of training data, some works manually collect summaries, which can be laborious and time-consuming, especially when the document is long or the number of documents is large (in case of MDS). These challenges can limit the size of the final dataset and prevent it from being sufficient for model training.

There are several mainstream approaches to address the summarization problem under low supervision. The first approach is to automatically build synthetic/pseudo (document, summary) pairs as the training data (Zhang et al., 2020*a*; Amplayo, Angelidis and Lapata, 2021; Magooda and Litman, 2020; Parida and Motlicek, 2019; Zhu et al., 2021). For example, PEGASUS (Zhang et al., 2020*a*) selects and masks salient sentences according to the ROUGE score between the selected sentences and the rest of the document, and then pre-trains a model to recover the masked sentences. In opinion summarization, Amplayo, Angelidis and Lapata (2021) sample a review as the pseudo summary and then sample a set of similar reviews as the input documents to build the synthetic training data.

The second approach is to use data augmentation to enlarge a small training set (Chen and Yang, 2021*a*; Moro and Ragazzi, 2022; Feng, Feng, Qin and Geng, 2021; Liu, Zou, Zhang, Chen, Ding, Yuan and Wang, 2021). For example, Chen and Yang (2021*a*) add perturbations (e.g., addition, deletion, rotation, and rewriting) to existing datasets to reduce the need for large training data. Moro and Ragazzi (2022); Liu, Zou, Zhang, Chen, Ding, Yuan and Wang (2021) segment the documents and summaries into chunks and align them into small pairs.

Besides data synthesis and augmentation, transfer learning (Pan and Yang, 2009) is also commonly applied in low supervision scenarios. A straightforward approach is domain/data transfer. The motivation is that models pre-trained on data-rich domains (e.g., news) can benefit data-deficient domains (Zhang, Tan and Wan, 2018; Magooda and Litman, 2020; Yu, Liu and Fung, 2021; Fabbri et al., 2021; Bai, Gao and Huang, 2021). For example, Zhang, Tan and Wan (2018) adapt the neural model trained on the SDS data to the MDS task. Yu, Liu and Fung (2021) propose several adaptive pre-training strategies such as source domain pre-training, domain-adaptive pre-training, and task-adaptive pre-training. Besides domain/data transfer, summarization models can also benefit from task transfer, where other tasks may provide useful supervision to facilitate the summarization task (Cao et al., 2017; Goodwin, Savery and Demner-Fushman, 2020; Bi, Li and Yang, 2021; Fu et al., 2021).

Lastly, prompt learning (Liu, Yuan, Fu, Jiang, Hayashi and Neubig, 2021) enables the creation of summaries using large-scale pre-trained language models and summarization-specific prompts or instructions, such as "TL;DR" or "create a short summary for the following document". The capability of following summarization instructions was initially observed in smaller models like GPT-2 (Radford et al., 2019). Then, by employing larger generative models such as GPT-3 (Brown et al., 2020) and ChatGPT, the quality of the generated summaries has significantly improved compared to smaller models (Goyal, Li and Durrett, 2022; Zhang et al., 2023; Yang et al., 2023). These studies further demonstrate that larger models can achieve comparable performance to fine-tuning approaches and crowd-sourced writers in terms of Rouge scores and human-evaluated criteria.

There are some other approaches to address the low supervision issue. For example, meta-learning (Finn, Abbeel and Levine, 2017) can make the summarization model able to adapt to a new domain with small training data through weight initialization (Chen and Shuai, 2021; Huh and Ko, 2022). Auto-encoders can be trained in an unsupervised way and create summaries from the document representation during inference (Chu and Liu, 2019; Basu Roy Chowdhury, Zhao and Chaturvedi, 2022).

In this dissertation, we address the low supervision challenge from three perspectives. We start by leveraging saliency-intensive external data, which can provide supervision signals to identify the salient information. We then propose a document-reordering method for transferring knowledge from SDS to MDS. Lastly, we propose a method to automatically construct high-quality paired pre-training data for summarization.

CHAPTER 3: LEVERAGING EXTERNAL INFORMATION FOR WEAKLY-SUPERVISED OPINION SUMMARIZATION

In this chapter, we begin by exploring the strategy of leveraging saliency-intensive external data sources to improve the quality of summarization. While these external data sources do not directly provide supervision signals for summarization, they can better guide the model to identify salient information from the document.

3.1 Background

We choose the problem of extractive opinion summarization. The goal of this task is to take a collection of reviews of the target product (e.g., a television) as input and selects a subset of review excerpts as a summary. It is especially helpful when the large and growing number of such opinions becomes overwhelming for users to read and process (Kim et al., 2011; Ding and Jiang, 2015). The last two boxes of Figure 3.1 show an example of user reviews of a television and a corresponding extractive summary.

This example illustrates that opinion summarization differs from the more general task of multi-document summarization (Lin and Hovy, 2002) in two major ways. First, while general summarization aims to retain the most important content, opinion summarization needs to cover a range of popular opinions and reflect their diversity (Di Fabbrizio, Stent and Gaizauskas, 2014). Second, opinion summary is more centered on the various *aspects* (i.e., components, attributes, or properties) of the target product, and their corresponding sentiment polarities (Liu, 2015). For example, highlighted sentences in Review 3 of Figure 3.1 express reviewer's negative opinions about the aspects of SOUND and IMAGE. To reflect these differences, Hu and Liu (2004) introduced a three-step pipeline to create an opinion summary by 1) mining product-related aspects and identifying sentences related to those aspects; 2) analyzing the sentiment of the identified sentences; and 3) summarizing the results. Each of these three tasks has often been addressed using supervised methods. Despite the fairly high performance, these methods require the corresponding human-annotated data.

Previous works addressed these problems using pure unsupervised methods. Still, they found it challenging to detect the aspect-related segments of reviews (e.g., those highlighted in Figure 3.1) with both high precision and recall (He et al., 2017). A better solution is to utilize knowledge sourced from existing

Feature descriptions:
- ENHANCED QUALITY : With the X1 Extreme Processor enjoy controlled contrast & wide range of brightness
- BEYOND HIGH DEFINITION : 4K HDTV picture offers stunning clarity & high dynamic range color & detail.
- PREMIUM DISPLAY : Enjoy vibrant colors with TRILUMINOS & clear on-screen action with X-Motion Clarity.
- VOICE COMPATIBILITY : 55in tv is compatible with Amazon Alexa & Google Home to change channels & more.

Review 1: Set up was extremely easy and the remote is simple to use. Simply plug it in and tune to a channel. It gets 4 stars because I don't think its worth the price.
Review 2: The color and definition are excellent. We wanted a small TV for our kitchen counter...and it fit the bill, it seemed.
Review 3: I have owned this TV for 10 months and am looking to replace it. The sound is TERRIBLE. The picture quality is also very rapidly decreasing.
Review n: ...

Summary: Set up was extremely easy and the remote is simple to use. The color and definition are excellent . It's great for casual TV watching. The sound is TERRIBLE. The picture quality is also very rapidly decreasing .

Figure 3.1: An example of the extractive summary from multiple reviews. A review may express opinions about multiple aspects of the target product. These are shown in the figure as highlighted texts in different colors.

external information about the target product i.e., the information beyond the customers' reviews. For example, on Amazon's product webpage, we can obtain not only customer reviews but also product-related information, such as the overall description, the feature descriptions (The top of Figure 3.1 gives an example), and attributes tables. These external information sources widely exist on e-commerce websites and are easily accessible. More importantly, they are closely related to the aspects of products and therefore are great resources to facilitate the aspect identification task. Automatically learning aspects from such external sources can reduce the risk that human-assigned aspects may be biased, unrepresentative, or not have the desired granularity. Meanwhile, it makes the model easy to adapt to different product categories. Here we use the feature descriptions of products as the information source, and leave other sources for future work.

In this chapter, we propose a generative approach that relies on the aspect-aware memory (ASPMEM) to better leverage this knowledge during aspect identification and opinion summarization. ASPMEM, which is inspired by Memory Networks (Weston, Chopra and Bordes, 2015), is an array of memory cells to store aspect-related knowledge obtained from external information. These memory cells cooperate with the model throughout learning, and judge the relevance of review sentences to the product aspects. Then the relevance is combined with the sentiment strength to determine the salience of an opinion. Finally, we extract a subset of salient opinions to create the final summary. By formalizing the subset selection process as an Integer Linear Programming (ILP) problem, the resulting summary maximizes the collective salience scores of the selected sentences while minimizing information redundancy.

We demonstrate the benefits of our model on two tasks: aspect identification and opinion summarization, by comparing it with previous state-of-the-art methods. On the first task, we show that even without any parameters to tune, our model still outperforms previously reported results, and can be further enhanced by introducing extra trainable parameters. For the summarization task, our method exceeds baselines on a variety of evaluation measures.

3.2 Problem Formulation

Extractive opinion summarization aims to select a subset of important opinions from the entire opinion set. For product reviews, the opinion set is a collection of review segments of a certain product. Formally, we use $\mathcal{P}_{c_i}$ to denote all the products belonging to the i-th category c_i (e.g., televisions or bags) in the corpus. Given a target product $p \in \mathcal{P}_{c_i}$, the corpus contains m reviews $\mathcal{R}_p = \cup_{j=1}^{m} \mathcal{R}_p^{(j)}$ of this product, while each review $\mathcal{R}_p^{(j)}$ contains n segments $\{s_1, s_2, \cdots, s_n\}$. We also collect the feature description $\mathcal{F}_p$ of the product as external information, which contains ℓ feature items $\{f_1, f_2, \cdots, f_\ell\}$. The summarization model aims to select a subset of important opinions $\mathcal{O}_p \subseteq \mathcal{R}_p$ that summarize reviews of the product p.

As previously mentioned, one challenge during summarization is to identify aspect-related opinions. In Sec. 3.3, we show how the proposed ASPMEM can tackle this problem, and how to incorporate domain knowledge to enhance model performance. The ranking and selection of the review segments are described in Sec. 3.4.

3.3 Aspect Identification

3.3.1 ASPMEM: Aspect-aware memory

This section describes the proposed ASPMEM model to identify the aspect-related review segments. ASPMEM contains an array of memory cells $\mathcal{A} = \{a_1, a_2, \cdots, a_k\}$ to store aspect-related information. Each cell a_i relates to one specific aspect, and has a low-dimensional embedding $a_i \in \mathbb{R}^d$ in the semantic space, where d is the dimension of the embedding. Each word v_i in a review segment $s = \{v_1, v_2, \cdots, v_n\}$ also has an embedding $v_i \in \mathbb{R}^d$ in the same semantic space.

Similar to topic models, we assume the review segment s is generated from these aspect (topic) memories. However, the LDA-based topic models parameterize the generation probability at word-level, which is too flexible to model short segments in reviews (Yan et al., 2013). We instead regard the review segment as a

whole from a single aspect during generation, but allow every word to have a different contribution to the segment representation.

Given a review segment s, the probability that this segment is generated by the i-th aspect a_i is proportional to the cosine similarity of their vector representations:

$$P(s|a_i) \propto \exp(\cos(s, a_i)), \tag{3.1}$$

where s is the embedding of the segment s, and is defined as the weighted average over embeddings of the words in s:

$$s = \sum_i z_i v_i. \tag{3.2}$$

z_i is the attention weight of the word v_i and is proportional to v_i's generation probability. That is, we focus more on those words which are more likely to be generated by the aspect memories. To compute these weights, we define the probability of v_i being generated from a_j in a similar way:

$$P(v_i|a_j) \propto \exp(\cos(v_i, a_j)), \tag{3.3}$$

$$P(v_i) = \sum_j P(v_i|a_j)P(a_j), \tag{3.4}$$

$$z_i = \frac{P(v_i)}{\sum_j P(v_j)}. \tag{3.5}$$

Without any prior domain knowledge of the aspects, the latent embeddings a_j and the prior probabilities of aspects $P(a_j)$ are parameters (denoted by θ) and can be estimated by minimizing the negative log-likelihood of the corpus $\mathcal{X}$ (i.e., all the review segments belonging to the same product category):

$$J(\theta) = -\sum_{s \in \mathcal{X}} \log P(s; \theta) + \lambda \left\| \hat{\mathbf{A}} \hat{\mathbf{A}}^{\mathbf{T}} - \mathbf{I} \right\|_2. \tag{3.6}$$

The estimation of the likelihood part $P(s; \theta)$ is similar to Eq. 3.4. The second term is a regularization term, where $\hat{\mathbf{A}} \in \mathcal{R}^{k \times d}$ is the aspect embedding matrix with ℓ_2 row normalization, and $\mathbf{I}$ is the identity matrix. It encourages the learned aspects to be diverse, i.e., the aspect embeddings are encouraged to be orthogonal to each other. λ is the hyper-parameter of the regularization.

Once we obtain all the parameters, we can calculate the probability of the review segment s belonging to the aspect a_i as

$$P(a_i|s) \propto P(s|a_i)P(a_i), \tag{3.7}$$

and then select the aspect with the highest posterior probability as the identified aspect.

3.3.2 Incorporating Domain knowledge

The aspect embeddings estimated merely from the data have several shortcomings. First, the model may learn some topics that are irrelevant to the aspects of products, such as sentiments and user profiles. Second, it is difficult to control the granularity of the learned aspects, which may lead to too coarse- or fine-grained aspects.

To address these problems, a simple yet effective method is to use domain knowledge about products. Specifically, rather than estimating a_i according to Eq. 3.6, one could collect several aspect-related seed-words, (e.g., *picture*, *color*, *resolution*, and *bright* for the DISPLAY aspect), and average the embeddings of these seed-words to produce a_i. Previous works have shown the benefit of such knowledge (Fast, Chen and Bernstein, 2017; Angelidis and Lapata, 2018), but they have to encode this knowledge manually or from the human-annotated data, which makes these methods less easy to adapt across product categories.

As we mentioned previously, feature descriptions of products can be a valuable external resource for seed-words mining. Here we describe our unsupervised method of collecting the seed-words from it. To increase the size of this resource, we assume all products in the same category have shared aspects, and collect seed-words from the category level. For each product category c_i, we collect the feature items $\mathcal{F}^{c_i}$ from all products of the same category as the document, i.e., $\mathcal{F}^{c_i} = \bigcup_{p \in \mathcal{P}_{c_i}} \mathcal{F}_p$, and then apply TF-IDF to extract seed-words from it[1]. For TF-IDF to work, we need the seed-words to have high term frequency and the general words to have high document frequency. We therefore aggregate all the items in $\mathcal{F}^{c_i}$ as one single document, and regard the remaining items belonging to other categories as individual documents to build the corpus. For example, assume we have six product categories, while each category contains ten products, and each product has ten feature descriptions. We therefore have 600 feature descriptions in total. To extract the seed-words of one category (e.g., the TV), we concatenate the 100 TV-related descriptions as one single document, while regarding the other 500 descriptions as individual documents. We then calculate the TF-IDF

[1]We also tried other algorithms, but the differences were not significant.

of each word based on these 501 documents. Finally, we select the top K words with the highest TF-IDF value as seed-words of the product category c_i.

3.4 Summary Generation

In the summary generation stage, we first evaluate the salience of each opinion segment, and then select a subset of opinions that form the final summary.

3.4.1 Salience of the opinion

Following Angelidis and Lapata (2018), we evaluate the salience of a review segment s from two perspectives: the relevance to aspects, and the sentiment strength.

Relevance depicts how relevant a segment is to the various aspects of the product. Since one segment may relate to more than one aspect (e.g., *The color is excellent but the sound is terrible.*), we calculate relevance at the word level rather than the segment level. Recall that the relevance of a word to an aspect memory is proportional to the cosine similarity between their embeddings. We assign each word its most related aspect memory (by $\max$ operation), and calculate the relevance of the entire segment as the averaged relevance over all words (by $\sum$ operation). That is,

$$\mathbb{S}_{rel}(s) = \frac{1}{|s|} \sum_{i} \max_{j=\{1,\cdots,K\}} g(\cos(v_i, a_j) \cdot w_j). \tag{3.8}$$

We use the K seed-words extracted from Sec. 3.3.2 as the aspect-related memory, and w_j and a_j are the weight and word embedding of the j-th seed-word. Here the $\cos(v_i, a_j)$ and w_j can be regarded as the unnormalized conditional and prior probabilities in Eq. 3.4. $g(x) = x \cdot I(x - \delta)$ is an activation function to filter the general words whose cosine similarity with any aspects is less than δ. $I(\cdot)$ is the step function. Compared with the relevance measure adopted by Angelidis and Lapata (2018), which uses the probability difference between the most probable aspect and the general one, our score takes a soft assignment between words and aspects, and thus allows the segment to relate to more than one aspect. Also, by regarding each seed-word as a fine-grained aspect, it does not require the seed-words to be clustered into aspects.

Sentiment reflects customers' preferences regarding products and their aspects, which is helpful in decision-making. Since sentiment analysis is not the major contribution of this work, we directly apply the CoreNLP (Socher et al., 2013) to get the sentiment distribution of the reviews. The sentiment distribution is

then mapped onto $[0, 1]$ range as the sentiment score $\mathbb{S}_{senti}$. Sentences with stronger sentiment polarities will have higher values.

Finally, we evaluate the salience of one opinion segment by multiplying the two scores:

$$\mathbb{S}_{sal}(s) = \mathbb{S}_{rel}(s) \times \mathbb{S}_{senti}(s). \tag{3.9}$$

3.4.2 Opinion selection

An ideal summary would contain as many high-salience opinions as possible. However, care should be taken to avoid redundant information. Also, there has to be a limit on the length of the summary (i.e. no longer than L words). These goals can be formalized as an ILP problem. We introduce an indicator variable $\alpha_i \in \{0, 1\}$ to indicate whether to include the i-th segment s_i in the final summary, and then find the optimal α of the following objective:

$$\alpha =_{\alpha} \sum_i \mathbb{S}_{sal}(s_i)\alpha_i - \sum_{i,j} sim_{ij}\beta_{ij}, \tag{3.10}$$

$$s.t. \quad \alpha_i, \beta_{ij} \in \{0, 1\} \qquad \forall i, j \tag{3.11}$$

$$\beta_{ij} \geq \alpha_i + \alpha_j - 1 \quad \forall i, j \tag{3.12}$$

$$\beta_{ij} \leq \frac{1}{2}(\alpha_i + \alpha_j) \quad \forall i, j \tag{3.13}$$

$$\textstyle\sum_i \alpha_i l_i \leq L \qquad \forall i \tag{3.14}$$

where sim_{ij} is the similarity between s_i and s_j. β_{ij} is an auxiliary binary variable that will be 1 iff both α_i and α_j equal to 1, and this is guaranteed by Eq. 3.12 - 3.13. Eq. 3.14 is used to restrict the length of the summary, where l_i is the length of s_i. We solve the ILP with Gurobi[2].

3.5 Experiments

In this experiment, we investigate the utility of ASPMEM for summarization, using the seed-words from external sources and the selection procedure described in Sec. 3.4. We refer to our method as ASPMEMSUM.

[2]http://www.gurobi.com/

Category	#prod	#feature	#token	vocab
Bags	254	5.1	9.2	1491
Headsets	88	4.9	9.5	796
Boots	106	6.0	5.0	472
Keyb/s	142	4.8	10.5	1328
TVs	169	5.0	9.8	905
Vacuums	122	5.0	10.3	878

Table 3.1: The statistics of the external data from six categories. The four columns are: the number of products, the average number of features per product, the average number of tokens per feature, and the entire vocabulary size.

3.5.1 Dataset

We utilize OPOSUM, a review summarization dataset provided by Angelidis and Lapata (2018) to test the efficiency of the proposed method. This dataset contains about 350K reviews from the amazon review dataset (He and McAuley, 2016) under six product categories: *Laptop bags*, *Bluetooth headsets*, *Boots*, *Keyboards*, *Televisions*, and *Vacuums*. Each review sentence is split into segments using a rhetorical structure theory (RST) parser (Feng and Hirst, 2012) to reduce the granularity of opinions. The annotated corpus includes ten products from each category, and ten reviews from each product. They annotate each review segment with an aspect label and produce summaries for each product. We describe the details below:

Aspect information. Each product category has nine pre-defined aspect labels. Each segment is labeled with one or more aspects, including a GENERAL aspect if it does not discuss any specific one. The annotated dataset is split into two equal parts for validation and test. Based on the validation data, they extract 30 seed-words for each aspect and produce the corresponding aspect embedding as a weighted average of seed-words embeddings.

Final summary. For each product, the annotators create a summary by selecting a subset of salient opinions from the review segments and limiting its length to 100 words. Each product has three referenced summaries created by different annotators, which are used only for evaluation.

Their dataset does not contain any external information. We therefore randomly collect the feature descriptions from about 100 products for each category. Table 3.1 gives a statistics about this data. [3]

[3]Available on `https://github.com/zhaochaocs/AspMem`

3.5.2 Experiments on aspect identification

We first investigate the model's ability to identify aspects, which aims to label each review segment with one of the nine aspects (eight specific aspects and one GENERAL aspect) as labeled in the dataset. The method is described in Sec. 3.3. However, instead of using the seed-words obtained from external information (Sec. 3.3.2), we still use those provided with the dataset to enable fair comparison with prior works. Our external seed-words will be used in the summarization experiments (Sec. 3.5.3).

Setup For the eight specific aspects, we assign their corresponding memory cells a_i with the average embedding of the 30 seed-words provided by OPOSUM. For the general aspect, although OPOSUM also provides 30 corresponding seed-words, we handle it differently for the following reasons. First, while the knowledge of specific aspects can be encoded as a few seed-words, it is hard to represent the GENERAL aspect in the same way. A better method is to allow the model to find its intrinsic patterns by relaxing the corresponding GENERAL embedding as trainable parameters. Also, since the number of the GENERAL reviews is approximately ten times more than the specific aspect on average, it is reasonable to assign more memory cells for the GENERAL aspects. Therefore, besides the fixed GENERAL embedding provided by MATE, we have another enhanced model with five extra memory cells to encode the GENERAL aspect. These extra memory cells are initialized randomly and trained to minimize the log-likelihood in Eq. 3.6.

We use 200-dimensional word embeddings which are pre-trained on the training set via word2vec (Mikolov et al., 2013). These embeddings are fixed during training. For simplicity, the prior distribution of aspects is set as uniform. We train the model with batch size of 300, and optimize the objective using Adam (Kingma and Ba, 2015) with a fixed learning rate of 0.001 and an early stopping on the development set. The λ is set as 100. Notice that the model without the extra aspect memories does not have any trainable parameters and therefore can directly be applied for prediction using Eq. 3.7.

We compare the proposed method with ABAE and MATE, two state-of-the-art neural methods, as well as a distillation approach (Karamanolakis, Hsu and Gravano, 2019) that uses the pre-trained BERT (Devlin et al., 2019) as the student model. To ensure a fair comparison, all models utilize the same seed-words. The performance is evaluated through multi-label F_1 score.

Results Table 3.2 shows the average F_1 scores for the four models on the six categories. MATE performs better than ABAE by introducing the human-provided seed-words, which demonstrates the effectiveness of domain knowledge. However, MATE applies the same neural architecture as ABAE, which may not be the best

Model	Bags	Headsets	Boots	Keyb/s	TVs	Vaccums	Average
ABAE	41.6	48.5	41.0	41.3	45.7	40.6	43.2
MATE	48.6	54.5	46.4	45.3	51.8	47.7	49.1
BERT	**61.4**	**66.5**	52.0	57.5	**63.0**	60.4	**60.2**
ASPMEM	52.4	58.1	54.5	51.4	53.9	54.6	54.2
w/ extra memory	60.0	62.0	**55.8**	**61.8**	60.0	**61.8**	**60.2**

Table 3.2: Evaluation of the aspect identification task via multi-class F_1 measure. Our method outperforms MATE on all the categories and achieves a 5.1% increase on average. The extra latent aspect embeddings for the GENERAL aspects further boost the performance by 6.0%.

Aspect	Seed-words
noun	tv television set hdtv item tvs product
adj	good great better awesome superb
verb	figure afford get see find hear watch
number	dd dddd d ddd
problem	issue problem occur encounter flaw
MATE	buy purchase money sale deal week

Table 3.3: The extra GENERAL aspects learned from the data, and the one provided by MATE. Numbers are delexicalized with their shape.

fit to fully leverage the power of the introduced knowledge. Our generative model instead directly cooperates with the aspect memory, not only during the prediction stage but also during the segment encoding. Without any trainable parameters, our method outperforms ABAE and MATE on all the categories and achieves a 5.1% increase on average. It indicates that ASPMEM can get a better aspect-aware segment representation for aspect identification. The extra latent aspect embeddings of the GENERAL aspect (ASPMEM w/ extra memory) help the model better fit the intrinsic structure of the data, which further improves the performance by 6.0%. When comparing with BERT, our model still has better performance on three categories and achieves the same average F_1 score. Note that while BERT is a pre-trained model with 110M parameters, our model only has 1K parameters.

Discussion To further demonstrate the contribution of the extra memories, Figure 3.2 provides the confusion matrices of the results with and without them. The comparison shows that extra memories improve the true-positive rate of the GENERAL aspect from 0.44 to 0.60, while only slightly hurting those of other aspects. Table 3.3 shows the automatically learned GENERAL aspects by listing their nearest words in the embedding space. Compared with the single GENERAL aspect provided by MATE, our model successfully identifies the more varied GENERAL aspects from the reviews, such as the NOUN, VERB, ADJECTIVE, NUMBER, and PROBLEM.

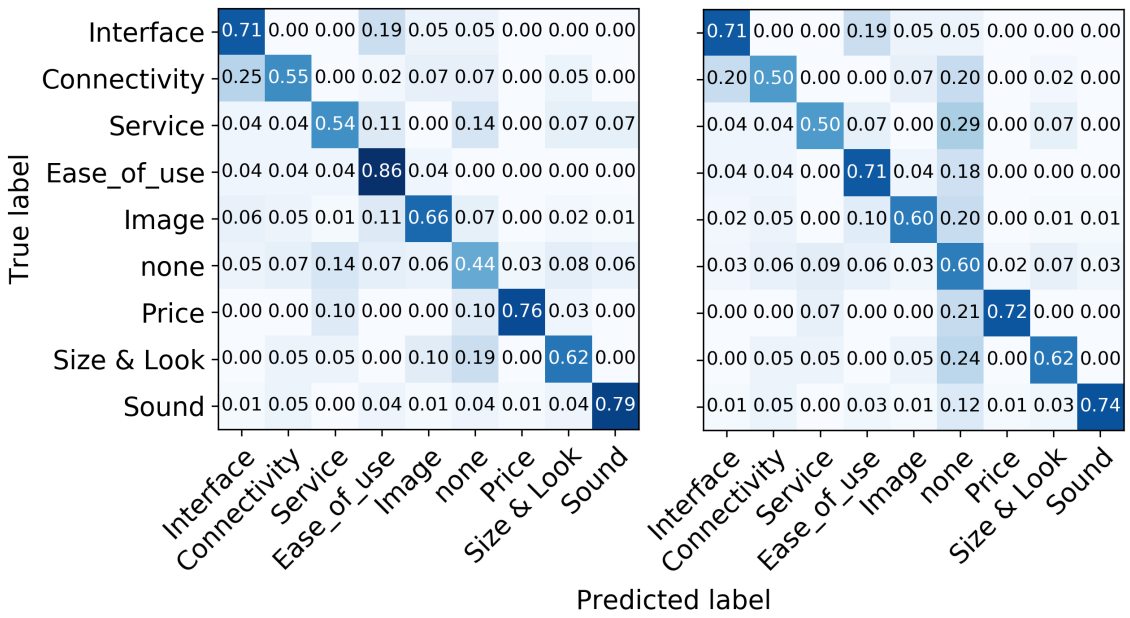

Figure 3.2: Confusion matrix of AspMem results w/o extra memory (left) and w/ extra memory (right). Having extra memories improves performance on the GENERAL aspect without hurting other aspects by much.

3.5.3 Experiments on Summarization

In this experiment, we investigate the utility of ASPMEM for summarization, using the seed-words from external sources and the selection procedure described in Sec. 3.4. We refer to our method as ASPMEMSUM.

Setup With the method described in Sec. 3.3.2, we select top 100 seed-words according to their TF-IDF values, and use their word embeddings as the 100 aspect memories. The similarity threshold δ is set as 0.3. The length of the summary is limited to 100 words or less to enable comparison with the ground-truth summaries. Similar to previous works, we add a redundancy filter to remove the repeated opinions by setting $sim_{ij} = \infty$ when $\cos(s_i, s_j) > 0.5$ otherwise as 0. Other settings are the same as those in the last experiment. We employ ROUGE (Lin, 2004) to evaluate the results. It measures the overlapping percentage of unigrams (ROUGE-1) and bigrams (ROUGE-2) between the generated and the referenced summaries. We compare our method with the reported results in Angelidis and Lapata (2018).

Results Table 3.4 reports the ROUGE-1 and ROUGE-2 scores of each system [4] and the inter-annotator agreement among three annotators. Our method (ASPMEMSUM) significantly outperforms the baselines on both ROUGE scores (approximate randomization (Noreen, 1989; Chinchor, 1992), $N = 9999, p < 0.001$).

[4]MILNET is a sentiment analyzer but its pre-trained model is not public. We therefore replaced it with CoreNLP and obtained the results of MATE as 43.9 and 22.0. There is no significant difference.

Methods	R-1	R-2
Lead	35.5	15.2
LexRank	37.7	14.1
Opinosis	36.8	14.3
MATE + MILNET	44.1	21.8
ASPMEMSUM	46.6	25.7
w/o filtering	**48.0**	**28.7**
w/o Relevance	41.5	20.5
w/o Sentiment	40.5	18.2
w/o ILP	46.2	25.1
Inter-annotator Agreement	54.7	36.6

Table 3.4: Summarization results evaluated by Rouge. The proposed ASPMEMSUM without redundancy filtering achieves the best performance on automatic metrics, and both two perform better than all the baselines.

When removing the redundancy filtering (w/o filtering), it achieves the highest performance. This observation is different from that made by Angelidis and Lapata (2018) who found that redundancy filtering improved the ROUGE scores of results produced by MATE. Upon eyeballing the generated summaries we found that in absence of redundancy filtering, ASPMEM's summaries often included the overlapping part of the three references (i.e., the segments with similar opinions but from different references) more than once. This results in the improvement of ROUGE scores: the more matched n-grams are found, the better the results. However, we prefer to avoid redundancy in order to improve readability.

Effectiveness of opinion selection During the opinion selection, we conduct an ablation study to investigate the contribution of the two salience scores: $\mathbb{S}_{rel}(s)$ for the relevance and $\mathbb{S}_{senti}(s)$ for the sentiment. As shown in Table 3.4, removing the relevance score drops R1 and R2 by 5.1 and 5.2, respectively. Similarly, without sentiment, R1 and R2 drop by 6.1 and 7.5. It demonstrates that both these scores are necessary to capture the salience of an opinion segment.

Finally, we back off our opinion selection procedure to the greedy method to have a fairer comparison with the baseline. As shown in Table 3.4 (w/o ILP), under the same greedy strategy, our method still outperforms the baselines, but using ILP can further improve the results.

Effectiveness of seed-words During the summarization, we extract the seed-words $\mathcal{V}_1$ from external information, whereas those used in MATE (denote by $\mathcal{V}_2$) are extracted from customer reviews with the help of aspect labels. Figure 3.3 provide the distribution of two seed-sets in word embedding space. We analyzed the difference between the two seed-sets, and find that about 81% of words in one seed-set do not appear in

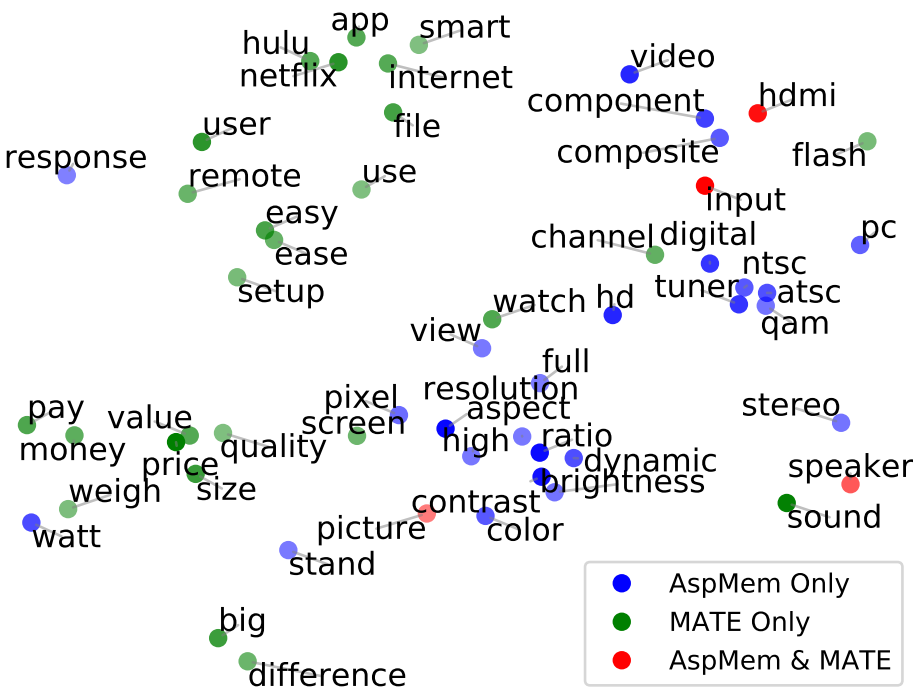

Figure 3.3: The distribution of seed-words in embedding space through t-SNE (Maaten and Hinton, 2008). Each node represents a seed-word and is colored according to the seed-sets it belongs to. Words with higher weights have higher degree of opacity.

the other seed-set. Even the remaining 19% shared seed-words have different weights. Another observation is that the seed-words from feature descriptions tend to be nouns, while those from review texts contain more adjectives. It can also be reflected in Figure 3.3, where the words from two seed-sets are separated into two parts. It reflects the fact that the content in feature descriptions is more objective than that in customer reviews, making it a better source to analyze the aspect relevancy than the reviews themselves.

We then replace our seed-words with those used in MATE to delineate the contributions of the model from that of the seed-set. When using the same seed-words, our model achieves 45.6 and 24.5 for ROUGE-1 and ROUGE-2, which are still better than the results of MATE. This indicates that the model itself also contributes to the performance gain.

Finally, we analyze the effect of two seeds-related hyperparameters on ROUGE metrics: the size of the seed-set, and the similarity threshold δ of seed-words (see $g(\cdot)$ in Eq. 3.8). We vary the size of the seed-set from 10 to 200, and δ from 0.1 to 0.5. The results are shown in Figure 3.4. When there are only a few seed-words, the model performance rapidly increases with the growth of the seed-set size. For larger

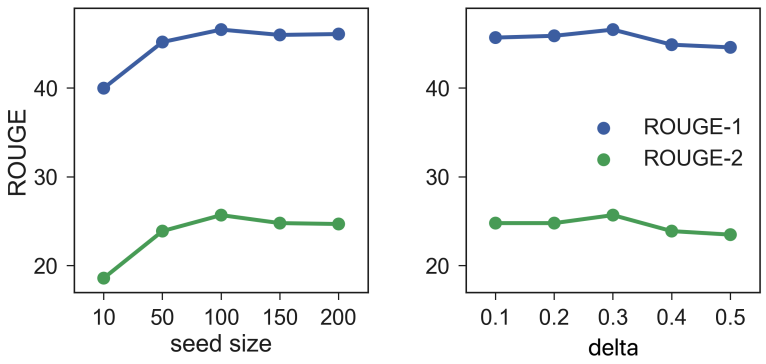

Figure 3.4: The effect of the seeds size (left) and the similarity threshold (right) on the ROUGE metrics.

MATE	Picture is crisp and clear with lots of options to change for personal preferences. Plenty of ports and settings to satisfy most everyone. The sound is good and strong. But the numbers of options available in the on-line area of the Tv are numerous and extremely useful! I am very disappointed with this TV for two reasons : picture brightness and channel menu. The software and apps built into this TV are difficult to use and setup Unit developed a high pitch whine
ASPMEM	Unit developed a high pitch whine. The picture is beautiful. This TV looks very good. The sound is clear as well. there is a dedicated button on the remote. I am very disappointed with this TV for two reasons : picture brightness and channel menu. which is TOO SLOW to stream HD video... and it will not work with an HDMI connection because of a conflict with Comcast's DHCP.
Human	Picture is crisp and clear with lots of options to change for personal preferences. Plenty of ports and settings to satisfy most everyone. The sound is good and strong. But the numbers of options available in the on-line area of the Tv are numerous and extremely useful! I am very disappointed with this TV for two reasons : picture brightness and channel menu. The software and apps built into this TV are difficult to use and setup Unit developed a high pitch whine

Table 3.5: A summary example generated by MATE and our method, compared with a human-generated summary. We use the same product (Sony BRAVIA HDTV) reported by Angelidis and Lapata (2018).

seed-sets (more than 100 words), the number of noisy words increases and this slightly hurts the performance. Meanwhile, we find that our model is also robust to the choice of δ, especially for small values (less than 0.3).

Qualitative analysis

Table 3.5 shows summaries of the same product generated by MATE, our method (ASPMEMSUM), and one of the human annotators. Similar to humans, MATE and ASPMEMSUM are also able to select aspect-related opinions. The difference is that ASPMEMSUM learns these aspects without any human effort.

3.6 Conclusion

In this work, we propose a generative approach to create summaries from online product reviews without specific human annotation. At the model level, we introduce the aspect-aware memory to fully leverage the domain knowledge. It also reduces the parameters and computation cost of the model. At the data level, we collect the domain knowledge from external information rather than through human effort, which makes the proposed method easier to adapt to other product categories. By comparing with the state-of-the-art models on both aspect identification and opinion summarization tasks, we experimentally demonstrate the effectiveness of our approach. Future works can design better measures for opinion selection, and incorporate abstractive methods to enhance the readability of the generated summaries.

CHAPTER 4: DOCUMENT REORDERING FOR MULTI-DOCUMENT NEWS SUMMARIZATION

In the previous chapter, we discussed how to leverage external data sources and derive supervision signals from them. In this chapter, we explore how to transfer knowledge from existing data-rich tasks to data-deficient tasks. More specifically, we focus on the knowledge transfer from single-document summarization to multi-document summarization, and propose a document reordering approach to better leverage knowledge from the single-document summarization model.

4.1 Background

Multi-document news extractive summarization (MDS) aims to extract the salient information from multiple related news documents into a concise summary. Some approaches use task-specific architectures for this problem. For example, Wang et al. (2020) organize multiple documents as a heterogeneous graph before summarizing them. Zhong et al. (2020) formulate the extractive summarization task as a semantic matching problem. Recent works also explored reformulating this problem as a single-document summarization (SDS) problem by concatenating all documents into a single *meta-document* and then using an SDS model to summarize it (Cao et al., 2017; Liu et al., 2018; Lebanoff, Song and Liu, 2018; Fabbri et al., 2019).

Due to the conventions of news writing (Hong and Nenkova, 2014; Hicks et al., 2016), salient information often appears at the beginning of a news article. As a result, many summarization systems, including recent neural models (Kedzie, McKeown and Daumé III, 2018; Zhong et al., 2019), pay more attention to the beginning of the document. Therefore, in MDS, it is important to consider the order in which the documents are concatenated to form the *meta-document* before applying the summarization model.

Specifically, we argue that the various documents in the input are not equally important. Some documents contain more salient or detailed information and are more important. Therefore, compared with concatenating documents in an arbitrary order, it would be beneficial to reorder the documents such that the important ones are in the front of the meta-document and it becomes easier for the summarization model to learn the salient content.

Motivated by these factors, we propose a simple yet effective approach to reorder the input documents according to their relative importance before applying a summarization model. We evaluate the effectiveness of our approach on Multi-News (Fabbri et al., 2019) and DUC-2004. [1] Results show that our simple reordering approach significantly outperforms the state-of-the-art methods with more complex model architectures. We also observe that this approach brings more performance gain with the increase in the number of input documents.

4.2 Problem Formulation

We refer to $\mathcal{D}$ as a meta-document of m documents $\{d_1, \ldots, d_m\}$ with n sentences $\{s_1, ..., s_n\}$ in total. The goal of extractive summarization is to extract a subset of sentences in $\mathcal{D}$ to summarize the input documents. It is usually formulated as a binary sentence classification problem, where each sentence is assigned a $\{0, 1\}$ label to determine if it is to be included in the summary.

4.3 Method

In the following subsections, we first introduce our document reordering approach, and then the base summarization model.

4.3.1 Document Reordering

Document reordering aims to rearrange documents of the meta-document in order of their salience. It can be formulated as determining the relative importance score of each document and then reordering the documents according to their importance scores. Here we propose a supervised approach and an unsupervised approach for this task.

Supervised Approach. In this approach, we use a BERT (Devlin et al., 2019) based model to learn document importance scores. For this, we first concatenate the documents together while inserting a `[CLS]` and a `[SEP]` token at the start and the end of each document. We then encode the concatenated documents using BERT to get the document representation $t_i \in \mathbb{R}^K$, which is the representation of the `[CLS]` token preceding it. To enhance the model's ability to capture the inter-document relationships, we use a 2-layer

[1] https://duc.nist.gov/data.html

Transformer to encode t_i and finally obtain a document's *contextualized* representation $h_i \in \mathbb{R}^K$.

$$\begin{aligned} t_1, \ldots, t_m &= \text{BERT}(d_1, \ldots, d_m) \\ h_1, \ldots, h_m &= \text{Transformer}(t_1, \ldots, t_m) \end{aligned} \tag{4.1}$$

Thereafter, in order to predict the importance score for the i-th document, $\hat{y}_i$, we apply a linear transformation with a Softmax function.

$$\hat{y}_i = \text{softmax}\left(W h_i + b\right), \tag{4.2}$$

where $W \in \mathbb{R}^{K \times K}$ and $b \in \mathbb{R}^K$ are parameters.

During training, we determine the oracle importance score of each document d_i as the normalized ROUGE-1 F score [2] between d_i and the gold abstractive summary S:

$$y_i = \frac{\text{ROUGE}(d_i, S)}{\sum_i \text{ROUGE}(d_i, S)}. \tag{4.3}$$

Our learning objective is to minimize the Kullback–Leibler divergence between the predicted distribution $\hat{y} = \{\hat{y}_1, \ldots, \hat{y}_m\}$ and the oracle distribution $y = \{y_1, \ldots, y_m\}$ of importance scores.

$$\mathcal{L} = \text{KL}(\hat{y}, y) \tag{4.4}$$

We train the document reordering model on the training set based on this learning objective.

During inference, we obtain the importance score of documents in the validation set and test set based on Eq. 4.2, and then reorder documents in descending order of their importance scores to create the meta-document.

Unsupervised Approach. We hypothesize that the importance of a document is related to its centrality. To test this hypothesis, we propose an unsupervised centrality-based document reordering approach. To compute the centrality of a document d_i, we first represent the topic of the input cluster, T_i, by concatenating the top-3 sentences of each document except d_i, and then calculate the centrality as $\text{ROUGE}(d_i, T_i)$. We choose top-3 sentences to represent the topic as it is a strong unsupervised summarization baseline. We avoid sentences of

[2] We also tried ROUGE-2 F or ROUGE-1 R but didn't observe a significant difference.

d_i to be included in T_i to prevent the centrality of d_i being dominated by its own sentences, leading to similar centrality scores for all documents.

Finally, we reorder the documents in descending order of their centrality scores and then concatenate them into a meta-document.

4.3.2 Base Extractive Summarization Model

Once the documents have been reordered and concatenated to form a meta-document, they are fed to a base summarization model. **For the supervised reordering approach**, we use PreSumm (Liu and Lapata, 2019*b*), a state-of-the-art SDS method. It uses BERT as the encoder to get the sentence representations, and a linear transformation with a Sigmoid as the decoder to get the probability of selecting a sentence. The loss function is the averaged cross-entropy between the predicted probability and the oracle $\{0, 1\}$ label of each sentence. When applying this model to MDS, we insert a null sentence ("`[CLS] [SEP]`") between consecutive (reordered) documents in the meta-document as the document delimiter. It helps the model to identify document boundaries and build inter-document relationships.

For training, the extractive oracle labels are obtained by incrementally adding sentences to the extracted summary until the ROUGE score between the extracted summary and the gold abstractive summary does not increase. Using an SDS-based model architecture also facilitates transferring knowledge from SDS datasets. For this, we first finetune the model on SDS datasets and then finetune it on our MDS dataset.

For the unsupervised reordering approach, we use PacSum (Zheng and Lapata, 2019), a BERT-based model to measure the centrality of each sentence in the meta-document and then select sentences accordingly. Different from other centrality-based methods, PacSum builds a directed graph to explicitly model the order of sentences. Therefore PacSum can benefit from a meta-document where the salient documents are rearranged to the front. When applying it to MDS, we build the graph for the meta-document and calculate the centrality of each sentence accordingly.

4.4 Experiments

In this section, we evaluate our document reordering based summarization approach.

4.4.1 Dataset

We conduct experiments on two MDS datasets: Multi-News and DUC-2004. Multi-News is the largest multi-document summarization dataset in the news domain. It contains 44,972/5,622/5,622 instances for training/validation/test. Each instance contains a set of news articles and an abstractive summary. The number of articles varies between 2 and 10. For evaluation, we compare the extracted summary to the gold abstractive summary. DUC-2004 contains 50 instances. Each instance has 10 documents and their abstractive summaries. Due to its small size, we use this dataset for out-of-domain evaluation only.

We also use CNN DailyMail (CNNDM) (Nallapati et al., 2016), a single-document news summarization dataset, to pretrain the base summarization model. It contains around 300K news articles and corresponding summaries from CNN and the Daily Mail.

4.4.2 Setup

We use $\text{BERT}_{\text{BASE}}$ as the encoder of both the document reordering model and base summarization model. We experiment with training the summarization model from scratch and also initializing it with parameters learned by training on CNNDM. The training loss is optimized using Adam (Kingma and Ba, 2015) with a learning rate of 2×10^{-3} and 10,000 training steps. We apply the warmup (Goyal et al., 2017) on the first 2,000 steps and the early stopping based on the ROUGE-1 score on the development set. The batch size is set as 6,000 tokens. Our model was trained on a single Quadro RTX 5000 GPU in 2 hours. During inference, we choose the top-K sentences with the highest score to compose the final summary, where K is selected based on the average length of summaries in the training set. We set $K = 9$ and 7 for Multi-News and DUC-2004, respectively.

We compare our approach with the following baselines: Lead-N, TextRank (Mihalcea and Tarau, 2004), LexRank (Erkan and Radev, 2004), HiBERT (Zhang, Wei and Zhou, 2019), MGSum-ext (Jin, Wang and Wan, 2020), HDSG (Wang et al., 2020), and MatchSum (Zhong et al., 2020). Lead-N concatenates the top-N sentence of each document. We try $N = \{1, 2, 3\}$ and report the best performance. Following these approaches, we evaluate the extractive summaries using ROUGE F_1 score.[3]

We evaluate the document reordering model by comparing the predicted document order with the oracle order via Kendall's Tau (τ) and Perfect Match Ratio (PMR), two common metrics for ranking tasks (Basu

[3]We use pyrouge (`https://github.com/bheinzerling/pyrouge`)

MODEL	R1	R2	RL
Lead (Fabbri et al., 2019)	43.08	14.27	38.97
LexRank (Erkan and Radev, 2004)	41.77	13.81	37.87
TextRank (Mihalcea and Tarau, 2004)	41.95	13.86	38.07
HiBERT (Zhang, Wei and Zhou, 2019)	43.86	14.62	-
MGSum-ext (Jin, Wang and Wan, 2020)	44.75	15.75	-
HDSG (Wang et al., 2020)	46.05	16.35	42.08
MatchSum (Zhong et al., 2020)	46.20	16.51	41.89
Unsupervised			
PACSUM	43.02	14.03	39.02
PACSUM + DR_{unsup} (Ours)	43.57	14.41	39.52
Trasfer + Zero-shot, w/ finetune on CNNDM			
PRESUMM	43.72	14.33	39.71
PRESUMM + DR_{unsup} (Ours)	44.09	14.54	40.05
PRESUMM + DR_{sup} (Ours)	44.62	15.00	40.58
Supervised, w finetune on Multi-News			
PRESUMM	46.05	16.56	41.91
PRESUMM + DR_{sup} (Ours)	46.34	16.88	42.20
Trasfer + Supervised, w/ finetune on CNNDM and Multi-News			
PRESUMM	46.25	16.75	42.11
PRESUMM + DR_{sup} (Ours)	**46.57**	**17.10**	**42.44**
Oracle	49.06	21.54	44.27

Table 4.1: Summarization results evaluated on Multi-News by ROUGE 1 (R1), ROUGE 2 (R2), and ROUGE L (RL). Our best results (in **bold**) show statistically significant difference with the baselines (using paired bootstrap resampling, $p < 0.05$ (Koehn and Monz, 2006)).

Roy Chowdhury, Brahman and Chaturvedi, 2021). We compare our approach with a random baseline and a length-based baseline that rearranges documents in decreasing order of their lengths.[4]

4.4.3 Results

Automatic Evaluation Table 4.1 shows results on Multi-News using either supervised or unsupervised document reordering approach. We first investigate the utility of transferring knowledge from SDS. For this, we compare the PreSumm models in both zero-shot and fully-supervised settings.

For the **zero-shot setting**, we directly use the PreSumm model trained on CNNDM to extract the summary. Results show that PreSumm w/ CNNDM outperforms all unsupervised methods, indicating that transfering knowledge from SDS is an effective method to improve MDS. By incorporating our unsupervised

[4]We do not include advanced baselines as performance of the document reordering is not the main focus of our work.

document reordering method, PreSumm + DR_{unsup} can further improve the model performance (44.09 vs. 43.72 on ROUGE-1). We further show that with a stronger document reordering model (e.g., the supervised model DR_{sup}), the performance on MDS can be further improved by a large margin to 44.62 on ROUGE-1. It demonstrates that with a model trained on SDS dataset, we can improve its performance on MDS by merely rearrange the input documents of MDS into a better order.

For the **fully-supervised setting**, We test the performance of our supervised document reordering (DR_{sup}) approach on the supervised model (PreSumm w/ CNNDM + Multi-News). Using document reordering, our approach, PreSumm + DR_{sup}, significantly outperforms the vanilla PreSumm on all ROUGE scores with or without CNNDM (46.57 vs. 46.25 on ROUGE-1). It demonstrates that document reordering is still effective under fully-supervised transfer setting.

Besides the task-transfer from SDS to MDS, we also show that our document reordering approaches can significantly improve the state-of-the-art unsupervised and supervised MDS models (PacSum and PreSumm w/ Multi-News) on all ROUGE scores. Similarly, the unsupervised approach, PacSum + DR_{unsup}, also outperforms the unsupervised baselines. All these improvements demonstrate that document reordering is an effective way to leverage existing strong models for summarization.

Human Evaluation We also conduct a human evaluation to better assess the performance of each system. We randomly select 100 test instances to evaluate the performance of each system as in Iskender, Polzehl and Möller (2021). The three measures we used are 1) Informativeness: whether or not the summary reflects the salient information of the reference summary; 2) Conciseness: whether or not the summary contains no redundant words or repeated information; and 3) Usefulness: whether or not the summary helps the reader catch the main idea of the news. Human judges were paid at a wage rate of $8 per hour, which is higher than the local minimum wage rate.

We conduct a pairwise comparison of PreSumm+DR_{sup} (the best model) with PreSumm and MatchSum, two strongest neural baselines, as well as LEAD, the best unsupervised baseline. For each test instance, we obtain the output summary from our model and one of the baselines, and then ask three workers on Amazon Mechanical Turk to compare the two summaries according to the three measures listed above.

When comparing a certain baseline approach to our model, we report the percentage of summaries created by the baseline that were judged to be better/worse/same than those of our model, yielding a score ranging from -1 (unanimously worse) to 1 (unanimously better). For example, when evaluating the informativeness

Model	Informative	Concise	Useful
LEAD	-0.20	-0.14	-0.17
MatchSum	-0.12	-0.05	-0.08
PreSumm	-0.06	0.03	-0.07

Table 4.2: Results of human evaluation by comparing three baselines with PreSumm+DR_{sup}. A positive score means the baseline is better than ours and vise versa.

MODEL	R1	R2	RL
Lead-1	33.86	7.51	29.64
TextRank	33.09	7.49	29.25
MatchSum	33.84	7.44	30.07
PreSumm	34.42	7.95	30.34
PreSumm + DR_{sup}	34.62	8.22	30.54

Table 4.3: Out-of-domain summarization results evaluated on DUC 2004 using the model trained on Multi-News. Our approach (last row) outperforms the baselines.

scores, Lead performs better/worse/same than our model for 36%/56%/8% of the instances, yielding a pairwise score as 0.36-0.56=-0.20.

The results are shown in Table 4.2. Negative scores indicate worse performance compared with PreSumm+DR_{sup}. The results show that our approach can generate more informative, concise, and useful summaries compared to baselines, which is consistent with the automatic results.

Out-of-domain Evaluation We further evaluate the performance of our approach in an out-of-domain setting. We compare our best approach with Lead-1, TextRank, MatchSum, and PreSumm. All models except Lead-1 and TextRank were trained on Multi-news and evaluated on the DUC 2004 dataset via Rouge F_1 scores. As shown in Table 4.3, our approach (last row of the table) achieves consistently better performance than the baselines, indicating that our approach can effectively transfer to new unseen domains.

4.4.4 Document-wise Analysis

In this section, we first compare our two document reordering approaches using ranking measures (τ and PMR) and ROUGE scores of the extracted summaries. Table 4.4 shows the results. Our supervised ranking method (DR_{sup}) outperforms the unsupervised method (DR_{unsup}), demonstrating that the oracle importance score of the document is an effective supervision signal for document reordering. DR_{unsup} achieves higher

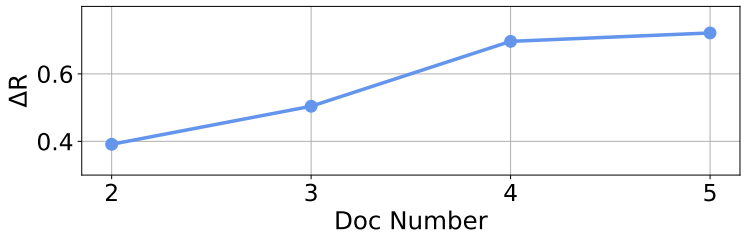

Figure 4.1: Performance gain of summarization w.r.t. the number of input documents. We don't include instances with 6 or more documents since the number of such instances is small. Our approach results in more performance gain for longer inputs.

MODEL	Reordering		Summarization		
	τ	PMR	R1	R2	RL
Random	-0.005	31.8	46.25	16.75	42.11
Length	0.189	43.2	46.30	16.73	42.15
DR_{unsup}	0.236	46.4	46.41	16.94	42.26
DR_{sup}	0.325	51.7	46.57	17.10	42.44

Table 4.4: Reordering methods evaluated on Multi-News. Our approaches, PreSumm + DR_{sup} and PreSumm + DR_{unsup} outperform the baselines.

scores than baselines. It supports our hypothesis that the importance of documents is related to their centrality to the topic.

We further analyze the impact of instance length (number of documents in the instance) on the model performance. In Figure 4.1, we group the test instances of Multi-News based on their lengths, and show the gain in summarization performance obtained from supervised reordering (measured using the ROUGE-1 difference ΔR between the models with and without document reordering). The figure shows that in general, ΔR increases as the instance length increases, indicating that instances with more documents benefit more from our reordering approach.

4.4.5 Summary-wise Analysis

The underlying assumption behind our document reordering approach is that extractive summarization models tend to select sentences from the beginning of the document. By reordering the important documents to the front of the meta-document, our approach makes the salient content easier to learn. In this section, we investigate if this is indeed what is happening by analyzing the distribution of the oracle and the generated summary sentences in the meta-document. We conduct three experiments.

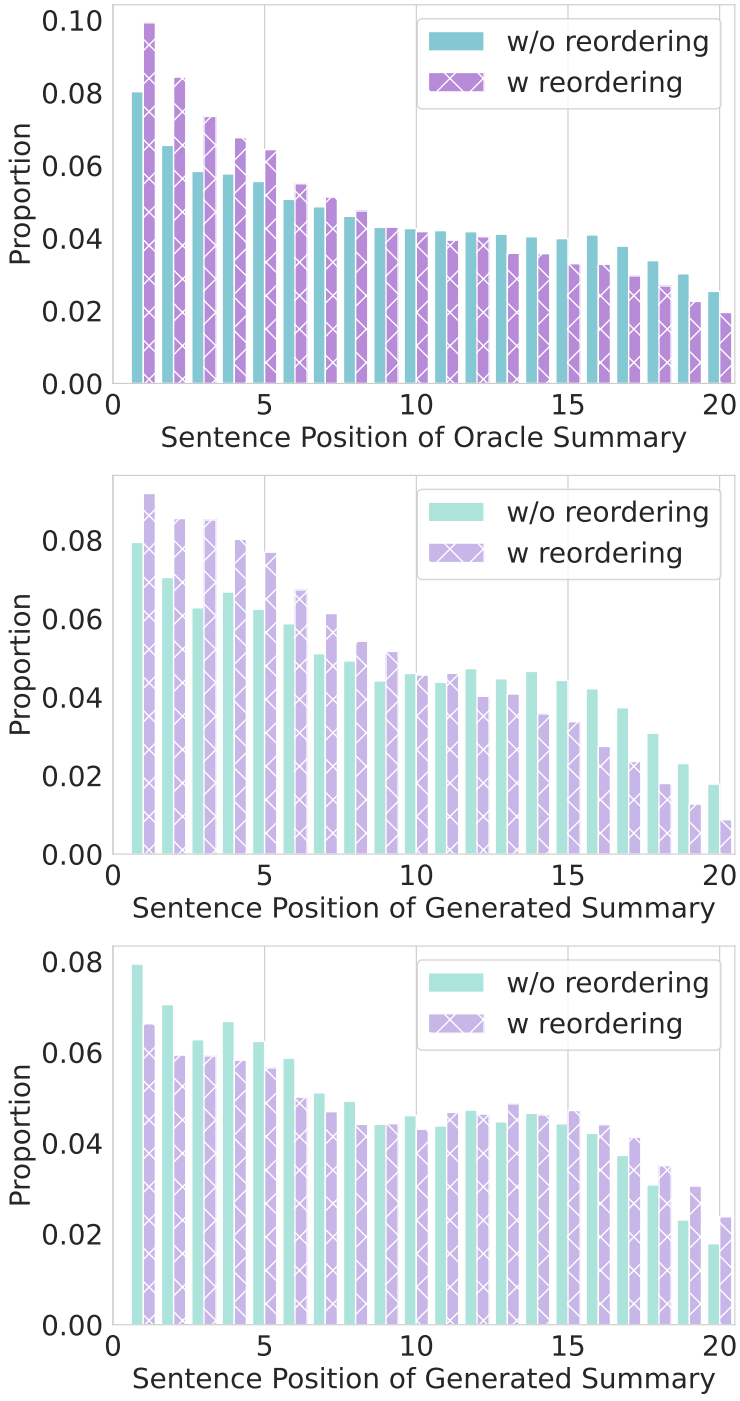

Figure 4.2: (a) The distribution of *oracle* extractive summaries according to their sentence positions in the meta-document with and without document reordering. (b) The distribution of *generated* extractive summaries according to their sentence positions in the meta-document with and without document reordering. (c) The distribution of *generated* extractive summaries according to their sentence positions in the original, unordered meta-document.

Experiment 1: We first investigate how reordering is changing the placement of important sentences. We represent important sentences as those in the oracle summaries, which is obtained by following the procedure described in Section 4.3.2. Figure 4.2(a) shows the distribution of oracle summary-sentences at various positions of the input meta-document when it is reordered (purple shaded bars) and when it is not reordered (blue solid bars). The x-axis shows the sentence positions in the input meta-document and the y-axis shows the fraction of sentences from the oracle summary that were at that position in the meta-document. Comparing the purple and blue bars in the left area, more oracle summary's sentences were located at the beginning of the reordered input meta-document compared with the unordered input meta-document. This indicates that reordering helps in placing the important sentences in the beginning of the input meta-document.

Experiment 2: We next investigate if the summarization model favors certain sentence positions. Figure 4.2(b) shows the distribution of (generated) summary-sentences with respect to various positions of the input meta-document for PreSumm+DR (w/ reordering, purple shaded bars) and PreSumm (w/o reordering, blue solid bars). Like Figure 4.2(a), the x-axis shows the sentence positions in the input meta-document, but the y-axis shows the fraction of sentences from the *generated* summary that are at that position in the meta-document. The bars on the left are, in general, higher than the bars on the right. This indicates that PreSumm tends to pick sentences appearing at the beginning of the input meta-document to create summaries.

Experiment 3: Finally, we want to investigate if the reordering can help the model select salient content that was originally scattered across the input. Figure 4.2(c) shows the distribution of (generated) summary-sentences with respect to various positions of the *original* unordered meta-document for PreSumm+DR (w/ reordering, purple shaded bars) and PreSumm (w/o reordering, blue solid bars). The x-axis shows the sentence positions in the *original* meta-document and the y-axis shows the fraction of sentences from the generated summary that were at that position in the meta-document. We see that compared with the blue bars, the purple bars have a more uniform distribution. This indicates that the reordering based model has a greater tendency to pick sentences that were located at unfavorable positions (towards the end) in the original meta-document. The reordering helps in moving these sentences to the front, and then the summarization models pick them for generating the summary.

Overall, from these experiments, we can conclude that since the base summarization model pays more attention to the beginning of the input (Experiment 2), by moving important content towards the beginning of the input (Experiment 1), the reordering method helps the summarization model also focus on information

that was scattered across the original unordered input (Experiment 3). We also provide a qualitative analysis below to show how the document reordering helps the model generate better summaries.

4.4.6 Qualitative Analysis

Table 4.5 shows an example with 4 source documents listed in the original order. The main event of this example is about a child abduction case, where source 3 and 4 provide more direct and detailed information compared with source 1 and 2.

We show the summaries generated by MatchSum, PreSumm, and our system, as well as the reference summary. MatchSum and PreSumm receive the documents in the original order, making them focus more on the top two documents. Our method first rearranges the documents as the order of $\{3, 4, 2, 1\}$ and then creates the summary based on the new re-ordered documents. With the help of the document reordering, our summary better captures the main event from the latter source documents (source 3 and source 4).

4.5 Conclusion

In this work, we propose a document reordering based approach for multi-document news summarization. We rearrange the documents according to their relative importance while concatenating them into a meta-document and then apply a summarization model. Our simple yet effective approach outperforms the baselines on two multi-document summarization datasets, demonstrating that document reordering is a promising direction for multi-document news summarization. The next step, which we leave for future work, is to explore the scalability of such approaches on large document clusters.

Source 1: these items are among those purchased by gary simpson , prior to taking 9-year-old carlie trent from her school in rogersville , tn on may 4th ... share this : twitter facebook linkedin google email like this : like loading ...

Source 2: by hayes hickman of the knoxville news sentinel two knoxville banking executives are offering a $ 10,000 reward for information leading to the return of missing 9-year-old carlie marie trent, who was abducted a week ago by her uncle in hawkins county . matt daniels , president and chief executive officer of apex bank , said he and his business partner , 21st mortgage president tim williams , felt compelled to get involved as the search continues ... “this is agonizing , ” daniels said wednesday .” “it ’ s not a 24-hour amber alert. ” daniels said carlie reminds him of his two daughters ...

Source 3: it ’ s been a week since carlie trent was in school . on wednesday , tennessee bureau of investigation asked the public to trust them that carlie trent really is in danger . josh devine , a spokesperson for tbi , said he has heard some disturbing rumors on social media that if simpson was trying to protect carlie trent . he told wate he has seen tweets that asked ” if simpson didn ’ t harm carlie when he had custody , why would he do it now .” ” i don ’ t think he ’ s trying to protect her .” this was not an innocent camping trip , this was a crime .

Source 4: the mother of missing tennessee 9-year old carlie trent says she älways had a bad feeling äbout gary simpson , the 57-year-old man who allegedly kidnapped her daughter last week on may 4 . simpson , carlie ’ s uncle by marriage who , along with his wife , had once had custody of carlie , picked her up from her rogersville elementary school . simpson and carlie did not return home later that afternoon and the following day an amber alert was issued in tennessee ...

MatchSum: by hayes hickman of the knoxville news sentinel two knoxville banking executives are offering a $ 10,000 reward for information leading to the return of **missing 9-year-old carlie marie trent , who was abducted a week ago by her uncle in hawkins county** . matt daniels , president and chief executive officer of apex bank , said he and his business partner , 21st mortgage president tim williams , felt compelled to get involved as the search continues ... “ this is agonizing , ” daniels said wednesday . “ it ’ s not a 24-hour amber alert . ” daniels said carlie reminds him of his two daughters ...

PreSumm: by hayes hickman of the knoxville news sentinel two knoxville banking executives are offering a $ 10,000 reward for information leading to the return of **missing 9-year-old carlie marie trent , who was abducted a week ago by her uncle in hawkins county .** matt daniels , president and chief executive officer of apex bank , said he and his business partner , 21st mortgage president tim williams , felt compelled to get involved as the search continues ... mother of allegedly abducted 9-year-old carlie trent **’ always had a bad feeling ’ about suspect** . these items are among those purchased by gary simpson , prior to **taking 9-year-old carlie trent from her school** in rogersville , tn on may 4th ...

Ours: the mother of **missing tennessee 9-year old carlie trent** says she **” always had a bad feeling ” about gary simpson , the 57-year-old man** who allegedly **kidnapped her daughter last week on may 4 , simpson, carlie ’ s uncle ... picked her up from her rogersville elementary school .** he told wate he has seen tweets that asked ” if simpson didn ’ t harm carlie when he had custody , why would he do it now . “it ’ s not a 24-hour amber alert. **this was not an innocent camping trip , this was a crime . ” i don ’ t think he ’ s trying to protect her .”** simpson and carlie did not return home later that afternoon and the following day an amber alert was issued in tennessee ...

Reference: – authorities are combing through more than 1,200 leads in a desperate search for **a 9-year-old girl they say was abducted by her uncle may 4** , wate reports . according to the knoxville news sentinel , **57-year-old gary simpson picked carlie trent up from her tennessee school** ... the tbi says **there have been rumors online that simpson is trying to protect carlie** , but it says that couldn ’ t be further from the truth . **this was not an innocent camping trip , this was a crime** ... shannon trent , who hasn ́ had custody of carlie in two years , says **she ” always had a bad feeling ” about simpson** ...

Table 4.5: Sample summaries generated by our method and the baselines. MatchSum and PreSumm receives the documents as the original order, making them focus more on the top two documents. Our method first rearrange the documents as the order of $\{3, 4, 2, 1\}$ and then create the summary. We highlight the contents of the generated summaries which are relevant to the referenced summary.

CHAPTER 5: NARRATIVE PRE-TRAINING FOR ZERO-SHOT DIALOGUE UNDERSTANDING AND SUMMARIZATION

In this chapter, we tackle the problem of low supervision by adopting a more direct approach: constructing synthetic (document, summary) pairs to pre-train a summarization model. Specifically, we focus on the dialogue summarization task and build a large-scale pre-training dataset to facilitate the model's acquisition of valuable knowledge for dialogue comprehension and summarization. Experimental results demonstrate that our pre-trained model achieves superior zero-shot performance on various dialogue understanding and summarization tasks.

5.1 Background

Dialogue summarization requires the model to generate a concise summary of a dialogue. Before generating the dialogue, the model needs to fully understand the salient information of the dialogue, which belongs to the understanding problem. Recent advances in pre-trained language models (PLMs) (Lewis et al., 2020; Radford et al., 2019) have been applied to both dialogue understanding (Jin et al., 2020; Liu, Feng, Wang, Song, Ren and Zhang, 2021) and summarization (Feng, Feng, Qin, Qin and Liu, 2021; Zhang, Ni, Yu, Zhang, Zhu, Deb, Celikyilmaz, Awadallah and Radev, 2021). However, these PLMs are generally pre-trained on formal-written texts, which are different from dialogue data in nature. Specifically, dialogues are composed of colloquial utterances from multi-speakers, and utterances usually have complex discourse structures (Afantenos et al., 2015). Therefore, applying these models directly to dialogue understanding and summarization, especially in low-resource settings, is sub-optimal.

To learn better dialogue representations, recent studies have designed several dialogue-specific pre-training objectives such as speaker prediction (Qiu, Zhang and Zhou, 2021), utterance prediction (Chapuis et al., 2020), response selection (Wu et al., 2020), and turn order restoration (Zhang and Zhao, 2021). These methods, albeit improve over the vanilla PLMs, usually rely on surface-level dialogue information. In particular, they still fail to train the models to explicitly learn the aforementioned capabilities which are critical for dialogue understanding (e.g., linguistic knowledge, world knowledge, and commonsense knowledge).

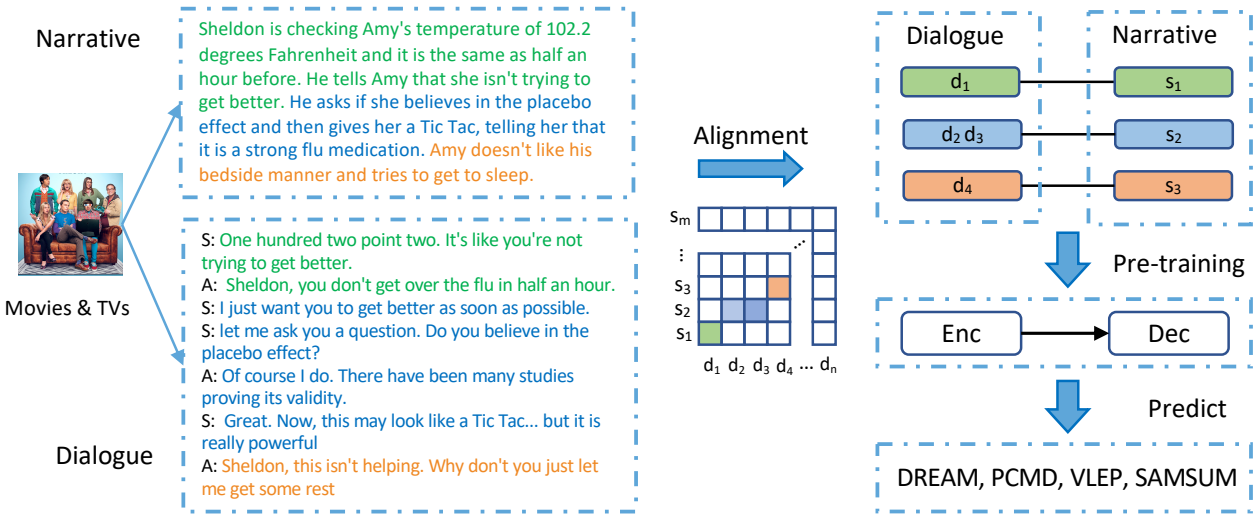

Figure 5.1: Overview of the *learning-by-narrating* strategy for pre-training a zero-shot dialogue understanding and summarization model (with an encoder-decoder architecture).

Furthermore, it was not able to incorporate knowledge beyond dialogue (e.g., non-verbal communications between speakers, as well as time and location information), which are also crucial for dialogue understanding and summarization.

To pre-train a zero-shot dialogue understanding and summarization model with the aforementioned features, we develop a novel generative pre-training strategy that *learns by narrating* the key information from a dialogue input (see Figure 5.1 for an example). In particular, the generated narrative text is supposed to not only (i) paraphrase the gists of the dialogue but also (ii) carry certain inferred information (e.g., the time and location of a scene and relations between speakers) that are not explicitly mentioned in the dialogues.

Learning to narrate such information helps the model to learn varied lexical, syntactic, and semantic knowledge of dialogue. It also enhances the model's ability to infer extra information beyond the literal meaning within dialogues, which will benefit the model's capability of dialogue understanding. Finally, during narrating, the model can naturally learn the ability to convert a multi-party dialogue to a monologue, which narrows the style gap between the input and the output of the summarization task.

However, the *learning-by-narrating* strategy would require a dialogue-narrative parallel corpus, which, to our best knowledge, is not publicly available. For this reason, we first create **DIANA**, a large-scale dataset with (**DIA**logue, **NA**rrative) pairs automatically collected from subtitles of movies and their corresponding plot synopses. We consider dialogues from movie subtitles as they are close to daily human-to-human conversations (Zhang and Zhou, 2019). In addition, the movie synopses include rich narrative information,

which is helpful for dialogue understanding. After data collection and strict quality control, we obtain a dataset with 243K (dialogue, narrative) pairs written in English. As the automatic data construction procedure is language-independent, it can be applied to low-resource languages as well.

We then pre-train a BART model (Lewis et al., 2020) on the constructed corpus with the proposed *learning-by-narrating* strategy, and evaluate it on four dialogue-based tasks that require understanding and summarization. In zero-shot settings, our pre-trained model outperforms the BART baseline by a large margin on both dialogue understanding (e.g., $+8.3\%$ on DREAM (Sun et al., 2019)) and summarization (e.g., 24.6% on SAMSum (Gliwa et al., 2019)), demonstrating the success of our approach.

5.2 DIANA: A Dialogue-Narrative Corpus

In this section, we describe the procedure to create the dialogue-narrative parallel dataset.

5.2.1 Data Collection and Segmentation

We collect 47,050 English subtitles of movies and TV episodes released from Opensubtitle (Lison, Tiedemann and Kouylekov, 2018) and their corresponding synopses from online resources such as Wikipedia and TMDB. To link the subtitle and synopsis of the same movie or TV episode, we require a subtitle and a synopsis to have the same title and the release year, as well as a high overlap rate ($> 50\%$) on role names.

The subtitle and synopsis of a movie are too long for a PLM. To facilitate pre-training, we split both the subtitle and synopsis into smaller segments and align the related segments from each part into shorter (dialogue, narrative) pairs. We split subtitles using the time interval δ_T between utterances and split a synopsis into sentences. We set $\delta_T = 5s$.

5.2.2 Data Alignment

We aim to align the dialogue sessions $\{d_1, \ldots, d_n\}$ and narrative segments $\{s_1, \ldots, s_m\}$ with maximum global similarity to form (dialogue, narrative) pairs. For each dialogue session d_j, the goal is to find its corresponding narrative segment s_i.

Inspired by (Tapaswi, Bäuml and Stiefelhagen, 2015) in which the narrative in a synopsis follows the timeline of a movie or a TV episode, we develop a dynamic time-warping method to find the globally optimal alignment score. During aligning, some narrative segments contain information beyond the dialogue, so

Similarity Function	Accuracy
Jaccard	57.98
Rouge-1F	60.01
TF-IDF	67.20
TF-IDF normalized	**71.95**

Table 5.1: Alignment accuracy of different similarity measures on MovieNet.

they cannot be aligned to any dialogue session. We therefore allow our algorithm to skip at most k narrative segments during alignment searching:

$$\mathcal{A}(i,j) = \max_{0 \le k \le K+1} \mathcal{A}(i-k, j-1) + \mathcal{S}(s_i, d_j), \tag{5.1}$$

where $\mathcal{A}(i,j)$ denotes the optimal alignment score of the first i narrative segments and the first j dialogue sessions. $\mathcal{S}(s_i, d_j)$ is the TF-IDF similarity between s_i and d_j.

We compare the performance of three text similarity measures: Jaccard similarity, Rouge-1F, and TF-IDF. In consideration of time efficiency, we don't apply more advanced neural methods. We compare these similarity measures on MovieNet dataset (Huang et al., 2020), which provides a manual alignment between the segments of subtitles and synopses of 371 movies. [1] We evaluate the performance of each similarity measure by alignment accuracy, a.k.a, the percentage of dialogue sessions that are correctly aligned to the corresponding narrative segment. As shown in Table 5.1, TF-IDF performs best among all similarity measures. We also find that a narrative-wise L_2 normalization of the TF-IDF can further improve the alignment accuracy. It helps to penalize the similarity of (d_j, s_i) when s_i has high similarity with many dialogues (e.g., when s_i contains common words or protagonists' names.) We therefore choose the normalized TF-IDF as our similarity function. We further analyze the errors during alignment and find that 85.94% of errors happen because the dialogue session is aligned to the previous or next segment of the gold narrative segment. It indicates that most of the errors happen locally. Figure 5.2 shows an example from MovieNet, where the red line and the blue line indicate the gold alignment and the predicted alignment via normalized TF-IDF, respectively. It shows that the two lines are generally well overlapped except for some local discrepancies.

5.2.3 Quality Control

After data alignment, each narrative segment s_i can be aligned to multiple dialogues. To consider the local alignment errors, we also merge the aligned dialogues of s_{i-1} and s_{i+1} to the dialogues of s_i. Some

[1] We use MovieNet for test purposes only.

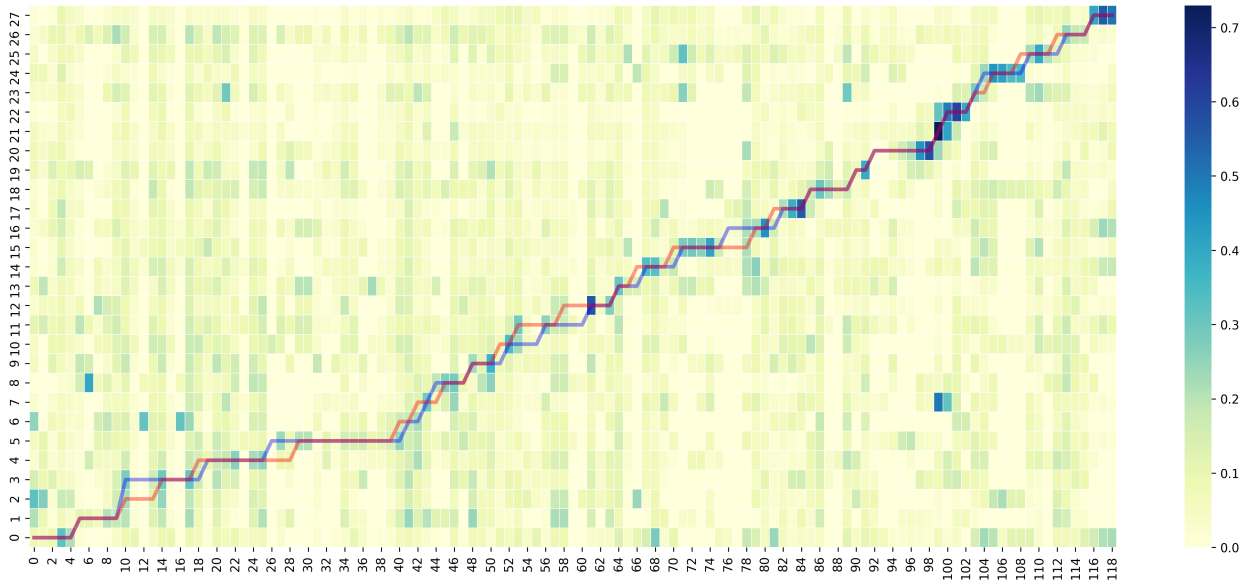

Figure 5.2: The Alignment of dialogues and narrative segments of a movie. X-axis and Y-axis are the ID of dialogue sessions and narrative segments, respectively. The variety of colors depicts the different similarity values between a dialogue session and a narrative segment. The blue line is the predicted alignment via normalized TF-IDF while the red line is the gold alignment.

of these dialogues may not be relevant to s_i. To select the relevant dialogues, we use a greedy method to incrementally select dialogues until the rouge-F score between the narrative and the selected dialogues doesn't increase. After selection, we concatenate the selected dialogues and preserve their relative position. We finally obtain around 1.5 Million (dialogue, narrative) pairs.

To further improve the quality of data, we filter out pairs where the dialogue and the narrative are irrelevant. To evaluate the relevance, we use two automatic measures: Coverage and Density (Grusky, Naaman and Artzi, 2018). Low Coverage and Density indicate that the narrative text is either too abstractive or irrelevant to the dialogue. We thus only select the pairs with Coverage > 0.5 and Density > 1. After this strict quality control, we obtain 243K (dialogue, narrative) pairs as the final DIANA dataset, which is a high-quality subset of the original dataset. The average length of the dialogue and the narrative are 58 tokens and 18 tokens, respectively.

To analyze what types of knowledge are included in DIANA, we randomly sample 100 instances and manually categorize the relation between dialogue and the corresponding narrative text into seven knowledge types. We show the percentage of each knowledge type in parentheses and in Figure 5.3 as well. The knowledge types are:

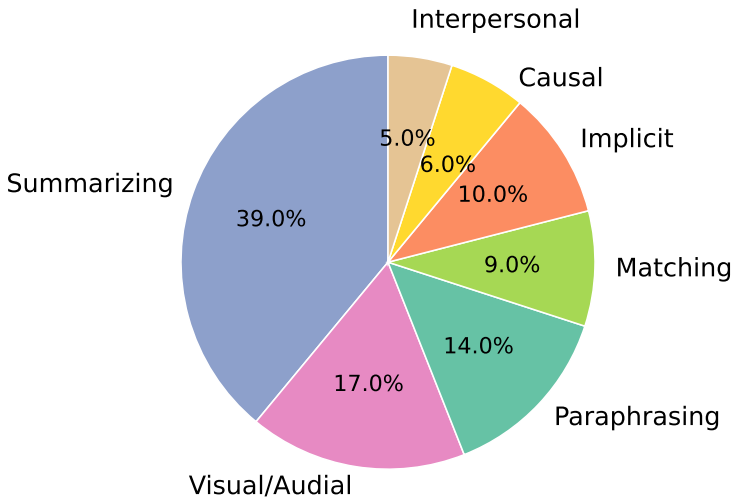

Figure 5.3: The knowledge type distribution in DIANA.

- **Summarizing** (39%): The narrative text summarizes multiple utterances as a concise statement to reflect the salient event or information of the dialogue.
- **Visual/Audial** (17%): The narrative text provides extra visual or audial information of the dialogue, such as the location of the dialogue, the speakers' actions, and ambient sounds.
- **Paraphrasing** (14%): The narrative text restates speakers' utterances using other words.
- **Text Matching** (9%): The narrative text is directly copied from the utterances of speakers.
- **Implicit** (10%): The narrative text provides extra information that is not explicitly mentioned in the dialogue.
- **Causal** (6%): The narrative text describes the cause and effect relationship between events.
- **Interpersonal** (5%): The narrative text reveals the relationships between speakers.

Among these knowledge types, *Summarizing* and *Visual/Audial* are the two most frequent ones. They are followed by *Paraphrasing* and *Text Matching*, which contribute to 23% in total. It also shows that narratives use paraphrasing more often than copying. Additionally, DIANA contains three higher-level knowledge types that require the awareness of real-world commonsense and more complicated inference such as implicit knowledge, causal relationships, and interpersonal relationships. The diverse knowledge types in DIANA indicate the benefit of this dataset for dialogue comprehension and other downstream tasks as well.

5.3 Pre-training: Learning-by-Narrating

During pre-training, we aim to inject the knowledge contained in DIANA into pre-trained models. One option is to ask the model to distinguish between a correct narrative and an incorrect narrative via a classification objective. However, it requires carefully designing additional non-trivial negative (dialogue, narrative) pairs. Therefore, we propose to directly generate a narrative text from the given dialogue by maximizing the generative probability:

$$p(\mathbf{y} \mid \mathbf{x}; \boldsymbol{\theta}) = \prod_{t=1}^{|\mathbf{y}|} p\left(y_t \mid y_{1:t-1}, \mathbf{x}; \boldsymbol{\theta}\right), \tag{5.2}$$

where $\mathbf{x}$ are dialogue texts and $\mathbf{y}$ are narrative texts.

There are two main advantages of using the generative objective. First, it can fully leverage the narrative information from each token of the narrative text with no need to construct negative pairs. Second, the pre-trained model can be directly applied to both generative and discriminative downstream tasks without further fine-tuning. For discriminative tasks, we calculate the probability of each candidate according to Equation 5.2 and choose the most probable candidate as the predicted answer.

5.4 Experiments

In this section, we evaluate the performance of the pre-trained model on four downstream tasks that require dialogue understanding or summarization.

5.4.1 Setting

We use BART, a state-of-the-art sequence-to-sequence model, as our baseline model.[2] We use its released checkpoint and further pre-train the model on DIANA. During pre-training, we concatenate the utterances as the input and update the parameters to maximize the probability of the corresponding narrative. We use Adam as the optimizer, and we set the learning rate and weight decay to 3×10^{-5} and 0.01, respectively. Following previous studies that suggest that a larger batch size helps pre-training, we set the batch size to 1,024 and pre-train the model for 1,000 steps.

[2]We also tried T5 and Pegasus in our early experiments but did not observe better performance compared with BART.

5.4.2 Tasks

We evaluate our model's ability of dialogue understanding and summarization on four downstream tasks. We adopt three dialogue understanding benchmarks: DREAM (Sun et al., 2019), PCMD (Ma, Jurczyk and Choi, 2018), and VLEP (Lei et al., 2020). **DREAM** aims to read a dialogue and select the correct answer from options of a dialogue-related question. To make the task similar to our pre-training task, we follow previous work (Chen, Choi and Durrett, 2021) to train a T5 model to convert each (question, answer) pair to a statement. **PCMD** is a passage completion task. Given a dialogue and a passage that describes the dialogue, a query is created by replacing a character mention with a variable x, and the model needs to recover the character mention. **VLEP** aims to select the most probable future event given the dialogue of the current event and two candidates for future events. For dialogue summarization, we test model performance on DialogSum (Chen et al., 2021) and SAMSum (Gliwa et al., 2019). **DialogSum** contains 13K face-to-face spoken dialogues, covering a diverse range of daily-life topics such as schooling, work, medication, shopping, leisure, and travel. The conversations mostly involve interactions between friends, colleagues, and service providers and customers. **SAMSum**, on the other hand, focuses on short conversations via messenger apps. It consists of 16K online chats, each accompanied by summaries that were annotated by language experts. The first three are discriminative tasks, and the last two are generative tasks. None of the source dialogues in these tasks are included in the DIANA dataset.

We evaluate the model performance on these tasks under the zero-shot setting. For discriminative tasks, we convert each test instance with K answer candidates as K (dialogue, narrative) pairs. Given the dialogue as input, we evaluate the conditional probability of each narrative according to Equation 5.2 and choose the most probable narrative as the predicted answer. We use accuracy (ACC) as the evaluation metric for understanding tasks and ROUGE for the summarization tasks.

We compare our pre-trained model (Narrator) with strong pre-trained baselines such as GPT-2, RoBERTa, and BART. To investigate the impact of the pre-training objective, we compare with 1) BART-DIAL-DE: the original BART de-noising objective, which is trained on the dialogue part of DIANA; and 2) BART-CNN-CLS: a classification objective, which is trained using the CNNDM dataset (See, Liu and Manning, 2017) to distinguish between positive and negative summaries based on the documents. Negative summaries are obtained from DocNLI (Yin, Radev and Xiong, 2021) by replacing the words, entities, and sentences of positive summaries. We also investigate the quality of DIANA by comparing it with two large summarization

	Data	Task	DREAM ACC	PCMD ACC	VLEP ACC
BART-FT	-	-	62.56	75.89	65.07
GPT-2	-	-	41.99	45.02	54.58
RoBERTa	-	-	45.22	46.25	52.28
BART	-	-	45.07	46.07	54.26
	DIAL	DE	46.69	47.34	55.98
	CNN	CLS	50.46	49.27	55.53
	CNN	GEN	52.72	45.34	58.13
	CRD3	GEN	52.96	45.71	57.12
Narrator	DIANA	GEN	**53.41**	**54.88**	**58.90**

(a) Results on three dialogue comprehension tasks: DREAM, PCMD, and VLEP. For models that require further pre-training, we list the corresponding pre-training dataset and task.

	Data	Task	DialogSum			SAMSum		
			R1	R2	RL	R1	R2	RL
BART-FT	-	-	47.28	21.18	44.83	49.18	24.47	47.12
GPT-2	-	-	11.86	1.50	8.64	10.83	0.74	11.68
BART	-	-	23.41	9.56	17.44	29.92	9.58	28.54
	DIAL	DE	24.18	9.58	18.31	30.08	9.52	29.36
	CNN	GEN	29.73	12.07	23.95	31.33	9.08	28.03
	CRD3	GEN	29.63	12.28	23.97	27.07	9.09	27.64
Narrator	DIANA	GEN	**34.72**	**15.68**	**29.20**	**37.27**	**13.23**	**36.12**

(b) Results on two dialogue summarization tasks: DialogSum and SAMSum. Since we create abstractive summaries, we remove the non-generative baselines such as RoBERTa and BART-CNN-CLS.

Table 5.2: Results on dialogue comprehension and summarization tasks.

datasets: CNNDM and CRD3 (Rameshkumar and Bailey, 2020). We pre-train BART to generate the summaries of these datasets from the corresponding documents and refer to the models as BART-CNN-GEN and BART-CRD3-GEN. Besides the zero-shot models, we list the supervised results finetuned on BART (BART-FT) as a reference for the upper bound.

5.4.3 Results

Results for dialogue comprehension and summarization are shown in Table 5.2a and Table 5.2b, respectively. Our observations are as follows. (i) When compared with vanilla PLMs, Narrator outperforms GPT-2, RoBERTa, and BART. For example, on SAMSum task, Narrator outperforms BART by 7.35 on Rouge-1 and 7.58 on Rouge-L. It demonstrates that the learning-by-narrating pre-training objective can improve the model's ability of dialogue understanding and summarization. (ii) When compared with different pre-training

SAMSum Example:
Anne: You were right, he was lying to me
Irene: Oh no, what happened?
Jane: who? that Mark guy?
Anne: yeah, he told me he's 30, today I saw his passport he's 40
Irene: You sure it's so important?
Anne: he lied to me Irene.

Summary:
CNN: Anne: You were right, he was lying to me Irene: Oh no, what happened? Jane: who? that Mark guy?
CRD3: Jester: Oh no, what happened? I told her he's 30, today I saw his passport he's 40.
Ours: Jane tells Anne that Mark told her he was 30 and that he's 40.
Reference: Mark lied to Anne about his age . Mark is 40 .

DialogSum Example:
#Person1#: Oh dear, my weight has gone up again.
#Person2#: I am not surprised, you eat too much.
#Person1#: And I suppose sitting at the desk all day at the office doesn't help.
#Person2#: No, I wouldn't think so.
#Person1#: I do wish I could lose weight.
#Person2#: Well, why don't you go on a diet?
#Person1#: I've tried diets before but they've never worked.
#Person2#: Perhaps you should exercise more. Why don't you go to an exercise class.
#Person1#: Yes, maybe I should.

CNN: Oh dear, my weight has gone up again. I am not surprised, you eat too much.
CRD3: She says she's not surprised that her weight has gone up again, but sitting at the desk all day at the office doesn't help. Percy suggests that she go on a diet, but she's never tried it before.
Ours: She says she's tried diets before but they've never worked. He suggests she go to an exercise class.
Reference: #Person1# wants to lose weight. #Person2# suggests #Person1# take an exercise class to exercise more.

Table 5.3: Sample summaries generated by baseline models and our method. For each example, we show the original dialogue, the referenced summary, and the output summaries from BART-CNN-GEN, BART-CRD3-GEN, and BART-DIANA-GEN (Ours).

tasks, Narrator outperforms BART-DIAL-DE, and BART-CNN-GEN outperforms BART-CNN-CLS. This indicates that the narrative-guided generative objective is more effective than the de-noising objective and the discriminative objective. (iii) When compared with different pre-training data, Narrator achieves better performance on all tasks compared with BART-CNN-GEN and BART-CRD3-GEN, demonstrating that DIANA is a more helpful resource for dialogue understanding and summarization compared with other non-dialogue summarization datasets.

Table 5.3 presents two examples extracted from SAMSum and DialogSum, respectively. Each example includes the dialogue, the referenced summary, as well as the predicted summaries generated by baselines

Question Type	BART	Narrator
Paraphrase+Matching	58.4	66.1 (+7.7)
Reasoning	42.2	46.2 (+4.0)
Summary	51.1	53.4 (+2.3)
Logic	43.8	48.2 (+4.4)
Commonsense	37.8	41.9 (+4.1)
Arithmetic	23.8	23.8 (+0.0)

Table 5.4: Accuracy by question types on DREAM.

and our method. BART-CNN-GEN is trained on CNNDM, a news summarization dataset. It is known that news summaries tend to exhibit bias towards the lead sentences of the document. Consequently, the model trained on this dataset tends to replicate the initial utterances when generating summaries. On the other hand, BART-CRD3-GEN is trained on CRD3, a dialogue summarization dataset. While this model doesn't suffer from lead bias issues, it faces challenges in narrating the dialogue and faithfully summarizing the key information. In contrast, our method is trained on DIANA, which can effectively identify the salient information within the dialogue and provide summaries from a third-person point of view. The results demonstrate the efficacy of DIANA in dialogue summarization.

We further analyze what types of knowledge are enhanced during pre-training. To this end, we test Narrator on a subset of the DREAM test set, which includes annotated knowledge types released along with the DREAM dataset. As shown in Table 5.4, compared with the vanilla BART, Narrator achieves better performance on all knowledge types except Arithmetic, which is not covered in DIANA. The performance gain indicates that the narrative pre-training contributes the most to the knowledge related to paraphrasing and matching. It also benefits from other knowledge types that require various reasoning abilities such as commonsense reasoning and logic reasoning.

5.5 Analysis

The Diana dataset comprises movies and TV episodes spanning various genres, as illustrated in Figure 5.4. In this section, we examine how it influences the performance of the pre-trained Narrator model. To accomplish this, we choose ten major genres and sample 35,000 instances from each genre, resulting in ten distinct pre-training datasets. We re-train the Narrator model using these genre-specific subsets and evaluate its performance on both SAMSum and DialogSum datasets. The results are listed in Table 5.5.

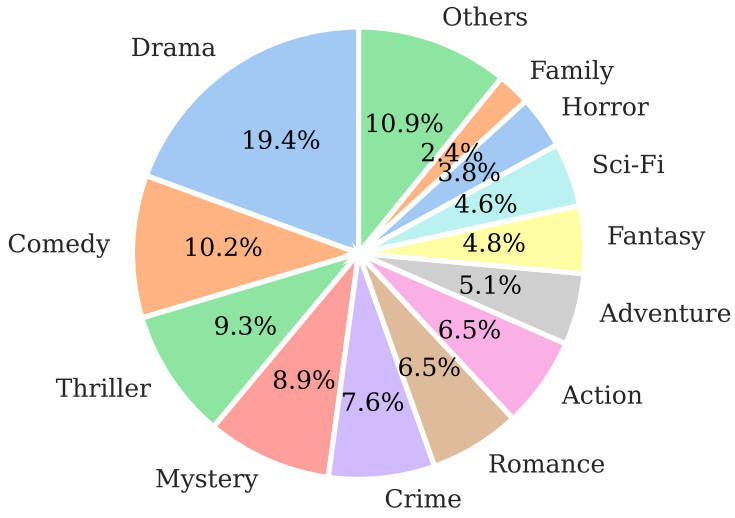

Figure 5.4: The genre distribution in DIANA.

Results show that there are substantial performance variations across different genre-specific models. For instance, the Mystery genre exhibits the poorest performance, whereas the Comedy genre achieves the highest performance. This performance gap is a 2.01 Rouge-1 score for DialogSum and a 3.3 Rouge-1 score for SAMSum. These results indicate the significant impact of genre on the model's overall performance.

To gain a deeper understanding of the performance disparities, we conduct an analysis of the statistical attributes of the genre-specific pre-training data. We examined the following four factors:

- Text Length: We consider metrics such as the number of utterances within the dialogue, the length of individual utterances, and the length of the summary. The statistics are presented in the left section of Table 5.6.

- Sentiment: We assessed the percentage of sentiment-related words within the dialogue using a sentiment lexicon. [3] We calculated these percentages through lexical matching and present them in the right section of Table 5.6.

- Word Frequency Usage: We compare the frequencies of words used in the pre-training dialogue with those in the test data and visualize the results in Figure 5.6.

[3]available at `https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon`

	DialogSum			SAMSum		
	R1	R2	RL	R1	R2	RL
Mystery	32.31	14.15	26.89	34.58	11.49	26.78
Crime	33.16	14.91	27.60	34.49	11.62	27.10
Action	33.12	14.67	27.95	35.57	12.31	28.36
Fantasy	32.39	13.38	26.30	36.60	12.42	28.73
Sci-Fi	32.87	14.44	27.11	36.30	12.31	28.36
Thriller	34.17	15.33	28.67	35.41	12.14	28.23
Drama	33.26	14.95	28.16	36.91	12.79	29.29
Adventure	33.37	14.48	27.98	36.90	12.31	29.54
Romance	33.65	15.08	28.23	37.62	13.20	30.10
Comedy	34.32	15.53	28.81	37.88	12.84	30.04

Table 5.5: Performance of genre-specific models on DialogSum and SAMSum.

Genre	Utt_num	Utt_len	Summ_len	Pos_Words(%)	Neg_Words(%)	Pos-Neg(%)
Mystery	16.1	7.42	3.85	2.47	2.66	-0.19
Crime	16.42	7.42	3.85	2.51	2.64	-0.12
Action	16.96	7.25	3.93	2.67	2.73	-0.06
Fantasy	17.88	7.02	3.86	2.82	2.79	0.03
Sci-Fi	17.12	7.21	3.96	2.60	2.73	-0.13
Thriller	16.78	7.28	3.86	2.49	2.67	-0.18
Drama	16.62	7.38	3.85	2.74	2.52	0.22
Adventure	16.98	7.28	3.95	2.93	2.64	0.30
Romance	17.22	7.30	3.83	2.98	2.36	0.62
Comedy	17.81	7.31	3.89	3.11	2.45	0.66

Table 5.6: Statistics of genre-specific pre-training datasets, including the average number of utterances (Utt_num), utterance length (Utt_len), summary length (Summ_len), rates of positive and negative words in utterances (Pos_Words% and Neg_Words%), and the rate difference (Pos-Neg%).

- Summary Abstractiveness: We draw a coverage-density distribution plot for each genre-specific pre-training dataset, illustrated in Figure 5.5.

The results indicate that there are no significant differences among genres with respect to text length and the coverage-density distribution. However, a strong correlation emerges between sentiment distribution and performance. In general, genres that achieve higher performance in summarization tasks tend to have a higher percentage of positive words and a lower percentage of negative ones. Moreover, these high-performing genres exhibit a greater degree of word usage similarity with the DialogSum and SAMSum datasets. To illustrate this, Figure 5.6 presents a comparison of word frequency between the pre-training datasets (depicted in green) and the test datasets (depicted in purple), which is a combination of DialogSum and SAMSum. We

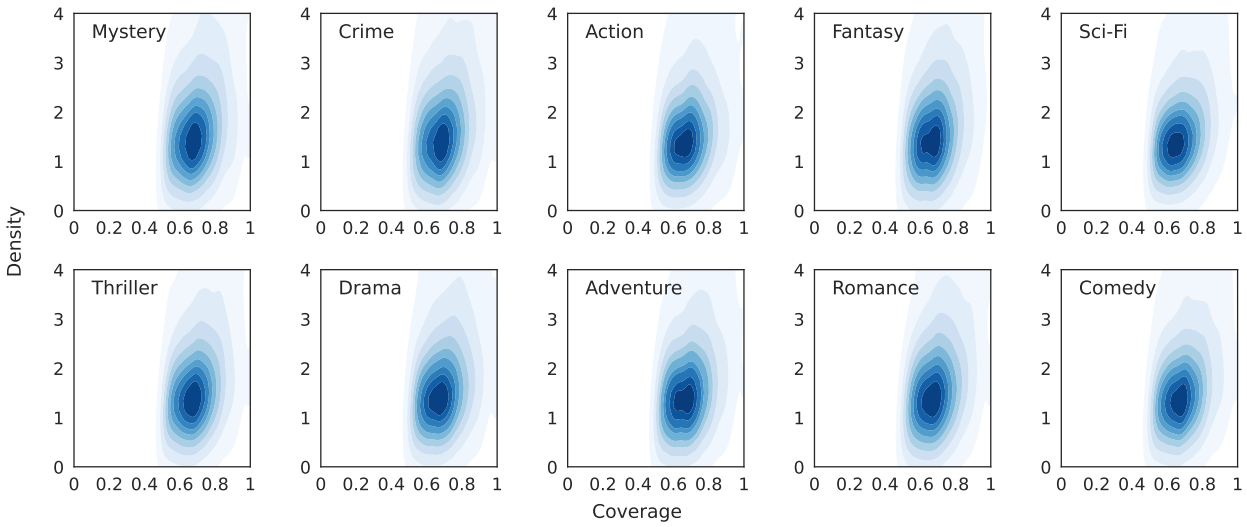

Figure 5.5: The coverage-density plot of genre-specific pre-training datasets.

choose four genres: Mystery, Crime, Romance, and Comedy. Words that cluster closely to the diagonal line indicate similar frequencies in both datasets, while those deviating significantly from this line signify varying frequencies between the two datasets. It shows that the word usage patterns of Romance and Comedy genres closely resemble those found in the two test datasets, differing only in a few colloquial terms (such as *Gonna*, *God*, and *huh*). In contrast, the Mystery and Crime genres contain more distinctive and genre-specific words such as *die*, *kill*, *blood*, *gun*, and *police*. These analyses suggest that one potential explanation for variations in genre performance lies in the alignment of word usage with the DialogSum and SAMSum datasets, such as the word frequency and sentiment polarity. Genres that exhibit closer correspondence in these aspects tend to perform better than others.

5.6 Conclusion

We propose a *learning-by-narrating* strategy to pre-train a zero-shot dialogue understanding and summarization model. We first construct a dialogue-narrative dataset named DIANA, which contains 243K (dialogue, narrative) pairs obtained by automatically aligning movie subtitles with their corresponding synopses. We then pre-train a model based on DIANA and evaluate its performance on four downstream tasks that require dialogue understanding or summarization abilities. Experiments show that our model outperforms strong pre-trained baselines, demonstrating that the learning-by-narrating strategy is a promising direction for dialogue understanding and summarization. We also hope that DIANA will promote future research in related areas.

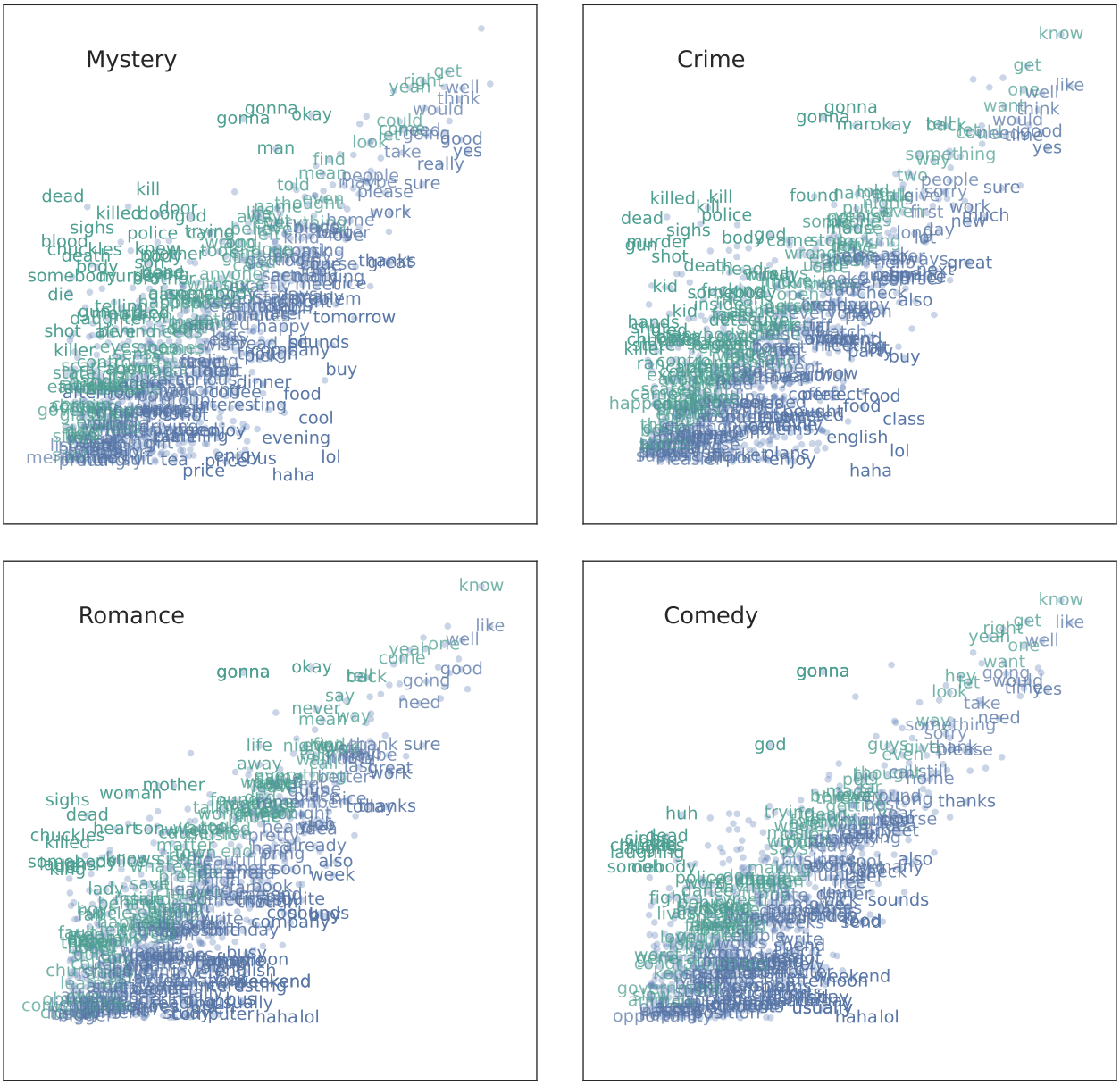

Figure 5.6: The word frequency comparison between the pre-training datasets (in green) and the test datasets (in purple), which is a combination of DialogSum and SAMSum. We choose four genres: Mystery, Crime, Romance, and Comedy. Words that cluster closely to the diagonal line indicate similar frequencies in both datasets, while those deviating significantly from this line signify varying frequencies between the two datasets.

CHAPTER 6: HARNESSING AUTOMATIC DATA PAIRING FOR ABSTRACTIVE NARRATIVE SUMMARIZATION

In this chapter, we shift our attention from dialogue summarization to narrative summarization. We introduce NARRASUM, an automatically collected large-scale narrative summarization dataset. It contains 122K narrative documents collected from plot descriptions of movies and TV episodes with diverse genres, and their corresponding abstractive summaries. Experiments show that there is a large performance gap between humans and the state-of-the-art summarization models on NARRASUM.

6.1 Introduction

A narrative is a story (e.g., a novel or a movie) composed of events and characters (Prince, 1973). *Narrative summarization* aims to produce a distilled version of a narrative, either extractively or abstractively, to contain its most salient events and major characters (Lehnert, 1981). This ability is especially crucial for the understanding of narratives, and in general, the understanding of human behaviors and beliefs (Piper, So and Bamman, 2021). Practically, a summary of a narrative can enable a reader to quickly discern the key points, which is useful in real-world scenarios such as content recommendations and advertisements.

While text summarization has been explored for over decades, most existing studies focus on summarizing news (Consortium and Company, 2008; Nallapati et al., 2016; Narayan, Cohen and Lapata, 2018) or structured documents (e.g., scientific papers (Gidiotis and Tsoumakas, 2019; Cohan et al., 2018)). These documents have specific writing styles. For instance, news is organized such that the first few sentences convey the most important information (Hicks et al., 2016). Scientific papers usually follow a standard structure with a few sections contributing the most to the summary (Gidiotis and Tsoumakas, 2020). It has been demonstrated that many summarization models, including recent ones, heavily rely on these structural clues (Kedzie, McKeown and Daumé III, 2018; Zhong et al., 2019; Zhao, Huang, Basu Roy Chowdhury, Chandrasekaran, McKeown and Chaturvedi, 2022). However, a typical narrative does not contain such structural cues. This suggests that a narrative summarization model has to understand the entire narrative to identify the salient events and characters. While some recent summarization tasks also require understanding an entire document, they

Document: (https://bigbangtheory.fandom.com/wiki/The_Big_Bran_Hypothesis)

Setting their dinner of Thai food, Sheldon gives the group a lecture of the use of the fork in Thai history. A little later, Penny talks with Leonard in the hallway about her work at The Cheesecake Factory. She then asks Leonard to sign for a piece of furniture while she is out. […]

It turns out the furniture is bigger than they had expected. The delivery man does not help them, so Leonard and Sheldon are forced to carry it up the stairs the themselves since the elevator doesn't work. Sheldon's only idea involves using a Green Lantern power ring. Finally, they eventually succeed in getting it up the stairs to her apartment. While there, Sheldon sees that Penny's apartment is a complete mess and insists on tidying up. […]

Leonard get up the next morning and Sheldon tells him that he slept well. Leonard remarks that a well known folk cure for insomnia is to break into your neighbor's apartment and clean. Sheldon asks if that was sarcasm. Penny awakens to find out that her apartment in a well ordered state and screams about those geeky bastards. Penny charges into Sheldon and Leonard's apartment in a fit of rage about them coming into her place while she was sleeping. She demands her key back. […]

Later, Penny runs into Raj in the hallway and talks to him about being upset over what happened (although he doesn't reply as he has selective mutism). Penny decides to forgive them while Raj was thinking; "Boy, her hair smells nice" and "Maybe my mother was right. Maybe I should marry an Indian girl. We would have the same cultural background and she could sing the same lullabies my mother sang to me". Penny then hugs Raj, much to his surprise. […]

Summary: (https://en.wikipedia.org/wiki/The_Big_Bang_Theory_(season_1)#ep2)

When Sheldon and Leonard drop off a box of flat pack furniture that came for Penny, Sheldon is deeply disturbed at how messy and disorganized her apartment is. Later that night, while Penny sleeps, the obsessive-compulsive Sheldon, unable to sleep, sneaks into her apartment to organize and clean it. Leonard finds out and reluctantly helps him. The next morning, Penny is furious to discover they had been in her apartment. Sheldon tries to apologize to Penny but fails by remarking that Leonard is a "gentle and thorough lover". Later, Penny encounters Raj in the hallway. Though he cannot talk to Penny, she calms down whilst telling him about the issue, reasoning the guys were just trying to help her, and hugs Raj. Then Leonard apologizes, prompting Penny to forgive and hug him.

Figure 6.1: Example of the narrative summarization task. The input is a narrative text (denoted by “Document”, pictures are not included), and the output is a summary containing its salient events and characters.

focus on conversational domains such as dialogues (Gliwa et al., 2019), emails (Zhang, Celikyilmaz, Gao and Bansal, 2021), and meetings (Zhong et al., 2021). Narratives are different from those genres in nature and are understudied.

Understanding an entire narrative faces unique challenges. A narrative organizes the story into a sequence of events (i.e., plot) in a chronological and causal order (Forster, 1985). Events unfold due to the actions of characters and other event participants, or external forces in stories (Mani, 2012). To identify the salient events, a model needs to understand both **plot** and **characters**. From the plot's perspective, the model needs to understand the causal and temporal relationships between events, as well as how the plot develops from the beginning to the end (Freytag, 1908). From the character's perspective, the model needs to understand the characters' profiles (e.g., personalities, roles, and interpersonal relationships), and how their desires and actions drive the story forward.

Figure 6.1 illustrates the importance of understanding the entire narrative for summarization. In this example, the main event is "*Sheldon cleans Penny's apartment and gets Leonard in trouble*", which is included in the summary. The side event "*Penny speaks to Raj and forgives Leonard*" is also included since it is the consequence and ending of the main event. Whereas, *"Sheldon gives a lecture of fork"* is not included as it does not impact the development of the plot. Besides the main events, the summary also explains Sheldon's motivation to clean the apartment.

A large-scale high-quality dataset is essential to promote research on this topic. Unfortunately, different from other domains, such as news and scientific papers, where the document and summary can be found from the same data source, narrative documents and their corresponding summaries are usually spread in separate sources. Previous studies collect document-summary pairs of narrative by either creating summaries manually (Ouyang, Chang and McKeown, 2017) or matching titles between documents and summaries followed by a manual inspection (Ladhak et al., 2020; Kryscinski et al., 2022), making it challenging to enlarge the resulting datasets.

In this work we propose an automatic data construction framework to build a narrative summarization dataset with both large scale and high quality. Specifically, we first collect narratives from plot descriptions of movies or TV episodes through online resources. We choose the plot description because it describes the overall narrative of the movie or TV episode, including the story arcs and major characters. This source is also widely used in narrative-related studies (Linebarger and Piotrowski, 2009; Bamman, O'Connor and Smith, 2013; Papalampidi, Keller and Lapata, 2019; Xiong et al., 2019). After data collection, we build an **align-and-verify** pipeline to automatically align plot descriptions of the same movie or TV episodes from different sources. Finally, we construct document-summary pairs by treating the long plot description as the document to be summarized and the shorter one (of the same movie or TV episode) as the corresponding

summary. After filtering out low-quality document-summary pairs, we build **NARRASUM**, a large-scale dataset that contains around 122K **narra**tive document-**sum**mary pairs in English. Our data construction framework is generic and thus can potentially be applied to other languages as well.

To gauge the feasibility of NARRASUM for the narrative summarization task, we explore different characteristics of this dataset. We observe that compared with other summarization datasets, the narratives in NARRASUM are of diverse genres, and the summaries are more abstractive and of varying lengths. Furthermore, rather than focusing on a particular part of the document (as in other summarization datasets), the summaries in NARRASUM are designed to cover the entire narratives. It brings new challenges to current summarization methods.

We investigate the performance of several strong baselines and state-of-the-art summarization models on NARRASUM. Results show that there is a large gap between human and machine performance in various dimensions, demonstrating that narrative summarization is a challenging task.

The contributions of this paper are four-fold:

- We propose an automatic data construction framework to build a large-scale, high-quality narrative summarization dataset.
- We release the largest narrative summarization dataset to date named NARRASUM, with detailed data analysis;
- We investigate the performance of recent summarization models on NARRASUM;
- We perform a thorough analysis of the models to point out the challenges and several promising directions.

6.2 Data Construction

We propose an automatic data construction framework to create a narrative summarization dataset. To this end, we first collect plot descriptions of movies and TV episodes from multiple resources as narratives (Section 6.2.1). We then align plot descriptions in these resources that refer to the same movie or TV episode (Section 6.2.2). Finally, we filter the aligned data to construct high-quality document-summary pairs. (Section 6.2.3). We describe the details of each step as follows.

6.2.1 Data Collection

We collect plot descriptions of movies and TV episodes from various movie websites and online encyclopedias such as Wikipedia,[1] Fandom,[2] IMDB, [3] TVDB, [4] and TMDB. [5] Note that while we use movie/TV plot descriptions as a source of narrative text, our goal is not to summarize movies and TV episodes themselves but rather to study the task of narrative summarization in a broader sense. Tasks of movie/TV summarization have been addressed by other datasets such as Scriptbase (Gorinski and Lapata, 2015), Screenplay (Papalampidi et al., 2020), and SummScreen (Chen et al., 2022). Those works focus more on summarizing screenplays, which describe the movements, actions, expressions, and dialogue of the characters in a specific structure and format. Compared with general narrative summarization, screenplay summarization presents a different set of challenges such as scene understanding and dialog parsing. Plot descriptions, on the other hand, describe the movie stories from a third-person point of view and present a different set of challenges as we described in Section 6.1.

To collect plot descriptions, we parse web pages of movies or TV episodes based on HTML tags and use heuristics to match keywords (e.g., *Synopsis*, *Summary*, and *Plot*) that are related to the plot. We then extract the texts under these sections as plot descriptions of the corresponding movies or TV episodes. Besides the plot descriptions, we also collect the meta information of movies or TV episodes such as their title, air date, director(s), and writer(s), which is used for data alignment.

6.2.2 Data Alignment

After data collection, we align the web pages that are from different websites but refer to the same movie or TV episode. It is a challenging task due to the ambiguity in natural language. For example, a single movie may have different surface forms of the title (e.g., *Avengers 4* and *Avengers: Endgame*), while those with the same title may refer to different movies (e.g., *Bad Company* may refer to fourteen movies.) Similar ambiguity issues arise when aligning air dates or names of crew members. Also, meta-information might be missing or incorrect due to the editing or parsing mistakes of web pages. To address these challenges, we propose an

[1] `https://www.wikipedia.org/`.

[2] `https://www.fandom.com/`.

[3] `https://www.imdb.com/`.

[4] `https://thetvdb.com/`.

[5] `https://www.themoviedb.org/`.

align-and-verify pipeline. It first aligns movie or TV episodes via fuzzy meta-information matching, which encourages high recall. Then, we use a verifier with high precision to re-check the aligned pairs and filter out the pairs with low confidence. We describe the details of this pipeline as follows.

During the **alignment** stage, we apply several heuristics for fuzzy meta-information matching. To align movies, we first normalize movie titles by removing non-alphanumeric characters, stopwords, and subtitles. We then collect the movie pairs where the Levenshtein distance between the normalized titles is less than a threshold.[6] Besides the title match, we also require the two movies to have the same air date or a partial overlap on directors or writers when such information is available. The ambiguity in titles of TV episodes is more severe than that of movies. To align TV episodes, we apply similar heuristics and further require the two episodes to belong to the same TV show.

During the **verification** stage, we improve the precision of alignment by comparing the aligned plot descriptions. Specifically, we train a classifier to take as input the concatenation of two plot descriptions to predict if they should be aligned. To train such a classifier, we first build a dataset with balanced positive aligned pairs and negative pairs. The positive pairs are a subset of heuristically aligned pairs where there is an link in one web page (e.g., "External links" in Wikipedia) pointing to the web page of the same movie or TV episode in the other website. Such links are edited by humans and are commonly used in entity linking (Shen, Wang and Han, 2014). Negative pairs are randomly sampled from different movies of the same movie series or different episodes of the same TV show. Negative pairs sampled by this strategy usually share a similar set of characters and background setting, preventing the model from relying on surface-level cues to solve the task.

Based on the data sampling method, we collected a large-scale balanced dataset with 60K positive pairs and 60K negative pairs. We then split the dataset into train/validation/test subsets with the ratio of $80\%/10\%/10\%$. We train a RoBERTa-base (Liu et al., 2019) classifier on this dataset and it achieves an accuracy of 97.13% on the test set, indicating that this model can serve as a reliable verifier to improve the precision of data alignment. We employ this classifier to further verify the heuristically aligned plot descriptions and filter out those where the predicted log-odds is smaller than 1. Finally, we obtain 2.6 million aligned plot description pairs.

[6]We set the threshold to be $0.2 \times l$, where l is the maximum length of the two titles. All thresholds in this section were chosen by experimenting with different values and manually analyzing the quality of a subset of the data.

6.2.3 Document-Summary Pairing

After obtaining the aligned plot description pairs, we regard the longer plot description as the document and the shorter one as the corresponding summary. However, not all pairs are of good quality for summarization. We identify three major issues compromising the quality and remove the relatively low-quality pairs from the final dataset.

First, the summary may contain hallucinated content that might not be included in the document. Similar to (Ladhak et al., 2020), we observe that hallucination is less common in plot description pairs with a noticeable difference in length. We therefore require the length of the summary to be shorter than half of the document to be summarized. We also calculate the semantic matching score between a summary and a document, and then remove the pairs with low scores. We adopt two scores. The first is the Rouge-1 Precision between the summary and the document. The second is the entailment probability between the summary and the document obtained from DocNLI (Yin, Radev and Xiong, 2021), a document-level NLI model. We add up the two scores, rank the instances accordingly, and remove the 3% document-summary pairs with the lowest score.

Second, sometimes the content in the shorter plot description is directly copied from the longer plot description. To create an abstractive summarization dataset, we use ROUGE-2 Precision (Lin, 2004) between the document and the summary to reflect whether the content of the summary is copied from the document, and remove the pairs where the ROUGE-2 Precision is larger than 0.5.

Third, a plot description may only describe part of the entire narrative such as a trailer but does not necessarily summarize the narrative. To filter out these cases, we set the minimum length of documents and summaries to make sure that they contain enough information. [7] We also extract oracle extractive summaries from the original document using the method proposed by Liu and Lapata (2019*b*). We remove the instances where less than 30% content of the oracle extractive summaries are from either the first half or the second half of the document.

After applying these filtering strategies, we obtain the final version of NARRASUM. It contains 122K aligned document-summary pairs, which is a high-quality subset (3.8%) of the original aligned pairs. We

[7] For movies, we set the minimum length of documents and summaries as 200 and 100. For TV episodes, we set the minimum length as 100 and 50.

Datasets	Domain	Size	L-doc	L-sum	Ratio
CNNDM	News	312K	781	56	13.9
XSum	News	227K	431	20	21.5
arXiv	Sci-Paper	215K	4,938	220	22.4
PubMed	Sci-Paper	133K	3,016	203	14.9
NovelChap	Novel	8K	5,165	372	13.9
BookSum	Novel	12K	5,102	505	10.1
NARRASUM	Movie/TV	122K	786	147	5.3

Table 6.1: Comparison between NARRASUM and other datasets accroding to the domain, size, document length, summary length, and compression ratio.

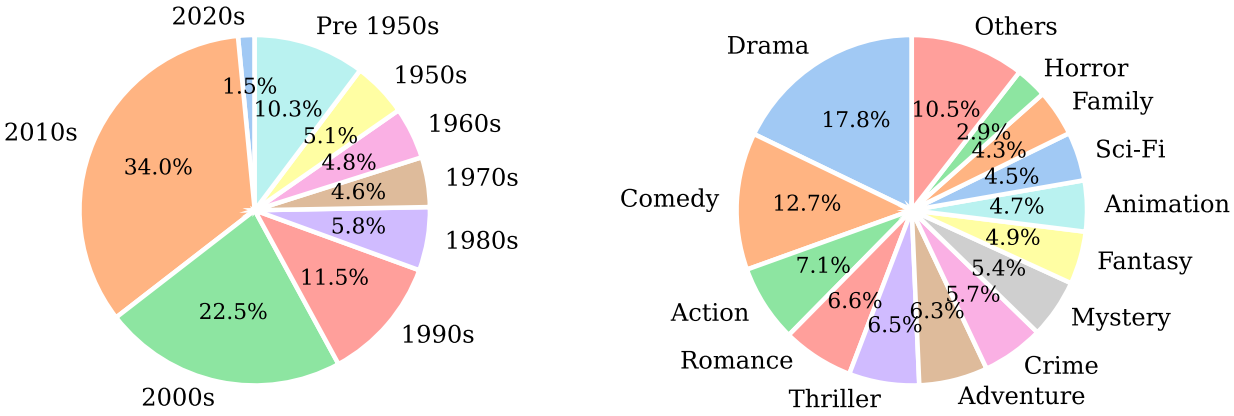

Figure 6.2: Distribution of production years and genres in NARRASUM.

split the dataset into training (90%), validation (5%), and testing (5%) sets at the title level in order to avoid data leakage and undesirable overlap between training and validation or test sets.

6.3 Data Analysis

This section provides basic statistics of NARRASUM. We then analyze the dataset in terms of the distribution of salient information and abstractiveness of summaries. Finally, we conduct a human assessment to evaluate the quality of NARRASUM.

6.3.1 Data Statistics

We compare NARRASUM with six datasets from different domains such as news, scientific papers, and narratives. These include CNN DailyMail (CNNDM) (See, Liu and Manning, 2017), XSum (Narayan, Cohen

Datasets	% of novel n-grams in summary			
	1-grams	2-grams	3-grams	4-grams
CNN/DM	17.00	53.91	71.98	80.29
XSum	35.76	83.45	95.50	98.49
Pubmed	18.53	48.23	68.28	78.39
NARRASUM	47.78	81.86	94.96	98.00

Table 6.2: Comparison of novel n-grams between NARRASUM and other summarization datasets.

and Lapata, 2018), ArXiv (Cohan et al., 2018), PubMed (Cohan et al., 2018), NovelChapter (Ladhak et al., 2020), and BookSum (Kryscinski et al., 2022). The comparison of statistics is shown in Table 6.1.

NARRASUM contains 122K instances from 22.8K unique movies and 28.5K unique TV episodes, which is ten times larger than the previous largest narrative summarization dataset. We provide the distribution of production years and genres of these movies or TV series in Figure 6.2, which illustrates that NARRASUM spans a wide time period and contains a broad range of genres. The average length of documents and summaries are 785.97 and 147.06 tokens, and the average compression ratio is 5.34. Most of the documents in NARRASUM are longer than 512 tokens, which is the maximum input length of many pre-trained language models. However, the average length of documents in NARRASUM is still shorter than that of a typical novel chapter ($\sim$5K). This requires the models to process long, but not prohibitively long, inputs while exposing them to the challenges of narrative summarization.

6.3.2 Summary Characteristics

Different from news articles, salient information in a narrative spreads across the entire text. To verify whether NARRASUM's summaries have this property, we first check the **distribution of the salient information** in the documents. Similar to Kim, Kim and Kim (2019), we use bi-grams of summary text to represent the salient content of the narrative and then obtain their normalized positions in the documents. Figure 6.3(a) shows the probability density distribution of the positions of the salient information. We compare the distribution of NARRASUM with CNNDM, XSum, and PubMed. Figure 6.3(a) indicates that while the salient information of CNNDM and PubMed are concentrated at certain parts of the document, the salient information of NARRASUM is more uniformly distributed over the entire document. It supports our claim that the summarization of NARRASUM requires an understanding of the entire document. There is no lead bias in XSum because the first sentence of the document is removed and is regarded as the summary. It

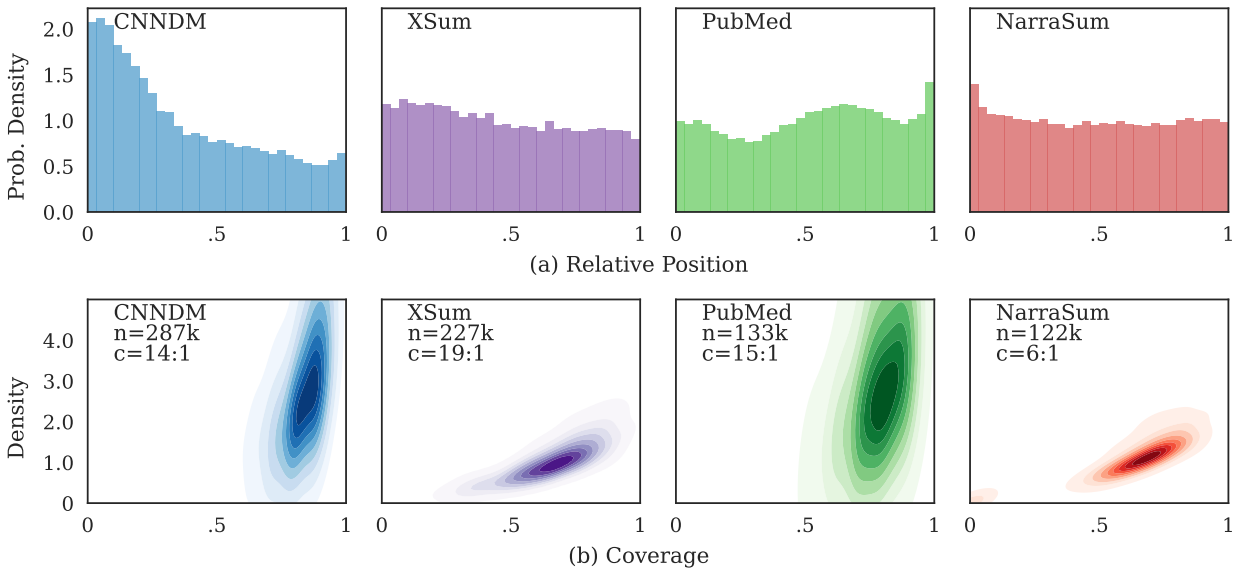

Figure 6.3: The upper figures show the relative positions of bi-grams of the gold summary in the document. The summary content of NARRASUM is more uniformly distributed over the entire document. The lower figures show the Coverage-Density plots. Compared with CNNDM and PubMed, the summary abstractiveness of NARRASUM is more close to XSum.

further demonstrates that the first sentence of a news document is enough to summarize the entire document. The section-wise bias in scientific papers is discussed by Gidiotis and Tsoumakas (2020).

Next, we measure the **abstractiveness of summaries** in NARRASUM. To this end, we calculate the Coverage and Density of each summary as suggested by Grusky, Naaman and Artzi (2018). Lower Coverage and Density scores indicate that the summary is more abstractive. The distribution is shown in Figure 6.3(b). The comparison shows that the summaries of NARRASUM are more abstractive than CNNDM and PubMed while being similar to XSum, the most abstractive dataset for news summarization.

We also report the percentage of novel n-grams that are included in the summary but not in the document. A higher percentage of novel n-grams implies a more abstractive summary. As shown in Table 6.2, the percentage of novel n-grams in NARRASUM is higher than CNNDM and PubMed, and is similar to XSum. This is in line with our observation from the Coverage-Density plot (Figure 6.3(b)). The difference is that XSum is a news summarization dataset with short summaries (one sentence). NARRASUM is a narrative summarization dataset, where the summaries are of varying length.

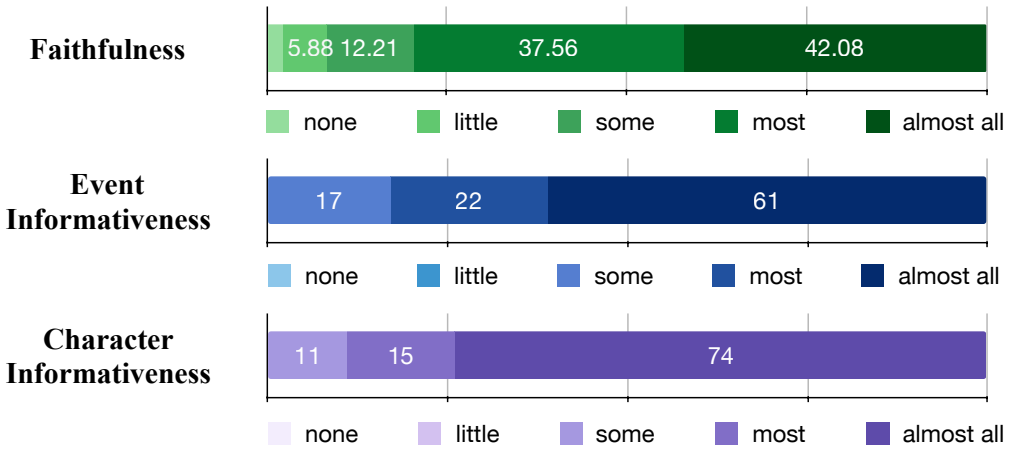

Figure 6.4: Human assessment results of the quality of NARRASUM.

6.3.3 Quality Assessment

We further conduct a human evaluation to better assess the quality of the NARRASUM. We randomly select 100 instances from the test set. For each instance, we ask three workers on Amazon Mechanical Turk to evaluate the summary in terms of *faithfulness* and *informativeness*. For faithfulness, we show annotators each summary sentence and ask them to evaluate how much of the information in this summary sentence is presented in the document. This is a precision-oriented measure and is commonly used for summary evaluation (Lu, Dong and Charlin, 2020). For informativeness, we ask annotators to first identify the most salient events and major characters from the document and then evaluate how much of that is covered by the summary. This is a recall-oriented measure. Both human evaluations are collected on a Likert scale of 1-5 (1 means "none", and 5 means "almost all").

To control the annotation quality, we require human judges to be in the United States, and have more than 1,000 HITs approved with an approval rate higher than 98%. We randomly check the annotation results and block the human judges who continually provide low-quality annotations. Human judges were paid a wage rate of $12 per hour, which is higher than the local minimum wage rate.

Figure 6.4 shows the distributions of human evaluation results. It shows that 80% of content in the summary is faithful to the document. For informativeness, 83% and 89% of summaries cover most of the salient events and characters, respectively. It demonstrates that NARRASUM is of high quality in both faithfulness and informativeness, and can foster further research on narrative summarization.

6.4 Baseline Models

We investigate the performance of several baselines and state-of-the-art neural summarization models on NARRASUM. We include both extractive and abstractive models. For extractive models, we use the following methods:

RANDOM selects n sentences from the document randomly.

LEAD selects the top-n sentences from the document to compose the summary. This is a strong baseline for news summarization.

TEXTRANK (Mihalcea and Tarau, 2004) is a graph-based extractive summarization model based on PageRank (Brin and Page, 1998) in a graph representation of sentences.

LEXRANK (Erkan and Radev, 2004) is another graph-based extractive summarization model based on eigenvector centrality .

HSG (Wang et al., 2020) is a heterogeneous graph-based neural extractive summarization model that uses word co-occurrence to enhance sentence contextual representation.

PRESUMM (Liu and Lapata, 2019*b*) relies on a pre-trained language model to enhance the sentence representation during text encoding and extractive summarization. We choose **BERT** (Devlin et al., 2019), **ROBERTA** (Liu et al., 2019), and **LONGFORMER** (Beltagy, Peters and Cohan, 2020) as the pre-trained models. BERT and RoBERTa limit the input length to be shorter than 512 tokens, while Longformer can accept up to 4,096 tokens.

For abstractive models, we use the following pre-trained sequence-to-sequence models: **BART** (Lewis et al., 2020), **T5** (Raffel et al., 2020), **PEGASUS** (Zhang et al., 2020*b*), and **LED** (Beltagy, Peters and Cohan, 2020). The input length of the first three models is limited to 512 (base version) or 1,024 (large version). LED uses Longformer as the encoder and therefore can accept up to 4,096 tokens as input.

6.5 Experiments

6.5.1 Settings

We conduct experiments with models described in Section 6.4 to evaluate their performances on NARRASUM. For extractive models, we follow the hyper-parameters of the original implementations. For abstractive models, we implement them using the Transformer library (Wolf et al., 2020). We fine-tune each model on

Model	**R-1**	**R-2**	**R-L**	**SC**
Extractive				
RAND	33.94	5.38	29.80	-
LEAD	35.11	6.71	30.82	-
LEXRANK	34.22	5.78	29.70	-
TEXTRANK	34.95	6.18	30.28	-
HSG	36.94	7.54	32.35	-
BERT-BASE	36.34	7.29	31.71	-
ROBERTA-BASE	36.47	7.31	31.80	-
LFORMER-BASE	**37.54***	**7.83***	**32.69***	-
ORACLE	42.42	11.44	36.65	-
Abstractive				
BART-BASE	35.81	7.49	31.72	65.19
T5-BASE	36.37	7.42	32.17	76.38
LED-BASE	37.32	8.14	33.05	62.63
BART-LARGE	36.80	8.20	32.62	**77.41***
T5-LARGE	37.67	8.11	**33.40**	74.14
PEGASUS-LARGE	36.97	7.93	32.64	75.23
LED-LARGE	**37.71**	**8.87***	33.34	66.91

Table 6.3: Summarization results evaluated on test set of NARRASUM over ROUGE 1 (R-1), ROUGE 2 (R-2), ROUGE L (R-L), and SummaC (SC). SC is only used to evaluate abstractive summaries as extractive summaries are faithful by design. We highlight the best scores separately for extractive and abstrative systems. * indicates a statistically significant difference compared with the second best score (bootstrap resampling, $p < 0.05$ (Koehn and Monz, 2006)).

the training set of NARRASUM with AdamW optimizer (Loshchilov and Hutter, 2019) and batch size of 64. We conduct a simple hyper-parameter search for the learning rate from $\{3e^{-4}, 1e^{-4}, 3e^{-5}\}$ based on the validation loss. We also adopt early stopping based on the validation loss to avoid overfitting. During inference, we use beam search with beam-size 5. Our model was trained on a single Quadro RTX 5000 GPU in up to 34 hours, depending on the model size.

Evaluation. We evaluate the generated summaries using ROUGE F_1 score.[8] We further include SummaC (Laban et al., 2022), an automatic measure for summary faithfulness. It achieves state-of-the-art on the benchmark of summary inconsistency detection, and is feasible to be applied to long input and output.

[8]https://github.com/google-research/google-research/tree/master/rouge.

6.5.2 Automatic Results

Table 6.3 shows the results on NARRASUM using extractive and abstractive summarization approaches.

Extractive Models. The supervised extractive methods outperform the unsupervised extractive methods (the first four models) on all measures by a large margin, indicating that NARRASUM can provide a strong supervision signal for identifying the salient information and creating the summary accordingly. PreSumm-BERT or PreSumm-Roberta models underperform HSG because these models have a maximum input length of 512 tokens whereas HSG can accept inputs with arbitrary length. Longformer achieves the best performance on extractive summarization by combining the advantage of pre-training and long document processing. However, there is still a large gap between Longformer's performance and the oracle upper-bound, indicating the challenges in narrative summarization.

Abstractive Models. Among these models, no particular model consistently outperforms others on all subsets. Larger models consistently outperform smaller models, which is inline with previous research. T5 outperforms BART on most Rouge scores, as they adopt summarization-specific pre-training objectives. LED outperforms other models on Rouge due to its ability to encode longer documents. This is consistent with the result of extractive summarization. However, LED performs worst on SummaC-based faithfulness evaluation. This indicates that though the model can process longer documents, understanding and faithfully summarizing lengthy texts is still challenging.

6.5.3 Human Evaluation

We further conduct a human evaluation on Amazon Mechanical Turk to better understand the models' behaviors and the challenges of this task. We randomly sample 100 instances from the test set and then evaluate the outputs of the best two systems (T5-Large and LED-Large) based on the following four dimensions.

- Fluency: whether or not the summary is grammatically correct and free of repetition;
- Faithfulness: whether or not the summary is faithful to the original document;
- Coherence: whether or not the plot of the narrative summary is logically coherent;
- Informativeness: whether or not the summary reflects the salient events and characters in the original document;

Model	T5-Large	LED-Large
Fluency	4.19	4.11
Faithfulness	3.34	3.23
Coherence	2.87	3.06
Informativeness	2.44	2.67

Table 6.4: Human evaluation of the generated summaries.

For each instance, we show annotators the original document and the generated summaries. We ask annotators to rate summaries using a 5-point Likert scale and report the average score over all instances. As shown in Table 6.4, while the pre-trained abstractive models are good at Fluency, they still struggle with other dimensions such as Faithfulness, Coherence, and Informativeness. It further indicates that narrative summarization is a challenging task for current models. In general, the summaries created by T5 are more fluent and faithful, while those created by LED are more coherent and informative.

Table 6.5 shows an example with the narrative document, gold summary, and predicted summaries. The narrative document is from Season 2, Episode 1 of *Zoey 101*, an American comedy-drama TV. This example shows that while the gold summary can faithfully cover the most salient information from the narrative document, summaries generated by machines contain some errors. Bart does not contain the information of "*Zoey returns to PAC*" and "*Dana will not return*". T5 fails to follow the causal and temporal relationships of events. The summary created by Pegasus is generally not coherent. The summary created by LED covers all important information but the writing is not fluent.

6.6 Analysis

We perform a series of analyses about the summary position and character consistency. For a fair comparison among models, we only choose test instances where the length of the document is shorter than the maximum input length of these models (1,024 tokens).

6.6.1 Analysis of Summary Position

A good narrative summary should preserve the original narrative structure that contains a start, middle, and ending of the narrative. To investigate this, we adopt the method in Kim, Kim and Kim (2019) to analyze the normalized position of summary bi-grams in the document, where 0 and 1 represent the start and ending of the document, respectively.

Document: It is the girls' second year at PCA. Dana has been accepted to a European Exchange Program, so Zoey and Nicole think will have their dorm room to themselves. However, Coco informs them they will be getting a new roommate. Zoey and Nicole go to the housing office hoping to choose a roommate, but their request is denied (After Nicole accidentally destroy's the secretary's perfume collection). When they return to their dorm, they find their new roommate, Lola Martinez, a goth-punk girl who disturbs them by drinking raw eggs and claiming to talk to the dead. As Lola's behavior grows more erratic, Zoey and Nicole are out of the dorm, but they later discover that Lola is not a goth-punk, but an aspiring actress who managed to fool them with her acting skills. Meanwhile, Michael and Chase have to deal with Logan, whose father has given him a huge entertainment center for their dorm room—which turns out to be a huge headache.
Gold Summary: Zoey and Nicole return to PCA for the second year. But they soon find out Dana will not be returning, and a very strange roommate, Lola, has come to take her place. The year gets to a bad start for Chase and Michael, because of Logan's very loud gifts from his dad.
Bart-large: **[Missing *Zoey returns to PAC* and *DANA will not return*]** Zoey and Nicole get a new roommate, Lola Martinez, a goth-punk girl who disturbs them by drinking raw eggs and claiming to talk to the dead. Meanwhile, Michael and Chase have to deal with Logan, whose father has given him a huge entertainment center for their dorm room–which turns out to be a huge headache.
T5-Large: Zoey and Nicole get a new roommate, Lola Martinez, who turns out to be a goth-punk. Meanwhile, Michael and Chase have to deal with Logan, whose father has given him a huge entertainment center for their dorm room. **[Fail to follow the causal and temporal relationship]** Meanwhile, Dana is accepted to a European exchange program, so the girls think they'll have their own room, but Coco tells them that they'll be getting a roommate.
Pegasus-Large: Zoey and Nicole's first year at PCA is coming to an end, and they're looking forward to living in their own dorm room. However, Coco informs them that Dana has been accepted to a European exchange program, so Zoey and Nicole think **[Not logically coherent]** they will have their own room to themselves. Zoey and Nicole go to the housing office hoping to choose a roommate, but their request is denied (After Nicole accidentally destroy's the secretary's perfume collection).
LED-Large: Zoey and Nicole are thrilled when Dana gets accepted to a European exchange program at PCA, **[Not fluent]** but when they find out they have to share a dorm room with Lola Martinez, a goth-punk girl who drinks raw eggs and talks to the dead. Meanwhile, Michael and Chase have to deal with Logan, whose father has given him a huge entertainment center for their dorm room, which turns out to be a huge headache.

Table 6.5: Sample summaries generated by baseline models. We show the original document, the gold summary, and the output summaries from four large models. We highlight the typical errors of each output summary.

Figure 6.5 shows that while the relative position of n-grams in gold summary is more close to uniformly distributed (Figure 6.3(a)), the generated summaries are still biased towards the beginning of the original document. It indicates that current models have difficulty understanding the entire documents and preserving the narrative structures.

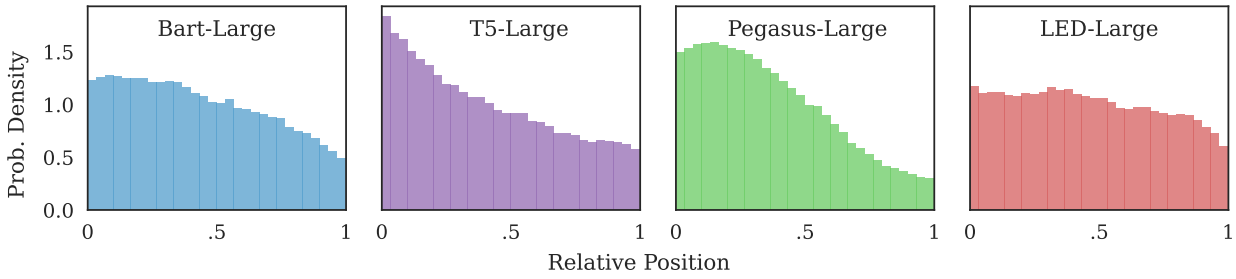

Figure 6.5: The relative positions of bi-grams of the predicted summaries in the document.

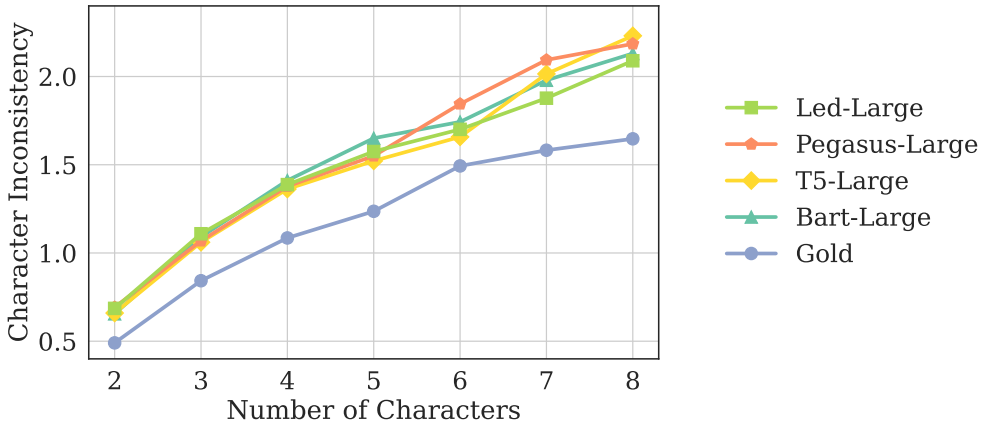

Figure 6.6: Character inconsistency between documents and summaries w.r.t. the number of characters in the document.

6.6.2 Character-Wise Analysis

Characters are essential for narratives. Since characters are not considered in Rouge scores, here we propose to measure character consistency by examining whether the major characters in the document are also mentioned in the summary. We assume that major characters appear more frequently in the narrative text. By comparing the distance between the frequency distributions of characters from the document and the summary, we can understand how well the summary includes the major characters of the document.

To this end, we first identify characters from the narrative. We run a coreference resolution model to extract clusters of entity mentions, and we only keep person entities to obtain clusters of characters.[9] We regard each cluster size as the frequency of the corresponding character and then normalize it as a probability. We measure the character inconsistency as the cross-entropy (CE) between the two frequency distributions of characters. A higher CE implies a higher character inconsistency.

[9]We use CoreNLP for coreference resolution and named entity recognition.

Model	**R-1**	**R-2**	**R-L**
Novel Chapter	32.56	6.83	16.25
w/ NARRASUM pretraining	32.88	6.80	16.19
BookSum-Paragraph	21.17	4.35	16.78
w/ NARRASUM pretraining	21.83	4.86	17.13

Table 6.6: Model performance on Novel Chapter and BookSum-Paragraph with and without pretraining on NARRASUM.

In Figure 6.6, we group the test instances of NARRASUM based on the number of distinct characters, and show the cross-entropy of the gold summary and the generated summaries. Compared with the gold summaries, the generated summaries are less consistent with the document at the character level. In general, the difference of cross-entropy between gold summary and generated summaries increases as the number of characters increases, indicating that it is harder for the summarizer to keep the character-level consistency when the document describes more characters.

6.7 Application to Other Tasks

Besides presenting NARRASUM as a benchmark for narrative summarization, we further explore the broader benefits of this dataset to narrative-related tasks. We first investigate whether pre-training on NARRASUM can improve performance on other narrative summarization tasks. To this end, we first pre-train a BART-Large model on NARRASUM and then finetune it on Novel Chapter and BookSum-Paragraph. We compare with the finetuned models without pre-training on NARRASUM. As shown in Table 6.6, pre-training on NARRASUM can improve model performance on both datasets, indicating that NARRASUM is beneficial to other narrative summarization tasks.

We then investigate if NARRASUM can help the model learn general knowledge of narrative understanding and summarization. For this, we first pre-train a BART-Large model on NARRASUM and then apply it to several downstream tasks in a zero-shot manner. We choose five tasks that are designed for narrative understanding, i.e., MCTest (Richardson, Burges and Renshaw, 2013), MovieQA (Tapaswi et al., 2016) , LiSCU (Brahman et al., 2021), CBT (Hill et al., 2016), and QuAIL (Rogers et al., 2020), and one task for narrative summarization, i.e., Reddit TIFU (Kim, Kim and Kim, 2019).

MCTest is a dataset designed for open-domain multiple-choice reading comprehension. The dataset contains 500 fictional stories, with four multiple choice questions per story.

CBT is also an dataset designed for open-domain reading comprehension. The dataset builds question-answer pairs from 108 children's books with clear narrative structure.

MovieQA aims to evaluate models' ability of automatic story comprehension. The dataset consists of 14,944 multiple-choice questions sourced from 408 movies. Each question has five options. We use the movie summaries as input to answer these questions.

LiSCU is a character-centric narrative understanding task to test the model performance from the perspective of characters. This dataset contains 1,708 literature summaries and 9,499 character descriptions. Given the literature summary, the model needs to identify the character's name from an anonymized character description and a list of character candidates.

QuAIL is a machine reading comprehension benchmark with varying types of reasoning. Solving this challenge requires an understanding of not only the text-based information from the document but also the world knowledge and commonsense knowledge. Documents in QuAIL are collected from fiction, user stories, and so on. Each question has four options.

Reddit TIFU is an abstractive summarization dataset. It consists of 120K crowd-generated posts from the online discussion forum Reddit, as well as their corresponding summaries. Different from other narrative summarization datasets we discussed in the paper, narratives in Reddit TIFU are mostly written in informal and conversational text, and the story is about the poster doing something wrong or messing everything up. These features make Reddit TIFU a good out-of-domain test data to evaluate the models' generalization power for narrative summarization.

We use models trained on the summarization task to solve these tasks in a zero-shot manner. In other words, we do not use any training data from these tasks. For discriminative tasks, we first convert the (question, answer) pair into a statement using a T5 model (Chen, Choi and Durrett, 2021), and then evaluate the probability of generating each statement conditioned on the document (Zhao, Yao, Yu, Song, Yu and Chen, 2022). We choose the candidate with the highest generation probability as the predicted answer. Models are evaluated using Accuracy. For the summarization task, we directly apply the trained model to create the summary. Models are evaluated using the Rouge-1 F measure.

We compare the model pre-trained on NARRASUM with those pre-trained on other narrative summarization datasets such as Novel Chapter and BookSum. As shown in Table 6.7, the model pre-trained on

Evaluated → Trained ↓	MCTest Accuracy	MovieQA Accuracy	LiSCU Accuracy	CBT Accuracy	QuAIL Accuracy	Reddit Rouge-1
NovelChapter	69.66	54.60	25.81	79.90	56.95	28.91
BookSum	70.50	55.21	26.75	80.24	56.33	26.08
NARRASUM	71.83	56.64	26.85	80.66	57.37	32.80

Table 6.7: Zero-shot performance (Accuracy or Rouge-1) of the model trained on NarraSum and those on other summarization datasets.

NARRASUM achieves better performance on all narrative-related downstream tasks compared with those pre-trained on other datasets. It indicates that NARRASUM contains high-quality knowledge about narrative understanding and summarization, which can be beneficial to general narrative-related tasks as well.

6.8 Conclusion

We present NARRASUM, a large-scale narrative summarization dataset that contains plot descriptions of movies and TV episodes and the corresponding summaries. Narratives in NARRASUM are of diverse genres, and the summaries are highly abstractive and of varying lengths. Summarizing the narratives in NARRASUM requires narrative-level understanding, which poses new challenges to current summarization methods. Experiments show that current models struggle with creating high-quality narrative summaries. We hope that NARRASUM will promote future research in text summarization, as well as broader NLP studies such as machine reading comprehension, narrative understanding, and creative writing.

CHAPTER 7: ENHANCING ZERO-SHOT NARRATIVE READING COMPREHENSION WITH NARRATIVE SUMMARIZATION

The primary objective of summarization is to help users better grasp key information and understand the document. In this chapter, we investigate the potential of utilizing automatically constructed summarization datasets to improve not only summarization itself but also machine reading comprehension in a zero-shot manner. We take narrative reading comprehension as an example and demonstrate that narrative summarization data can facilitate narrative comprehension with minimal supervision.

7.1 Introduction

Narratives have long been recognized as a valuable resource for linguistic, scientific, cultural, and social learning (Rosen, 1985; Knoespel, 1991; Lyle, 2000; Nash, 2005; Bettelheim, 2010). Narrative comprehension, therefore, is considered a fundamental aspect of human intelligence (Bruner, 1997) and an important tool for cognitive development and meaning-making (Polkinghorne, 1988). With this motivation, previous research has tackled the task of narrative reading comprehension, which involves automatically comprehending a given narrative and answering questions related to it (Hirschman et al., 1999; Richardson, Burges and Renshaw, 2013).

However, in comparison to general text comprehension, which typically focuses on the understanding of named entities and factual information (Rajpurkar et al., 2016), narrative comprehension presents unique challenges. Specifically, it requires understanding the foundational elements of narratives. These elements include events along with their temporal and causal connections; settings such as the time, place, and environment; as well as characters, including their motivations, desires, emotions, and relationships with other characters. Together they exhibit intricate plot structures and involve complex character interactions, making it challenging for machines to comprehend. Despite the availability of extensively annotated data for general text reading comprehension, there is currently a lack of sufficient annotated data in the narrative domain, and it is not optimal to directly use models trained on general text data for narrative reading

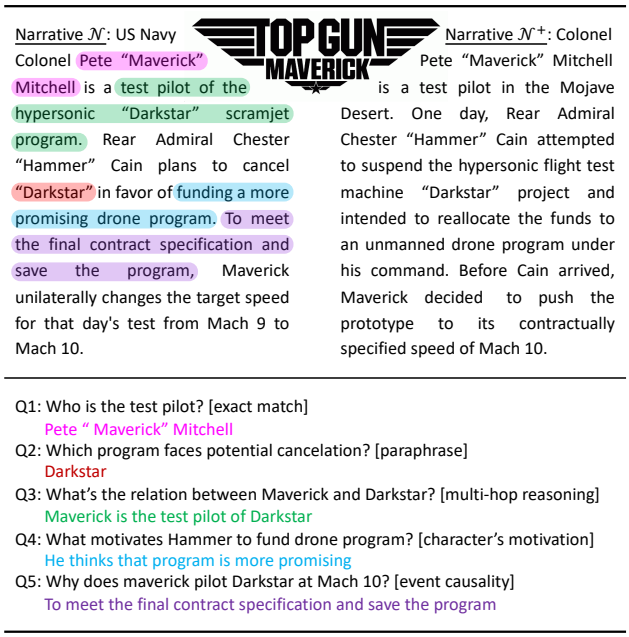

Figure 7.1: Illustration of parallel reading. $\mathcal{N}$ and $\mathcal{N}^+$ are different renderings of the same story. The key idea is to ask questions from $\mathcal{N}$ and encourage the model to answer them from $\mathcal{N}^+$. This helps the model in learning deep comprehension skills (as indicated in []).

comprehension. Hence, there is a need to develop data-efficient learning approaches for narrative reading comprehension.

To address the aforementioned challenges, our idea is to leverage *parallel reading*: reading two parallel narratives that convey the same story but differ in various aspects of story-telling style. This idea aligns with the classical model of narrative theory (Genette, 1983), which emphasizes the perspectival nature of narratives – narratives encompass not only the sequence of events (the *story*), but also the ordering, granularity, point-of-view, and localization (the *discourse* and *narrating*). Ideally, comprehending either narrative would result in the same understanding of the story. Therefore, we can teach the model to develop reading comprehension skills by asking questions based on one narrative and encouraging the model to answer them by reading the parallel narrative. Figure 7.1 illustrates this concept. We will explain later how we operationalize this idea of asking and answering questions through masked language modeling.

Learning from parallel reading offers two advantages. Firstly, by exposing the model to narrative variations of the same story, we discourage its reliance on text-matching and enhance its ability to comprehend paraphrases, integrate information from long contexts, and perform multi-hop reasoning (as seen in Q2 and Q3 in Figure 7.1). Secondly, one narrative may contain information that is not explicitly stated in the other narrative, but can be implicitly inferred through a deeper understanding of the context. Training a

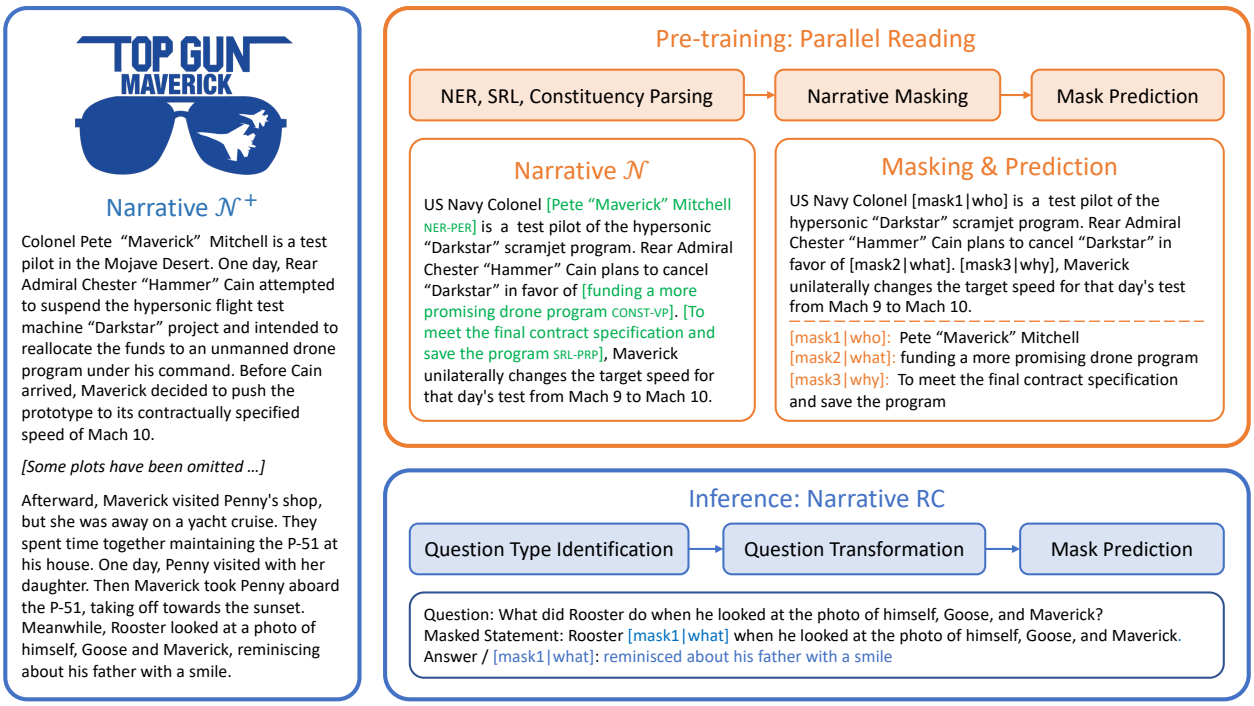

Figure 7.2: Illustration of the proposed approach, PARROT. During pre-training, we collect two parallel narratives, $\mathcal{N}^+$ and $\mathcal{N}$. We mask narrative-specific spans in $\mathcal{N}$ and pre-train the model to predict these spans by reading $\mathcal{N}^+$. During inference, we transform the question into a masked statement, following the pre-training format. Then we apply the pre-trained model to predict the answer based on the narrative and the masked statement. Note that for illustrative purposes, $\mathcal{N}^+$ is shared between pre-training and inference, but in real scenarios, there is no overlap.

model to deduce such implicit information empowers it to surpass superficial understanding and grasp the implicit information and underlying meaning within the narrative (as seen in Q4 and Q5 in Figure 7.1). These advantages have been demonstrated in pedagogy to improve students' reading comprehension abilities (Schumaker, Denton and Deshler, 1984; Grellet, 1981; Yano, Long and Ross, 1994; Sipe, 2001).

With this idea in mind, we propose PARROT [1], a novel pre-training approach for zero-shot narrative comprehension. Figure 7.2 shows an overall illustration. It selectively masks important narrative elements within one narrative, and then pre-trains the model to predict these masked elements by reading the parallel narrative. To encourage PARROT to learn about a wide array of narrative elements, we mask a diverse set of elements covering characters, events, time, place, environments, and more. Lastly, to enable PARROT to perform narrative reading comprehension in a zero-shot manner, we narrow the disparity between the pre-

[1]Stands for **pa**rallel **r**eading for ze**ro**-shot narrative comprehension.

training task of span prediction and the downstream task of reading comprehension by aligning their data formats.

We conducted experiments on two narrative reading comprehension benchmarks, narrativeQA (Kočiský et al., 2018) and FairytaleQA (Xu et al., 2022), to evaluate PARROT . The results demonstrate that without any human annotation, PARROT achieves performance that is comparable to that of a fully supervised model. Furthermore, PARROT exhibits superior performance compared to supervised models when applied to out-of-domain datasets, demonstrating its effectiveness in transfer learning scenarios.

Our contributions are three-fold:

- We present PARROT , a novel pre-training approach for effective zero-shot narrative comprehension.
- We introduce a novel *parallel reading* strategy that involves utilizing different versions of narratives during pre-training to foster genuine narrative understanding.
- Our approach achieves competitive or better performance when compared to supervised models, showcasing its effectiveness in narrative comprehension tasks.

7.2 Method

In narrative reading comprehension, the input is a narrative $\mathcal{N}$ and a question q, while the output is a concise answer a. We develop PARROT , a zero-shot solution for this problem. PARROT utilizes a masked language modeling (MLM) based pre-training approach, which incorporates a selective span masking strategy to mask essential narrative elements (Sec. 7.2.1) and a parallel reading strategy to learn to predict the masked spans (Sec. 7.2.2). Next, to utilize the pre-trained model in a zero-shot fashion, we transform the downstream narrative reading comprehension task to match the format of the pre-training task (Sec. 7.2.3). Figure 7.2 shows an overall illustration.

7.2.1 Selective Span Masking

In model pre-training, a commonly used technique is masked language modeling (MLM), where spans are randomly masked for the model to predict (Devlin et al., 2019; Raffel et al., 2020). In previous works on pre-training for reading comprehension, named entities and recurring spans were masked as they are more closely associated with factual information (Ram et al., 2021; Bian et al., 2021). However, for narrative

comprehension, the model needs to understand not just named entities but also various other narrative elements such as events, causality, temporal relationships, environmental settings, characters, their desires, personality traits, and relationships with others, to name a few. Previous masking strategies do not cover all these essential elements adequately.

Therefore, to enhance the model's ability to comprehend narratives, we incorporate a diverse set of masked spans to encourage the learning of a wide range of comprehension skills specific to narratives. We carefully select three types of spans to mask.

- Named entities: Named entities play a crucial role in narratives as they help identify characters and settings (such as time and place) within the narrative. We choose nine types of named entities: [2] Person, Location, Geopolitical Entity, Facility, Organization, Time, Date, Event, and Products.

- Semantic roles: Named entities alone can not encompass all narrative elements, such as settings like *last week* and *a small town*, event causality, characters' purpose, and more. Since these narrative elements usually unfold along with events, we focus on the associated arguments of verbs and include five semantic roles: [3] Direction (ARGM-DIR), Location (ARGM-LOC), Time (ARGM-TMP), Purpose (ARGM-PRP), Cause (ARGM-CAU), and Manner (ARGM-MNR).

- Verb and adjective phrases: A narrative can be seen as a sequence of events organized by a narrator (Schank and Abelson, 2013). To directly comprehend events, we identify verb phrases using constituency parsing [4] and mask them. Additionally, we mask adjective phrases to enhance the understanding of narrative settings and character characterization.

Given a narrative $\mathcal{N}$, we first identify and mask some spans, $m(\mathcal{N})$, and then pre-train a sequence-to-sequence model to predict these spans using the remaining text, $\mathcal{N}_{\setminus m(\mathcal{N})}$, and the original narrative, $\mathcal{N}$. We refer to this model as PARROT$_{\text{single}}$. The loss function is

$$\mathcal{L}_{\text{single}} = -\log p\left(m(\mathcal{N}) \mid \mathcal{N}_{\setminus m(\mathcal{N})}, \mathcal{N}\right). \tag{7.1}$$

[2]We employ Spacy for NER. `https://spacy.io/`

[3]We employ AllenNLP for SRL. `https://allenai.org/allennlp`

[4]We employ AllenNLP for constituency parsing.

7.2.2 Parallel Reading

Predicting masked spans using the original narrative $\mathcal{N}$ may result in the model trivially relying on the superficial lexical overlap. To mitigate this issue, we propose "parallel reading". Instead of solely masking and predicting spans within a single narrative $\mathcal{N}$, we leverage an additional parallel narrative, denoted as $\mathcal{N}^+$, which tells the same story as $\mathcal{N}$ but differs in granularity, point-of-view, etc. By simultaneously reading $\mathcal{N}$ and $\mathcal{N}^+$ and predicting the masked key information, we encourage the model to abstract the textual content and foster a genuine and deeper comprehension of narratives, avoiding overreliance on superficial textual matching clues. For example, in Figure 7.2, predicting `[mask2]` and `[mask3]` in $\mathcal{N}$ based on $\mathcal{N}^+$ requires more advanced comprehension skills, such as understanding character motivation and event causality.

Here we provide more details regarding parallel reading. Without loss of generality, we assume that $\mathcal{N}^+$ is longer than $\mathcal{N}$. We selectively mask spans in the shorter narrative, $\mathcal{N}$, and utilize the longer narrative, $\mathcal{N}^+$, as a source of evidence to predict the masked spans, since the longer narrative is likely to contain the necessary information present in the shorter narrative.

However, $\mathcal{N}$ might also contain some spans that are not answerable from $\mathcal{N}^+$. Masking such spans can result in noise in the training data. To mitigate this noise, we apply two filtering steps: one at the sentence level and another at the span level. At the sentence level, for each sentence s in $\mathcal{N}$, we require the Rouge-1 Precision score (Lin, 2004) between s and $\mathcal{N}^+$ to surpass a predefined threshold. If it does not, we do not mask spans from s. This criterion ensures that the remaining sentences in $\mathcal{N}$ align closely with the corresponding content in $\mathcal{N}^+$. At the span level, we selectively mask spans in $\mathcal{N}$ that directly or indirectly appear in $\mathcal{N}^+$. For spans that correspond to a named entity, we verify their presence in $\mathcal{N}^+$ using exact match. For spans that correspond to semantic roles or constituency phrases, which are more likely to be paraphrased, we adopt a more lenient criterion. For them, we calculate the Rouge-1 Precision score between the span and $\mathcal{N}^+$, setting a threshold to determine the acceptability of the span candidates for masking.

Lastly, we pre-train the model to predict the masked content within $\mathcal{N}$, given the concatenation of the masked narrative, $\mathcal{N}_{\backslash m(\mathcal{N})}$, together with the longer narrative, $\mathcal{N}^+$. The loss function is

$$\mathcal{L}_{\text{parallel}} = -\log p\left(m(\mathcal{N}) \mid \mathcal{N}_{\backslash m(\mathcal{N})}, \mathcal{N}^+\right). \tag{7.2}$$

7.2.3 Adapting to Reading Comprehension

In general, after the MLM pre-training, the pre-trained model requires fine-tuning with additional data to adapt to the specific downstream task. This fine-tuning is necessary because the pre-training task and the downstream task can be in different formats. However, in this paper, we do not assume access to the availability of any fine-tuning data and directly utilize the pre-trained model in a zero-shot manner. The key insight is that the reading comprehension task can be transformed into the MLM task. For example, in Figure 7.2, the question "*What did Rooster do when he looked at the photo?*" can be transformed into "*Rooster* `[mask]` *when he looked at the photo*", and the answer can be obtained by filling in the masked part. To achieve this transformation, we use QA2D (Demszky, Guu and Liang, 2018), which leverages a neural sequence model to generate masked statements from questions.

One drawback of this transformation strategy, as well as the pre-training strategy, is that the masked statement does not contain the question-type information typically conveyed by the wh-word in questions. For instance, without the original *what* question with the answer of "*reminisced about his father with a smile*", the masked statement in our example from Figure 7.2 can also be interpreted as a *how* question, leading to the possibility of filling the mask with a different answer such as "*felt delighted*".

To mitigate this ambiguity, we introduce a special type token preceding the mask to provide more accurate information about the question type. This token, as we illustrated in Figure 7.2, is typically a wh-element, such as *who* and *what*, which is extracted from the original question. To extract these words, we employ a constituency parser to parse the question and then identify the elements labeled with syntactic tags such as "WHNP", "WHADVP", "WHADJP", or "WHPP". During pre-training, since we lack the actual questions, we infer the question type based on the type of masked spans. The mapping between the span type and the question type is provided in Table 7.1.

By transforming the question to a masked statement during inference and incorporating the question type during pre-training, we establish a consistent data format for both pre-training and inference. This thereby empowers the model to perform zero-shot inference without explicit fine-tuning.

7.3 Experiments

In this section, we evaluate the performance of PARROT .

Wh- Type	Span Type
Who	NER-PERSON
When	NER-TIME/DATE, ARGM-TMP
Where	NER-LOC/GEO/FAC, ARGM-LOC/DIR
Why	ARGM-CAU/PRP
How	ARGM-MNR, ADJP
What	Others

Table 7.1: The mapping between the type of question and the corresponding type of masked span. This mapping enables the model to identify the appropriate type of question during pre-training.

7.3.1 Datasest

Datasets for Pre-training: For parallel reading in the pre-training phase, we utilize NarraSum Zhao, Brahman, Song, Yao, Yu and Chaturvedi (2022), the dataset we collected of 122K parallel narrative pairs obtained from plot descriptions of movies and TV episodes. After processing, we obtain a total of 57.4K paired narratives and 154.5K question-answer pairs. The average lengths of $\mathcal{N}$ and $\mathcal{N}^+$ are 125 and 926 tokens, respectively. Each narrative pair includes 2.7 masked spans on average.

To reduce input length and enhance computational efficiency, we partition the shorter narrative $\mathcal{N}$ into smaller segments and predict the spans within each segment separately. However, we also need to strike a balance as excessively short segments would increase the overall number of training instances. Therefore, we opt to divide $\mathcal{N}$ into segments based on every three sentences.

Datasets for Evaluation: To evaluate the performance of PARROT , we conduct experiments on two narrative reading comprehension benchmarks: NarrativeQA (Kočiský et al., 2018) and FairytaleQA (Xu et al., 2022). Since PARROT is zero-shot, we solely use the test sets of these datasets for evaluation. The narratives in FairytaleQA are derived from children's stories, while the narratives in NarrativeQA consist of plot summaries from books and movie scripts. For NarrativeQA, to avoid any potential overlap with the pre-taining data, we only consider instances derived from books for evaluation purposes. The average length of the narratives in these datasets is 150 and 659 tokens, respectively, and their test sets contain 1,007 and 10,557 question-answer pairs, respectively.

7.3.2 Setup

Implementation Details: The underlying model in PARROT is a T5-base (Raffel et al., 2020). We chose T5 because it has been pre-trained on a similar MLM task. Furthermore, compared with other MLM-based

pre-trained models such as BART, T5 only predicts the masked tokens, making it more computationally efficient. During pre-training, we employ the AdamW optimizer (Loshchilov and Hutter, 2019) with a learning rate of 3×10^{-5} and a batch size of 512. We choose a large batch size because the pre-training data can be noisy. We incorporate warmup for the first 50 steps and implement early stopping based on the model's performance on the validation set. Training the models is conducted on four Tesla 3090 GPUs with 24 GB memory, taking approximately 4 hours to complete the pre-training process.

Baselines: Our first baseline is an information retrieval (IR) baseline adopted by Kočiský et al. (2018), which selects the most similar sentence in the narrative to the given question and considers it as the answer. For computing this similarity, we use TF-IDF based cosine similarity. To establish stronger baselines, we compare PARROT with the model described in Lewis et al. (2021), which automatically generates question and answer pairs from the narrative. This involves utilizing an answer extraction (AE) model and a question generation (QG) model trained on three MRC datasets: NQ (Kwiatkowski et al., 2019), TriviaQA (Joshi et al., 2017), and SQuAD (Rajpurkar et al., 2016). With these models, we generate question and answer pairs from the narratives in NarraSum, and then train a reading comprehension model based on T5-base. We refer to this baseline as AE-QG.

Additionally, we compare PARROT with ChatGPT [5], a state-of-the-art large language model, and Vicuna-13B (Chiang et al., 2023), one of its best open-source alternatives. To use these models, we use the instruction "`Please generate a brief answer rather than a complete sentence to the following question based on the provided passage as evidence.`", alongside the passage and question that are appended. [6]

Lastly, we compare with fine-tuned models. We fine-tune the T5-base model on the training sets of narrativeQA and FairytaleQA, resulting in two fine-tuned models. We treat these results as upper bounds due to their supervised nature.

Evaluation Measure: Following the official evaluation of the two benchmarks, we use Rouge scores (Lin, 2004) between the predicted and the gold answers to evaluate the models.

	FairytaleQA			NarrativeQA		
	Rouge-1	**Rouge-2**	**Rouge-L**	**Rouge-1**	**Rouge-2**	**Rouge-L**
T5 Finetuned on FairytaleQA	54.64	40.43	54.03	45.29	25.73	44.59
T5 Finetuned on NarrativeQA	49.64	38.45	49.10	65.05	36.04	64.49
IR (TF-IDF) (Kočiský et al., 2018)	21.64	14.30	20.82	16.61	7.79	15.67
AE-QG (Lewis et al., 2021)	43.29	29.81	42.89	53.61	28.11	53.27
Vicuna-13B (Chiang et al., 2023)	37.52	20.44	35.98	32.59	17.37	31.46
ChatGPT	44.32	27.10	43.49	41.27	24.63	40.07
PARROT$_{\text{single}}$	40.32	30.35	40.01	50.01	26.78	49.60
PARROT	**48.56**	**36.83**	**48.10**	**55.71**	**30.81**	**55.32**

Table 7.2: Results evaluated on FairytaleQA and NarrativeQA by Rouge scores. PARROT outperforms all baselines and achieves comparable or superior performance compared to supervised models in the out-of-domain setting.

7.3.3 Results

Table 7.2 presents the performance of PARROT and baselines on FairytaleQA and NarrativeQA datasets. PARROT exhibits superior performance, significantly surpassing all zero-shot baselines (approximate randomization (Noreen, 1989; Chinchor, 1992), $p < 0.01$) . Additionally, it achieves performance that is 89.0% and 85.8% comparable to those of fully supervised upper-bounds in terms of Rouge-L (48.10 vs. 54.03 and 55.32 vs. 64.49). These results demonstrate the effectiveness of PARROT in narrative reading comprehension. When comparing the strategies of single and parallel reading, PARROT achieves significantly higher performance compared to its single-reading counterpart, PARROT$_{\text{single}}$, on both datasets. This result emphasizes the crucial role of parallel reading in enhancing model performance.

We also compare PARROT to supervised models under the out-of-domain setting, i.e., training the supervised model on one dataset and evaluating it on another. These results are displayed in gray font in the table. PARROT demonstrates competitive performance on FairytaleQA (Rouge-L of 48.10 vs. 49.10) and superior performance on NarrativeQA (Rouge-L of 55.32 vs. 44.59). This further demonstrates that PARROT can acquire general narrative comprehension skills and effectively apply them to diverse narratives.

Among the large language model baselines, ChatGPT exhibits stronger performance than Vicuna-13B. AE-QG models also achieve strong performance. However, these models require additional training data for training the answer extraction and question generation components. Furthermore, the generated question-

[5] `https://chat.openai.com/`

[6] We tried different instructions and select the best-performing one.

	FairytaleQA	NarrativeQA
IR (TF-IDF)	2.12	2.37
AE-QG	2.56	2.91
Vicuna-13B	2.36	2.61
ChatGPT	2.49	2.86
PARROT$_{\text{single}}$	2.30	2.78
PARROT	2.71	3.10

Table 7.3: Results of human evaluation on FairytaleQA and NarrativeQA.

answer pairs may contain errors, which could potentially impact the model's overall performance during pre-training.

7.3.4 Human Evaluation

To obtain a more reliable assessment of the model performance, we further conduct a human evaluation via Amazon Mechanical Turk (AMT). We randomly select 100 test instances from the test sets of both datasets. For each instance, we show three independent annotators the question, correct answers, and the answers generated by various systems. We then ask annotators to rate the quality of the predicted answers on a Likert scale ranging from 1 to 5. To maintain the evaluation quality, we require annotators to be AMT Masters based in the United States, with more than 1,000 HITs approved and an approval rate exceeding 98%. We manually review the annotation results, and if we identify annotators consistently providing low-quality annotations, we block them and re-assign their tasks. Annotators are compensated at a rate of $14 per hour, exceeding the local minimum wage.

Table 7.3 shows the results of human evaluation. The inter-annotator agreement score is 0.7003 in Gwet's gamma. Results from both datasets, along with the automatic measures, consistently demonstrate that Parrot outperforms the baseline models.

7.4 Analysis

We conduct analysis to better understand the behavior of PARROT .

7.4.1 Type of Masked Spans

One of our work's major contributions is incorporating a carefully selected and diverse set of masked spans geared toward narrative comprehension. To highlight the diversity, we analyze the distributions of

Figure 7.3: Distribution of the types of wh-elements and the sources of masked spans in pre-training data.

different question types and the types of masked spans in the pre-training data. The results are presented in Figure 7.3. In terms of question types, the pre-trained data contains six major types: *what* (41.7%), *who* (25.6%), *when* (13.4%), *where* (10.4%), *why* (5.6%), and *how* (3.3%). In terms of masked spans, it shows that named entities (NE), semantic roles (SR), and constituency phrases (CP) are evenly distributed within the pre-training data. Specifically, named entities are predominantly represented by PERSON (72.0%) and ORG (16.1%) categories. Within semantic roles, Time (41.5%), Location (25.5%), and Purpose (14.9%) are the top three categories. Within the constituency phrases, almost all of them fall under the category of verb phrases.

To investigate the impact of different mask types on the overall performance, we conduct an ablation study. During the construction of the pre-training data, we gradually expand the type of masked elements from named entities to semantic roles and constituency phrases. We also compare with a random masking strategy that aligns with the original pre-training objective of T5. The performance of the models trained on these three versions of the pre-trained data is presented in Table 7.4. It reveals that focusing on named entities can improve the model performance, which aligns with previous research findings. However, relying solely on named entities is insufficient to encompass all narrative elements. By incorporating semantic roles, the model achieves a substantial improvement in performance. By including constituency phrases, we observe a further enhancement. On the contrary, when we continue pre-training with a random span masking strategy, we do not observe improvement in model performance. These results support our hypothesis that incorporating a diverse range of masked spans can significantly enhance models' ability of narrative comprehension.

	FairytaleQA	NarrativeQA
Random	8.38 / 3.29 / 8.31	13.54 / 4.02 / 13.52
None	11.99 / 6.14 / 11.84	10.67 / 4.38 / 10.54
+ NE	35.10 / 23.89 / 34.98	48.72 / 24.93 / 48.53
+ SR	45.55 / 34.20 / 45.21	54.83 / 29.95 / 54.46
+ CP	48.56 / 36.83 / 48.10	55.71 / 30.81 / 55.32

Table 7.4: The contribution of each source of masked spans to the final performance (R-1/R-2/R-L). We start with T5-base with and without further pre-training (Random and None). We then incrementally introduce named entities (NE), semantic roles (SR), and constituency phrases (CP) into the pre-trained data.

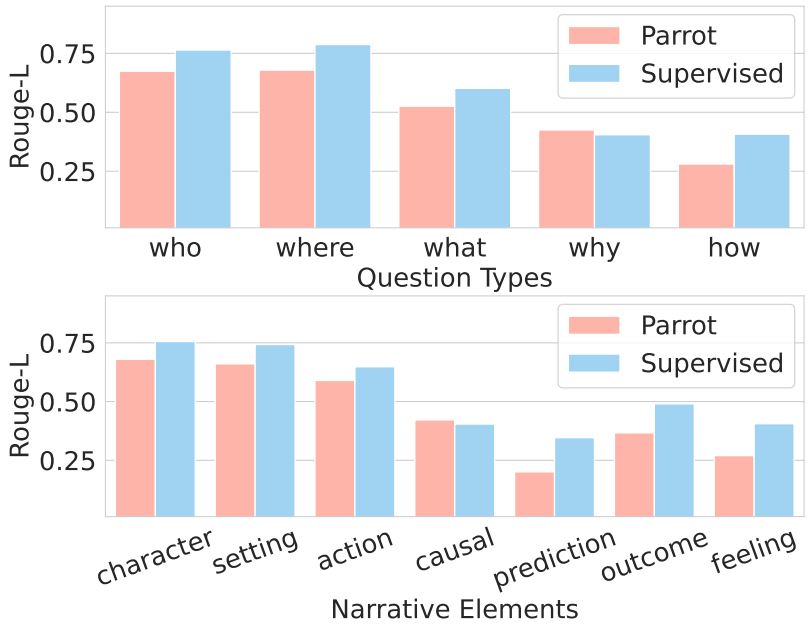

Figure 7.4: Fine-grained model performance on FairytableQA w.r.t. the types of questions (top) and narrative elements (bottom).

7.4.2 Decomposition of Model Performance

We proceed to conduct a thorough analysis of the model's performance at a finer granularity. To accomplish this, we partition the FairytaleQA dataset into smaller subsets based on question types and narrative elements, as annotated within the dataset. Then we evaluate the model's performance on the individual subsets and compare it with the performance of the supervised model. The results are illustrated in Figure 7.4.

Comparing the results with the supervised model, PARROT demonstrates competitive performance in questions that involve identifying characters (*who*) and their activities (*what*), establishing causal relationships between events (*why*), and understanding the setting of the narrative (*where*). However, when dealing with

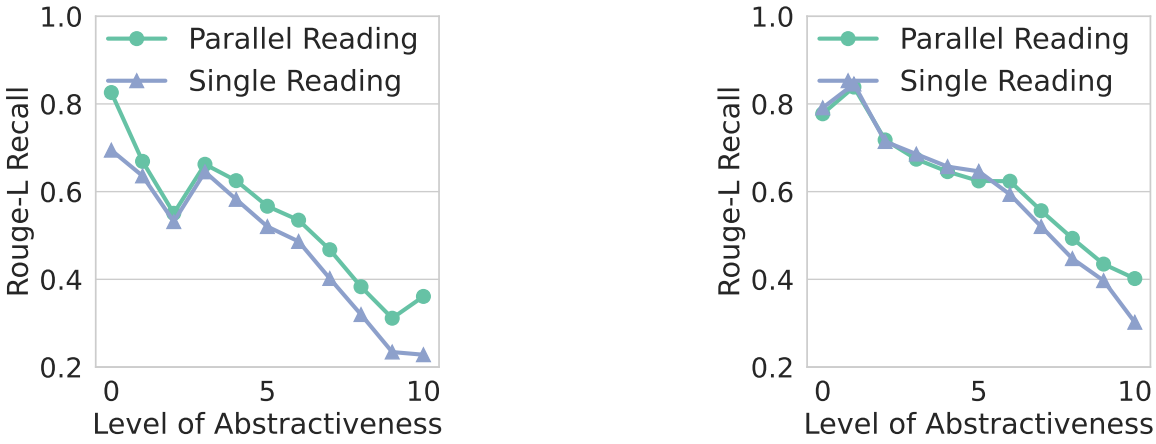

Figure 7.5: Model performance on FairytaleQA (left) and narrativeQA (right) w.r.t. the abstractiveness level between the question and the narrative. We report Rouge-L Recall to evaluate whether the correct answer is included in the predicted answer.

more intricate narrative aspects such as pinpointing outcomes, predicting unknown events, and deciphering characters' emotional states (*how*), PARROT exhibits a larger performance gap. This particular strength and weakness align with the distribution of the types of masked spans present in the pre-training data. We leave enhancing the comprehension of these narrative components for future work.

7.4.3 Impact of Parallel Reading

In addition to incorporating a diverse array of narrative elements, a significant contribution of PARROT is leveraging parallel reading to abstract the textual content and comprehend the underlying meaning of the narrative. As discussed in Section 7.3.3, PARROT achieves better overall performance compared to $\text{PARROT}_{\text{single}}$. In this section, we analyze how parallel reading impacts the model's performance when the question is less lexically overlapped with the narrative.

To accomplish this, we divide the test set into subsets based on the level of abstractiveness between the question and the narrative. More specifically, we first identify the most similar sentence in the narrative with the question as the evidence sentence, and then use the sum of Rouge-1 precision and Rouge-2 precision between the question and the evidence sentence to approximate the level of abstractiveness. Higher Rouge Precision indicates lower abstractiveness. Figure 7.5 shows the model's performance based on the degree of abstractiveness between the question and the narrative.

The results demonstrate that, in general, as the question becomes increasingly abstractive (the right side of the x-axis), the performance gap between $\text{PARROT}_{\text{single}}$ and PARROT becomes more significant. This

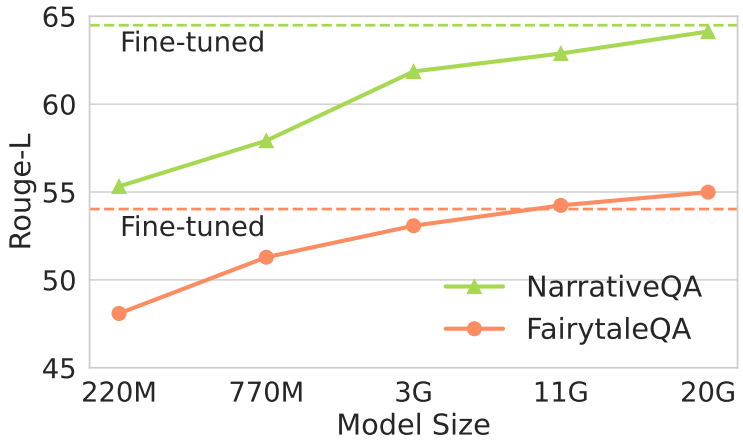

Figure 7.6: Model performance w.r.t. the size of underlying models.

finding indicates that compared with PARROT$_{\text{single}}$, PARROT is better at understanding abstractive questions and finding answers based on genuine comprehension, rather than mere text matching. It supports our motivation that parallel reading enhances the model's ability to comprehend the underlying meaning of the narrative.

Interestingly, in highly extractive scenarios (the left side of the x-axis), PARROT also outperforms its single-reading counterpart on FairytaleQA. This is because PARROT$_{\text{single}}$ tends to directly copy text from the narrative, which sometimes results in errors related to answer resolution when the copied part includes a pronoun instead of the proper entity mention. In contrast, PARROT is capable to select the appropriate entity mention as the answer, rather than mechanically copying the pronoun.

7.4.4 Scaling to Larger Models

We further investigate the potential benefits of PARROT 's pre-training strategy for larger models. To this end, we experiment with different sizes of the underlying T5 model, namely base (220M), large (770M), XL (3B), XXL (11B), and UL2 20B (Tay et al., 2023), a T5-like large model. We fine-tune the entire parameters for T5 base and T5 large. For larger models, we utilize LORA (Hu et al., 2022) to facilitate more efficient fine-tuning due to the large computation. The results of these models are presented in Figure 7.6.

The results show that as we increase the size of the underlying model, the performance improves gradually, approaching and eventually surpassing the performance of the fine-tuned baselines. This experiment indicates that PARROT can effectively enhance a model's ability of narrative comprehension irrespective of its size.

Narrative I: Two years passed, and the youth no longer thought of any danger. Hence, when the flowers began to bloom in the spring, Rose-Red went driving in a small wagon outside the city, near the river. And there one of the prince's servants saw her, and informed his master. The latter sent for the youth, who, since he could not conceal the matter, told him the whole story exactly as it had happened. Said the prince: "The whole blame rests on Rose-Red. I do not reproach you. Yet since she is now your wife I will let the whole matter rest. But Molo will have to suffer for it!"

Question: Why did Tsui tell the prince the whole story exactly as it had happened?

Model Outputs:
Vicuna-13B: Rose-Red told the prince the whole story because the prince asked her to.
ChatGPT: To inform the prince of what had happened.
AE-QG: He could not conceal the matter.
PARROT : Because he could not conceal the matter.

Reference: He could not conceal the matter.

Narrative II: "That is a pity," returned the young man politely, but in his heart he rejoiced, for he wished to be alone. Leaving the house, he entered a grove of trees which stood behind, and took the box from his pocket. He raised the lid, and out flew the three little birds. "Good master, what shall we do for thee?" asked they, and he answered, "Bring me the finest horse that ever was seen, and the grandest dress, and glass shoes." "They are here, master," said the birds, and so they were, and never had the young man seen anything so splendid. Mounting the horse he rode into the ground where the horses were assembling for the great race, and took his place among them. Many good beasts were there which had won many races, but the horse of the fisher's son left them all behind, and he was first at the winning post. The king's daughter waited for him in vain to claim his prize, for he went back to the wood, and got off his horse, and put on his old clothes, and bade the box place some gold in his pockets. After that he went back to the weaver's house, and told him that the gold had been given him by the man who had won the race, and that the weaver might have it for his kindness to him.

Question: Why did Ian need the finest horse?

Model Outputs:
Vicuna-13B: Ian needed the finest horse for the race.
ChatGPT: To participate in the great race.
AE-QG: A pity.
PARROT : To win the race.

Reference: He wanted to win the horse-race.

Table 7.5: Sampled answers generated by different systems. For each example, we show the narrative, model predicted answers, and the referenced answer.

7.4.5 Qualitative Analysis

Table 7.5 shows two examples of generated answers from different systems. To address the first question correctly, the model must distinguish that "Tsui" refers to the young man rather than "Rose-Red," who is his wife. Vicuna falls short in making this distinction. ChatGPT offers a consecutive answer that lacks useful information. In contrast, PARROT provides the correct answer. The second narrative is notably longer than the first, with the answer to the question not explicitly provided in the text. To answer this question, the model needs to establish a connection between the earlier plot when Ian requested the finest horse and the later plot where he won the race. AE-QG fails to establish this connection, resulting in a lack of useful information. Vicuna and ChatGPT partially answer the question by mentioning the race, but do not emphasize Ian's motivation to "win the race," which is the primary reason he sought the "finest" horse. Thanks to the long-term reasoning skills acquired during parallel pre-training, PARROT accurately answered this question.

7.5 Conclusion

We introduce PARROT , a novel zero-shot approach for narrative reading comprehension based on pre-training. By selectively masking significant elements within the narrative and pre-training the model to predict these spans through parallel reading, PARROT learns to abstract essential textual content and gains a genuine understanding of the narrative. Experimental results on two diverse narrative datasets demonstrate the superiority of PARROT , showcasing its effectiveness in enhancing narrative reading comprehension. Our analysis further emphasizes the significance of employing a diverse range of masked spans and leveraging the parallel reading strategy during model pre-training.

CHAPTER 8: SUMMARY, LIMITATIONS, AND FUTURE WORKS

8.1 Summary

In this dissertation, we investigated various summarization algorithms in low-supervision settings. Firstly, we demonstrated the effectiveness of utilizing external information to enhance the model's ability to identify salient information and generate more relevant summaries. We studied opinion summarization and illustrated that our proposed approach can be adapted to diverse product categories without the need for additional manual annotation. Secondly, we explored data transformation methods for transferring knowledge from data-rich tasks to data-deficient tasks. We studied the knowledge transfer from single-document summarization (SDS) to multi-document summarization (MDS), and emphasized that not all documents carry equal importance within the context of MDS. To address this, we proposed a document-reordering approach to prioritize important documents during summarization, which can benefit the overall quality of the generated summaries. Thirdly, we introduced automated approaches to construct high-quality paired training datasets for summarization tasks. To this end, we developed DIANA and NARRASUM, two large-scale datasets for dialogue summarization and narrative summarization, respectively. Pre-training models on these datasets resulted in outstanding performance gains in downstream dialogue summarization and narrative summarization tasks. Lastly, we bridged text summarization and reading comprehension by introducing a parallel reading approach. It involved selectively masking important elements within the narrative and pre-training the model to predict these masked spans. Through this process, our approach learned to abstract essential textual content and gained a genuine understanding of the narrative. It outperformed previous zero-shot approaches and achieved comparable performance with fully supervised models.

8.2 Limitations and Future Works

8.2.1 Unveiling Salience Factors

While current approaches can generate summaries with decent quality in some domains, there is a lack of investigation into how these models identify salient information. In other words, the models cannot explain

why certain information is considered more salient than others and should be included in the summary. In a low-supervised setting, we often rely on heuristic factors such as frequency, position, or similarity to saliency-intense content. However, these factors are domain-specific and cannot be universally applied to all domains. For instance, in narratives, it is challenging to solely analyze salient events based on heuristics.

Unraveling these underlying factors is crucial for summarization algorithms as it can enhance the transparency and impartiality of the algorithm. It can also provide valuable guidance for designing individual experts within unified summarization models, and enhance controllability in generating personalized summaries, as we elaborate in the following two sections.

8.2.2 Unified Summarization Models

At present, we propose various approaches for summarization in different domains, including opinions, news, and dialogues. While these approaches can address domain-specific challenges, models developed for one particular domain cannot be easily transferred to another domain. To tackle the issue of limited supervision more effectively, a practical approach is to train a unified summarization model that can be applied across different domains.

There are two potential solutions to achieve this goal. The first solution involves leveraging large language models (LLMs), which have demonstrated their ability to understand the text and perform tasks in a zero-shot manner. Although we can assume that LLMs possess essential knowledge for summarizing texts, general instructions may not always lead to the desired summary output. In practice, we have observed that the summary generated by LLMs often tends to simply shorten the text through text simplification, rather than effectively identifying and re-expressing salient information. Therefore, it is necessary to thoroughly evaluate the performance of LLMs in different summarization domains. Additionally, evaluating the output of LLMs presents its own set of challenges.

Another possible solution is to employ a mixture of expert models, where each expert specializes in summarization for a specific domain or aspect. As mentioned earlier, different summarization tasks exhibit distinct characteristics, making it challenging to train a single model that caters to all of these characteristics. Instead, by combining multiple expert models, each model can learn specific skills for summarization. Therefore, compared with a single model, this approach can result in more controllable, explainable, and adaptable summarization systems.

8.2.3 Personalized Summarization

While the goal of summarization is to facilitate quicker information acquisition, most current summarization systems do not consider individual information needs. Different individuals may prioritize different aspects of information. While recent works have proposed aspect-based summarization or question-driven summarization approaches, creating alike models under low supervision is an intriguing research direction.

Besides, whether or not these approaches can resolve the problem is questionable, since it is challenging for people to determine in advance which aspects they may be more interested in or what questions they should ask before reading the original document. Therefore, an interesting alternative is to leverage recommendation techniques to model user profiles and generate personalized summaries. Instead of requesting users to provide aspects or questions, we can gain insights into a user's interests when summarizing documents by analyzing their browsing or clicking history. By doing so, we can generate summaries that align better with the user's preferences. In the case of opinion summarization, we can analyze a user's previous reviews and shopping histories to gain a better understanding of their preferences. Subsequently, we can focus on summarizing the aspects that are of greater importance to them.

8.2.4 Facilitating Text Understanding and Generation

Text summarization has traditionally been considered a specific NLP task requiring both text understanding and generation. However, the reciprocal benefits between text summarization and the processes of text understanding and generation have been relatively unexplored. In the preceding two chapters, we demonstrated how summaries can facilitate the understanding of dialogues and narratives, illustrating a promising avenue for future exploration. For instance, summaries can assist in revealing hierarchical relationships between events and identifying recurring event patterns. Summaries can also help us to recognize salient events and uncover the higher-level discourse structure within a document.

On the other hand, summaries can also play a pivotal role in improving the text generation process. For instance, when generating long stories, maintaining global coherence and high-level discourse structure can be challenging. Integrating summaries as an intermediate step during generation offers a solution to this challenge. Summaries can provide the model with a cohesive outline of the entire story, which makes the generated content remain focused and adhere to its intended topic. This leads to more globally coherent and consistent output in the generated content. Another possibility is to reduce the length of the history text by

replacing it with the corresponding summary. This allows the model to generate longer and more expansive content while preserving alignment with the original intent encapsulated within the summary.

One potential limitation of these approaches is their reliance on high-quality summaries, which can be challenging to scale when obtained through manual collection. However, we demonstrated in Chapter 5 and Chapter 6 that it is feasible to automatically generate extensive summaries from online resources. While these summaries are not written by humans, they still offer valuable training signals to enhance text comprehension. Future work will focus on implementing filtering strategies to improve data quality during the pre-training phase.

8.2.5 Improving Human Annotation and Evaluation

When developing a summarization system, there are two critical stages that demand substantial human effort: annotating document-summary pairs for a summarization dataset, and evaluating the quality of the summaries generated by the system. These tasks are non-trivial for two reasons. Firstly, annotators must grasp the salient content within lengthy and often poorly structured source documents. Then, they must generate or assess summaries that effectively represent the salient information. Secondly, determining the saliency of information is a subjective process, and different annotators may hold varying opinions regarding what information is relatively more important. Due to these challenges in summarization, current data collection and system evaluation procedures are problematic.

From the perspective of data collection, most publicly available large-scale summarization datasets rely on automatic collection approaches. For instance, they pair news articles with their highlights or scientific papers with their abstracts as summaries. However, these highlights and abstracts, though crafted by humans, diverge in purpose from summarization, which aims to create concise, coherent summaries to faithfully convey the most important information in a given document. For instance, news highlights may be presented in bullet points and contain additional information, resulting in ”unfluent” and ”hallucinatory” summaries. Likewise, abstracts in scientific papers often adhere to a specific structure (e.g., objective-method-result-conclusion) that may not precisely reflect the salient content of the paper. Consequently, when training a model using such datasets and encountering issues with the model-generated summaries, it is challenging to distinguish whether the problems arise from inherent model limitations or deficiencies with the training data.

From the perspective of system evaluation, a common standard for evaluation is to compare system-generated summaries to human-generated ones to determine which is better. However, this approach can

lead to misleading conclusions if the human-generated summaries are of poor quality. Even when human-generated summaries are of high quality, human annotators may still prefer system-generated summaries, as they are often more extractive than human-written ones, making annotators heuristically assume that the system-generated summaries are more faithful to the original document, even when this may not be the case. Given these limitations, it is not surprising that some LLMs can generate better summaries when compared to those generated by humans.

To address these issues, instead of directly tasking humans with creating or evaluating summaries, one potential solution is to break down the summarization process into manageable, controllable, verifiable, and replicable steps, such as content unit identification and linking, saliency estimation, factuality verification, redundancy assessment, etc. While adhering to these steps might be labor-intensive and susceptible to human errors, it is worth to explore if LLMs can assist in these steps. If yes, adopting such a machine-in-the-loop approach could significantly expedite the data collection and evaluation process, enhance annotation and evaluation quality, and mitigate bias introduced by human errors and subjective judgments.

BIBLIOGRAPHY

Afantenos, Stergos, Eric Kow, Nicholas Asher and Jérémy Perret. 2015. Discourse parsing for multi-party chat dialogues. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics pp. 928–937.
URL: *https://aclanthology.org/D15-1109*

Amplayo, Reinald Kim, Stefanos Angelidis and Mirella Lapata. 2021. Unsupervised Opinion Summarization with Content Planning. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. AAAI Press pp. 12489–12497.
URL: *https://ojs.aaai.org/index.php/AAAI/article/view/17481*

Angelidis, Stefanos and Mirella Lapata. 2018. Summarizing Opinions: Aspect Extraction Meets Sentiment Prediction and They Are Both Weakly Supervised. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics pp. 3675–3686.
URL: *https://aclanthology.org/D18-1403*

Bai, Yu, Yang Gao and Heyan Huang. 2021. Cross-Lingual Abstractive Summarization with Limited Parallel Resources. 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)*. Online: Association for Computational Linguistics pp. 6910–6924.
URL: *https://aclanthology.org/2021.acl-long.538*

Bamman, David, Brendan O'Connor and Noah A. Smith. 2013. Learning Latent Personas of Film Characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Sofia, Bulgaria: Association for Computational Linguistics pp. 352–361.
URL: *https://aclanthology.org/P13-1035*

Banerjee, Siddhartha, Prasenjit Mitra and Kazunari Sugiyama. 2015. Multi-Document Abstractive Summarization Using ILP Based Multi-Sentence Compression. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, ed. Qiang Yang and Michael J. Wooldridge. AAAI Press pp. 1208–1214.
URL: *http://ijcai.org/Abstract/15/174*

Banko, Michele, Vibhu O. Mittal and Michael J. Witbrock. 2000. Headline Generation Based on Statistical Translation. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*. Hong Kong: Association for Computational Linguistics pp. 318–325.
URL: *https://aclanthology.org/P00-1041*

Barzilay, Regina and Kathleen R. McKeown. 2005. "Sentence Fusion for Multidocument News Summarization." *Computational Linguistics* 31(3):297–328.
URL: *https://aclanthology.org/J05-3002*

Barzilay, Regina, Kathleen R. McKeown and Michael Elhadad. 1999. Information Fusion in the Context of Multi-Document Summarization. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*. College Park, Maryland, USA: Association for Computational Linguistics pp. 550–557.
URL: *https://aclanthology.org/P99-1071*

Basu Roy Chowdhury, Somnath, Chao Zhao and Snigdha Chaturvedi. 2022. Unsupervised Extractive Opinion Summarization Using Sparse Coding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics pp. 1209–1225.
URL: *https://aclanthology.org/2022.acl-long.86*

Basu Roy Chowdhury, Somnath, Faeze Brahman and Snigdha Chaturvedi. 2021. Is Everything in Order? A Simple Way to Order Sentences. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics pp. 10769–10779.
URL: *https://aclanthology.org/2021.emnlp-main.841*

Beltagy, Iz, Matthew E. Peters and Arman Cohan. 2020. "Longformer: The Long-Document Transformer." *ArXiv preprint* abs/2004.05150.
URL: *https://arxiv.org/abs/2004.05150*

Bengio, Yoshua, Aaron Courville and Pascal Vincent. 2013. "Representation learning: A review and new perspectives." *IEEE transactions on pattern analysis and machine intelligence* 35(8):1798–1828.

Bettelheim, Bruno. 2010. *The uses of enchantment: The meaning and importance of fairy tales*. Vintage.

Bi, Qiwei, Haoyuan Li and Hanfang Yang. 2021. Boosting Few-shot Abstractive Summarization with Auxiliary Tasks. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. pp. 2888–2893.

Bian, Ning, Xianpei Han, Bo Chen, Hongyu Lin, Ben He and Le Sun. 2021. "Bridging the gap between language model and reading comprehension: Unsupervised mrc via self-supervision." *ArXiv preprint* abs/2107.08582.
URL: *https://arxiv.org/abs/2107.08582*

Brahman, Faeze, Meng Huang, Oyvind Tafjord, Chao Zhao, Mrinmaya Sachan and Snigdha Chaturvedi. 2021. "Let Your Characters Tell Their Story": A Dataset for Character-Centric Narrative Understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2021*. Punta Cana, Dominican Republic: Association for Computational Linguistics pp. 1734–1752.
URL: *https://aclanthology.org/2021.findings-emnlp.150*

Bražinskas, Arthur, Mirella Lapata and Ivan Titov. 2020. Few-Shot Learning for Opinion Summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics pp. 4119–4135.
URL: *https://aclanthology.org/2020.emnlp-main.337*

Brin, Sergey and Lawrence Page. 1998. "The anatomy of a large-scale hypertextual web search engine." *Computer networks and ISDN systems* 30(1-7):107–117.

Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever and Dario Amodei. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020,*

virtual, ed. Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan and Hsuan-Tien Lin.
URL: *https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html*

Bruner, Jerome. 1997. The culture of education. In *The Culture of Education*. Harvard university press.

Cao, Ziqiang, Furu Wei, Sujian Li, Wenjie Li, Ming Zhou and Houfeng Wang. 2015. Learning Summary Prior Representation for Extractive Summarization. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*. Beijing, China: Association for Computational Linguistics pp. 829–833.
URL: *https://aclanthology.org/P15-2136*

Cao, Ziqiang, Wenjie Li, Sujian Li and Furu Wei. 2017. Improving Multi-Document Summarization via Text Classification. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, ed. Satinder P. Singh and Shaul Markovitch. AAAI Press pp. 3053–3059.
URL: *http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14525*

Cao, Ziqiang, Wenjie Li, Sujian Li and Furu Wei. 2018. Retrieve, Rerank and Rewrite: Soft Template Based Neural Summarization. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics pp. 152–161.
URL: *https://aclanthology.org/P18-1015*

Carbonell, Jaime and Jade Goldstein. 1998. The Use of MMR, Diversity-Based Reranking for Reordering Documents and Producing Summaries. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '98 New York, NY, USA: Association for Computing Machinery p. 335–336.
URL: *https://doi.org/10.1145/290941.291025*

Carletta, Jean, 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*. Springer pp. 28–39.

Chapuis, Emile, Pierre Colombo, Matteo Manica, Matthieu Labeau and Chloé Clavel. 2020. Hierarchical Pre-training for Sequence Labelling in Spoken Dialog. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics pp. 2636–2648.
URL: *https://aclanthology.org/2020.findings-emnlp.239*

Chen, Jiaao and Diyi Yang. 2021*a*. Simple Conversational Data Augmentation for Semi-supervised Abstractive Dialogue Summarization. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics pp. 6605–6616.
URL: *https://aclanthology.org/2021.emnlp-main.530*

Chen, Jiaao and Diyi Yang. 2021*b*. Structure-Aware Abstractive Conversation Summarization via Discourse and Action Graphs. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for

Computational Linguistics pp. 1380–1391.
URL: *https://aclanthology.org/2021.naacl-main.109*

Chen, Jifan, Eunsol Choi and Greg Durrett. 2021. Can NLI Models Verify QA Systems' Predictions? In *Findings of the Association for Computational Linguistics: EMNLP 2021*. Punta Cana, Dominican Republic: Association for Computational Linguistics pp. 3841–3854.
URL: *https://aclanthology.org/2021.findings-emnlp.324*

Chen, Mingda, Zewei Chu, Sam Wiseman and Kevin Gimpel. 2022. SummScreen: A Dataset for Abstractive Screenplay Summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics pp. 8602–8615.
URL: *https://aclanthology.org/2022.acl-long.589*

Chen, Qian, Xiaodan Zhu, Zhenhua Ling, Si Wei and Hui Jiang. 2016. Distraction-Based Neural Networks for Modeling Documents. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*. IJCAI'16 AAAI Press p. 2754–2760.

Chen, Yi-Syuan and Hong-Han Shuai. 2021. Meta-Transfer Learning for Low-Resource Abstractive Summarization. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. AAAI Press pp. 12692–12700.
URL: *https://ojs.aaai.org/index.php/AAAI/article/view/17503*

Chen, Yulong, Yang Liu, Liang Chen and Yue Zhang. 2021. DialogSum: A Real-Life Scenario Dialogue Summarization Dataset. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Online: Association for Computational Linguistics pp. 5062–5074.
URL: *https://aclanthology.org/2021.findings-acl.449*

Cheng, Jianpeng and Mirella Lapata. 2016. Neural Summarization by Extracting Sentences and Words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics pp. 484–494.
URL: *https://aclanthology.org/P16-1046*

Chiang, Wei-Lin, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica and Eric P. Xing. 2023. "Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality.".
URL: *https://lmsys.org/blog/2023-03-30-vicuna/*

Chinchor, Nancy. 1992. The Statistical Significance of the MUC-4 Results. In *Fourth Message Uunderstanding Conference (MUC-4): Proceedings of a Conference Held in McLean, Virginia, June 16-18, 1992*.
URL: *https://aclanthology.org/M92-1003*

Cho, Kyunghyun, Bart van Merriënboer, Dzmitry Bahdanau and Yoshua Bengio. 2014. On the Properties of Neural Machine Translation: Encoder–Decoder Approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*. Doha, Qatar: Association for Computational Linguistics pp. 103–111.
URL: *https://aclanthology.org/W14-4012*

Chopra, Sumit, Michael Auli and Alexander M. Rush. 2016. Abstractive Sentence Summarization with Attentive Recurrent Neural Networks. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. San Diego, California: Association for Computational Linguistics pp. 93–98.
URL: *https://aclanthology.org/N16-1012*

Chu, Eric and Peter J. Liu. 2019. MeanSum: A Neural Model for Unsupervised Multi-Document Abstractive Summarization. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, ed. Kamalika Chaudhuri and Ruslan Salakhutdinov. Vol. 97 of *Proceedings of Machine Learning Research* PMLR pp. 1223–1232.
URL: *http://proceedings.mlr.press/v97/chu19b.html*

Cohan, Arman, 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)*. New Orleans, Louisiana: Association for Computational Linguistics pp. 615–621.
URL: *https://aclanthology.org/N18-2097*

Cohn, Trevor and Mirella Lapata. 2008. Sentence Compression Beyond Word Deletion. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*. Manchester, UK: Coling 2008 Organizing Committee pp. 137–144.
URL: *https://aclanthology.org/C08-1018*

Collobert, Ronan, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu and Pavel Kuksa. 2011. "Natural language processing (almost) from scratch." *Journal of machine learning research* 12(ARTICLE):2493–2537.

Conroy, John M and Dianne P O'leary. 2001. Text summarization via hidden markov models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. pp. 406–407.

Conroy, John M., Judith D. Schlesinger and Dianne P. O'Leary. 2006. Topic-Focused Multi-Document Summarization Using an Approximate Oracle Score. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*. Sydney, Australia: Association for Computational Linguistics pp. 152–159.
URL: *https://aclanthology.org/P06-2020*

Consortium, Linguistic Data and New York Times Company. 2008. *The New York Times Annotated Corpus*. LDC corpora Linguistic Data Consortium.
URL: *https://books.google.com/books?id=D4F2AQAACAAJ*

Demszky, Dorottya, Kelvin Guu and Percy Liang. 2018. "Transforming question answering datasets into natural language inference datasets." *ArXiv preprint* abs/1809.02922.
URL: *https://arxiv.org/abs/1809.02922*

Devlin, Jacob, Ming-Wei Chang, Kenton Lee and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics pp. 4171–4186.
URL: *https://aclanthology.org/N19-1423*

Di Fabbrizio, Giuseppe, Amanda Stent and Robert Gaizauskas. 2014. A Hybrid Approach to Multi-document Summarization of Opinions in Reviews. In *Proceedings of the 8th International Natural Language Generation Conference (INLG)*. Philadelphia, Pennsylvania, U.S.A.: Association for Computational Linguistics pp. 54–63.
URL: *https://aclanthology.org/W14-4408*

Ding, Ying and Jing Jiang. 2015. Towards Opinion Summarization from Online Forums. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*. Hissar, Bulgaria: INCOMA Ltd. Shoumen, BULGARIA pp. 138–146.
URL: *https://aclanthology.org/R15-1020*

Edmundson, Harold P. 1969. "New methods in automatic extracting." *Journal of the ACM (JACM)* 16(2):264–285.

Erkan, Günes and Dragomir R. Radev. 2004. "LexRank: Graph-Based Lexical Centrality as Salience in Text Summarization." *Journal of artificial intelligence research* 22(1):457–479.

Ernst, Ori, Avi Caciularu, Ori Shapira, Ramakanth Pasunuru, Mohit Bansal, Jacob Goldberger and Ido Dagan. 2022. Proposition-Level Clustering for Multi-Document Summarization. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Seattle, United States: Association for Computational Linguistics pp. 1765–1779.
URL: *https://aclanthology.org/2022.naacl-main.128*

Fabbri, Alexander, 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*. Florence, Italy: Association for Computational Linguistics pp. 1074–1084.
URL: *https://aclanthology.org/P19-1102*

Fabbri, Alexander, Simeng Han, Haoyuan Li, Haoran Li, Marjan Ghazvininejad, Shafiq Joty, Dragomir Radev and Yashar Mehdad. 2021. Improving Zero and Few-Shot Abstractive Summarization with Intermediate Fine-tuning and Data Augmentation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics pp. 704–717.
URL: *https://aclanthology.org/2021.naacl-main.57*

Fast, Ethan, Binbin Chen and Michael S. Bernstein. 2017. Lexicons on Demand: Neural Word Embeddings for Large-Scale Text Analysis. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017*, ed. Carles Sierra. ijcai.org pp. 4836–4840.
URL: *https://doi.org/10.24963/ijcai.2017/677*

Feng, Vanessa Wei and Graeme Hirst. 2012. Text-level Discourse Parsing with Rich Linguistic Features. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Jeju Island, Korea: Association for Computational Linguistics pp. 60–68.
URL: *https://aclanthology.org/P12-1007*

Feng, Xiachong, Xiaocheng Feng, Bing Qin and Xinwei Geng. 2021. Dialogue Discourse-Aware Graph Model and Data Augmentation for Meeting Summarization. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, ed. Zhi-Hua Zhou. International Joint Conferences on Artificial Intelligence Organization pp. 3808–3814. Main Track.
URL: *https://doi.org/10.24963/ijcai.2021/524*

Feng, Xiachong, Xiaocheng Feng, Libo Qin, Bing Qin and Ting Liu. 2021. Language Model as an Annotator: Exploring DialoGPT for Dialogue Summarization. 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)*. Online: Association for Computational Linguistics pp. 1479–1491.
URL: *https://aclanthology.org/2021.acl-long.117*

Filatova, Elena and Vasileios Hatzivassiloglou. 2004. A Formal Model for Information Selection in Multi-Sentence Text Extraction. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*. Geneva, Switzerland: COLING pp. 397–403.
URL: *https://aclanthology.org/C04-1057*

Filippova, Katja. 2010. Multi-Sentence Compression: Finding Shortest Paths in Word Graphs. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*. Beijing, China: Coling 2010 Organizing Committee pp. 322–330.
URL: *https://aclanthology.org/C10-1037*

Filippova, Katja and Michael Strube. 2008. Sentence Fusion via Dependency Graph Compression. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*. Honolulu, Hawaii: Association for Computational Linguistics pp. 177–185.
URL: *https://aclanthology.org/D08-1019*

Finn, Chelsea, Pieter Abbeel and Sergey Levine. 2017. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, ed. Doina Precup and Yee Whye Teh. Vol. 70 of *Proceedings of Machine Learning Research* PMLR pp. 1126–1135.
URL: *http://proceedings.mlr.press/v70/finn17a.html*

Forster, Edward Morgan. 1985. *Aspects of the Novel*. Vol. 19 Houghton Mifflin Harcourt.

Freytag, Gustav. 1908. *Freytag's technique of the drama: an exposition of dramatic composition and art*. Scott, Foresman and Company.

Fu, Xiyan, Yating Zhang, Tianyi Wang, Xiaozhong Liu, Changlong Sun and Zhenglu Yang. 2021. RepSum: Unsupervised Dialogue Summarization based on Replacement Strategy. 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)*. Online: Association for Computational Linguistics pp. 6042–6051.
URL: *https://aclanthology.org/2021.acl-long.471*

Fuentes, Maria, Enrique Alfonseca and Horacio Rodríguez. 2007. Support Vector Machines for Query-focused Summarization trained and evaluated on Pyramid data. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*. Prague, Czech Republic: Association for Computational Linguistics pp. 57–60.
URL: *https://aclanthology.org/P07-2015*

Galley, Michel. 2006. A Skip-Chain Conditional Random Field for Ranking Meeting Utterances by Importance. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*. Sydney, Australia: Association for Computational Linguistics pp. 364–372.
URL: *https://aclanthology.org/W06-1643*

Gehrmann, Sebastian, Yuntian Deng and Alexander Rush. 2018. Bottom-Up Abstractive Summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics pp. 4098–4109.
URL: *https://aclanthology.org/D18-1443*

Genette, Gérard. 1983. *Narrative discourse: An essay in method*. Vol. 3 Cornell University Press.

Gidiotis, Alexios and Grigorios Tsoumakas. 2019. Structured summarization of academic publications. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer pp. 636–645.

Gidiotis, Alexios and Grigorios Tsoumakas. 2020. "A divide-and-conquer approach to the summarization of long documents." *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 28:3029–3040.

Gillick, Dan and Benoit Favre. 2009. A Scalable Global Model for Summarization. In *Proceedings of the Workshop on Integer Linear Programming for Natural Language Processing*. Boulder, Colorado: Association for Computational Linguistics pp. 10–18.
URL: *https://aclanthology.org/W09-1802*

Gliwa, Bogdan, 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*. Hong Kong, China: Association for Computational Linguistics pp. 70–79.
URL: *https://aclanthology.org/D19-5409*

Goldstein, Jade, Vibhu Mittal, Jaime Carbonell and Mark Kantrowitz. 2000. Multi-Document Summarization By Sentence Extraction. In *NAACL-ANLP 2000 Workshop: Automatic Summarization*.
URL: *https://aclanthology.org/W00-0405*

Gong, Yihong and Xin Liu. 2001. Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. pp. 19–25.

Goodwin, Travis, Max Savery and Dina Demner-Fushman. 2020. Towards Zero-Shot Conditional Summarization with Adaptive Multi-Task Fine-Tuning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics pp. 3215–3226.
URL: *https://aclanthology.org/2020.findings-emnlp.289*

Gorinski, Philip John 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*. Denver, Colorado: Association for Computational Linguistics pp. 1066–1076.
URL: *https://aclanthology.org/N15-1113*

Goyal, Priya, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia and Kaiming He. 2017. "Accurate, large minibatch sgd: Training imagenet in 1 hour." *ArXiv preprint* abs/1706.02677.
URL: *https://arxiv.org/abs/1706.02677*

Goyal, Tanya, Junyi Jessy Li and Greg Durrett. 2022. "News summarization and evaluation in the era of gpt-3." *ArXiv preprint* abs/2209.12356.
URL: *https://arxiv.org/abs/2209.12356*

Grellet, Francoise. 1981. *Developing reading skills: A practical guide to reading comprehension exercises*. Cambridge university press.

Grusky, Max, 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)*. New Orleans, Louisiana: Association for Computational Linguistics pp. 708–719.
URL: *https://aclanthology.org/N18-1065*

Gu, Jiatao, Zhengdong Lu, Hang Li and Victor O.K. Li. 2016. Incorporating Copying Mechanism in Sequence-to-Sequence Learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Berlin, Germany: Association for Computational Linguistics pp. 1631–1640.
URL: *https://aclanthology.org/P16-1154*

Hakkani-Tur, Dilek and Gokhan Tur. 2007. Statistical Sentence Extraction for Information Distillation. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07*. Vol. 4 pp. IV–1–IV–4.

He, Ruidan, Wee Sun Lee, Hwee Tou Ng and Daniel Dahlmeier. 2017. An Unsupervised Neural Attention Model for Aspect Extraction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics pp. 388–397.
URL: *https://aclanthology.org/P17-1036*

He, Ruining and Julian J. McAuley. 2016. Ups and Downs: Modeling the Visual Evolution of Fashion Trends with One-Class Collaborative Filtering. In *Proceedings of the 25th International Conference on World Wide Web, WWW 2016, Montreal, Canada, April 11 - 15, 2016*, ed. Jacqueline Bourdeau, Jim Hendler, Roger Nkambou, Ian Horrocks and Ben Y. Zhao. ACM pp. 507–517.
URL: *https://doi.org/10.1145/2872427.2883037*

Hicks, Wynford, Adams Sally, Harriett Gilbert, Tim Holmes and Jane Bentley. 2016. *Writing for journalists*. Routledge.

Hill, Felix, Antoine Bordes, Sumit Chopra and Jason Weston. 2016. The Goldilocks Principle: Reading Children's Books with Explicit Memory Representations. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, ed. Yoshua Bengio and Yann LeCun.
URL: *http://arxiv.org/abs/1511.02301*

Hirschman, Lynette, Marc Light, Eric Breck and John D. Burger. 1999. Deep Read: A Reading Comprehension System. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*. College Park, Maryland, USA: Association for Computational Linguistics pp. 325–332.
URL: *https://aclanthology.org/P99-1042*

Hochreiter, Sepp and Jürgen Schmidhuber. 1997. "Long short-term memory." *Neural computation* 9(8):1735–1780.

Hong, Kai and Ani Nenkova. 2014. Improving the Estimation of Word Importance for News Multi-Document Summarization. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*. Gothenburg, Sweden: Association for Computational Linguistics pp. 712–721.
URL: *https://aclanthology.org/E14-1075*

Hovy, Eduard and ChinYew Lin. 1997. Automated Text Summarization in SUMMARIST. In *Intelligent Scalable Text Summarization*.
URL: *https://aclanthology.org/W97-0704*

Hu, Baotian, Qingcai Chen and Fangze Zhu. 2015. LCSTS: A Large Scale Chinese Short Text Summarization Dataset. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics pp. 1967–1972.
URL: *https://aclanthology.org/D15-1229*

Hu, Edward J., Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang and Weizhu Chen. 2022. LoRA: Low-Rank Adaptation of Large Language Models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.
URL: *https://openreview.net/forum?id=nZeVKeeFYf9*

Hu, Minqing and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on KDD*. ACM pp. 168–177.

Huang, Qingqiu, Yu Xiong, Anyi Rao, Jiaze Wang and Dahua Lin. 2020. Movienet: A holistic dataset for movie understanding. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*. Springer pp. 709–727.

Huh, Taehun and Youngjoong Ko. 2022. Lightweight Meta-Learning for Low-Resource Abstractive Summarization. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*. pp. 2629–2633.

Iskender, Neslihan, Tim Polzehl and Sebastian Möller. 2021. Reliability of Human Evaluation for Text Summarization: Lessons Learned and Challenges Ahead. In *Proceedings of the Workshop on Human Evaluation of NLP Systems (HumEval)*. Online: Association for Computational Linguistics pp. 86–96.
URL: *https://aclanthology.org/2021.humeval-1.10*

Jia, Ruipeng, Yanan Cao, Hengzhu Tang, Fang Fang, Cong Cao and Shi Wang. 2020. Neural Extractive Summarization with Hierarchical Attentive Heterogeneous Graph Network. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics pp. 3622–3631.
URL: *https://aclanthology.org/2020.emnlp-main.295*

Jin, Di, Shuyang Gao, Jiun-Yu Kao, Tagyoung Chung and Dilek Hakkani-Tür. 2020. MMM: Multi-Stage Multi-Task Learning for Multi-Choice Reading Comprehension. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*. AAAI Press pp. 8010–8017.
URL: *https://aaai.org/ojs/index.php/AAAI/article/view/6310*

Jin, Hanqi, Tianming Wang and Xiaojun Wan. 2020. Multi-Granularity Interaction Network for Extractive and Abstractive Multi-Document Summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics pp. 6244–6254.
URL: *https://aclanthology.org/2020.acl-main.556*

Jin, Hanqi and Xiaojun Wan. 2020. Abstractive Multi-Document Summarization via Joint Learning with Single-Document Summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2020*. Online: Association for Computational Linguistics pp. 2545–2554.
URL: *https://aclanthology.org/2020.findings-emnlp.231*

Jing, Hongyan. 2000. Sentence Reduction for Automatic Text Summarization. In *Sixth Applied Natural Language Processing Conference*. Seattle, Washington, USA: Association for Computational Linguistics pp. 310–315.
URL: *https://aclanthology.org/A00-1043*

Jing, Hongyan and Kathleen R McKeown. 1999. The decomposition of human-written summary sentences. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. pp. 129–136.

Jing, Hongyan and Kathleen R. McKeown. 2000. Cut and Paste Based Text Summarization. In *1st Meeting of the North American Chapter of the Association for Computational Linguistics*.
URL: *https://aclanthology.org/A00-2024*

Joshi, Mandar, Eunsol Choi, Daniel Weld and Luke Zettlemoyer. 2017. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics pp. 1601–1611.
URL: *https://aclanthology.org/P17-1147*

Kågebäck, Mikael, Olof Mogren, Nina Tahmasebi and Devdatt Dubhashi. 2014. Extractive Summarization using Continuous Vector Space Models. In *Proceedings of the 2nd Workshop on Continuous Vector Space Models and their Compositionality (CVSC)*. Gothenburg, Sweden: Association for Computational Linguistics pp. 31–39.
URL: *https://aclanthology.org/W14-1504*

Kan, Min-Yen and Kathleen R. McKeown. 2002. Corpus-trained Text Generation for Summarization. In *Proceedings of the International Natural Language Generation Conference*. Harriman, New York, USA: Association for Computational Linguistics pp. 1–8.
URL: *https://aclanthology.org/W02-2101*

Karamanolakis, Giannis, Daniel Hsu and Luis Gravano. 2019. Training Neural Networks for Aspect Extraction Using Descriptive Keywords Only. In *The 2nd Learning from Limited Labeled Data (LLD) Workshop*.

Kedzie, Chris, Kathleen McKeown and Hal Daumé III. 2018. Content Selection in Deep Learning Models of Summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics pp. 1818–1828.
URL: *https://aclanthology.org/D18-1208*

Kim, Byeongchang, Hyunwoo Kim and Gunhee Kim. 2019. Abstractive Summarization of Reddit Posts with Multi-level Memory Networks. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics pp. 2519–2531.
URL: *https://aclanthology.org/N19-1260*

Kim, Hyun Duk, Kavita Ganesan, Parikshit Sondhi and ChengXiang Zhai. 2011. Comprehensive review of opinion summarization. Technical report.

Kingma, Diederik P. and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, ed. Yoshua Bengio and Yann LeCun.
URL: *http://arxiv.org/abs/1412.6980*

Kipf, Thomas N. and Max Welling. 2017. Semi-Supervised Classification with Graph Convolutional Networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.
URL: *https://openreview.net/forum?id=SJU4ayYgl*

Knoespel, Kenneth J. 1991. "The emplotment of chaos: instability and narrative order." *Chaos and order: Complex dynamics in literature and science* pp. 100–122.

Kočiský, Tomáš, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis and Edward Grefenstette. 2018. "The NarrativeQA Reading Comprehension Challenge." *Transactions of the Association for Computational Linguistics* 6:317–328.
URL: *https://aclanthology.org/Q18-1023*

Koehn, Philipp and Christof Monz. 2006. Manual and Automatic Evaluation of Machine Translation between European Languages. In *Proceedings on the Workshop on Statistical Machine Translation*. New York City: Association for Computational Linguistics pp. 102–121.
URL: *https://aclanthology.org/W06-3114*

Kryscinski, Wojciech, Nazneen Rajani, Divyansh Agarwal, Caiming Xiong and Dragomir Radev. 2022. BOOKSUM: A Collection of Datasets for Long-form Narrative Summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics pp. 6536–6558.
URL: *https://aclanthology.org/2022.findings-emnlp.488*

Kupiec, Julian, Jan Pedersen and Francine Chen. 1995. A trainable document summarizer. In *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*. pp. 68–73.

Kwiatkowski, Tom, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le and Slav Petrov. 2019. "Natural Questions: A Benchmark for Question Answering Research." *Transactions of the Association for Computational Linguistics* 7:452–466.
URL: *https://aclanthology.org/Q19-1026*

Laban, Philippe, Tobias Schnabel, Paul N. Bennett and Marti A. Hearst. 2022. "SummaC: Re-Visiting NLI-based Models for Inconsistency Detection in Summarization." *Transactions of the Association for Computational Linguistics* 10:163–177.
URL: *https://aclanthology.org/2022.tacl-1.10*

Ladhak, Faisal, 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*. Online: Association for Computational Linguistics pp. 5043–5054.
URL: *https://aclanthology.org/2020.acl-main.453*

Lebanoff, Logan, Kaiqiang Song and Fei Liu. 2018. Adapting the Neural Encoder-Decoder Framework from Single to Multi-Document Summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Brussels, Belgium: Association for Computational Linguistics pp. 4131–4141.
URL: *https://aclanthology.org/D18-1446*

LeCun, Yann, Bernhard Boser, John Denker, Donnie Henderson, Richard Howard, Wayne Hubbard and Lawrence Jackel. 1989. "Handwritten digit recognition with a back-propagation network." *Advances in neural information processing systems* 2.

Lehnert, Wendy G. 1981. "Plot units and narrative summarization." *Cognitive science* 5(4):293–331.

Lei, Jie, Licheng Yu, Tamara Berg and Mohit Bansal. 2020. What is More Likely to Happen Next? Video-and-Language Future Event Prediction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics pp. 8769–8784.
URL: *https://aclanthology.org/2020.emnlp-main.706*

Lewis, Mike, 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*. Online: Association for Computational Linguistics pp. 7871–7880.
URL: *https://aclanthology.org/2020.acl-main.703*

Lewis, Patrick, Yuxiang Wu, Linqing Liu, Pasquale Minervini, Heinrich Küttler, Aleksandra Piktus, Pontus Stenetorp and Sebastian Riedel. 2021. "PAQ: 65 Million Probably-Asked Questions and What You Can Do With Them." *Transactions of the Association for Computational Linguistics* 9:1098–1115.
URL: *https://aclanthology.org/2021.tacl-1.65*

Li, Piji, Wai Lam, Lidong Bing and Zihao Wang. 2017. Deep Recurrent Generative Decoder for Abstractive Text Summarization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Copenhagen, Denmark: Association for Computational Linguistics pp. 2091–2100.
URL: *https://aclanthology.org/D17-1222*

Li, Siyao, Deren Lei, Pengda Qin and William Yang Wang. 2019. Deep Reinforcement Learning with Distributional Semantic Rewards for Abstractive Summarization. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics pp. 6038–6044.
URL: *https://aclanthology.org/D19-1623*

Li, Wei, Xinyan Xiao, Jiachen Liu, Hua Wu, Haifeng Wang and Junping Du. 2020. Leveraging Graph to Improve Abstractive Multi-Document Summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics pp. 6232–6243.
URL: *https://aclanthology.org/2020.acl-main.555*

Lin, Chin-Yew. 2004. ROUGE: A Package for Automatic Evaluation of Summaries. In *Text Summarization Branches Out*. Barcelona, Spain: Association for Computational Linguistics pp. 74–81.
URL: *https://aclanthology.org/W04-1013*

Lin, Chin-Yew and Eduard Hovy. 2002. From Single to Multi-document Summarization. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics pp. 457–464.
URL: *https://aclanthology.org/P02-1058*

Lin, Hui and Jeff Bilmes. 2010. Multi-document Summarization via Budgeted Maximization of Submodular Functions. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Los Angeles, California: Association for Computational Linguistics pp. 912–920.
URL: *https://aclanthology.org/N10-1134*

Linebarger, Deborah L and Jessica Taylor Piotrowski. 2009. "TV as storyteller: How exposure to television narratives impacts at-risk preschoolers' story knowledge and narrative skills." *British journal of developmental psychology* 27(1):47–69.

Lison, Pierre, Jörg Tiedemann and Milen Kouylekov. 2018. OpenSubtitles2018: Statistical Rescoring of Sentence Alignments in Large, Noisy Parallel Corpora. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*. Miyazaki, Japan: European Language Resources Association (ELRA).
URL: *https://aclanthology.org/L18-1275*

Liu, Bing. 2015. *Sentiment analysis: Mining opinions, sentiments, and emotions*. Cambridge University Press.

Liu, Junpeng, Yanyan Zou, Hainan Zhang, Hongshen Chen, Zhuoye Ding, Caixia Yuan and Xiaojie Wang. 2021. Topic-Aware Contrastive Learning for Abstractive Dialogue Summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2021*. Punta Cana, Dominican Republic: Association for Computational Linguistics pp. 1229–1243.
URL: *https://aclanthology.org/2021.findings-emnlp.106*

Liu, Pengfei, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi and Graham Neubig. 2021. "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing." *ArXiv preprint* abs/2107.13586.
URL: *https://arxiv.org/abs/2107.13586*

Liu, Peter J., Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser and Noam Shazeer. 2018. Generating Wikipedia by Summarizing Long Sequences. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.
URL: *https://openreview.net/forum?id=Hyg0vbWC-*

Liu, Yang and Mirella Lapata. 2019*a*. Hierarchical Transformers for Multi-Document Summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics pp. 5070–5081.
URL: *https://aclanthology.org/P19-1500*

Liu, Yang and Mirella Lapata. 2019*b*. Text Summarization with Pretrained Encoders. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics pp. 3730–3740.
URL: *https://aclanthology.org/D19-1387*

Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer and Veselin Stoyanov. 2019. "RoBERTa: A Robustly Optimized BERT Pretraining Approach." *ArXiv preprint* abs/1907.11692.
URL: *https://arxiv.org/abs/1907.11692*

Liu, Yongkang, Shi Feng, Daling Wang, Kaisong Song, Feiliang Ren and Yifei Zhang. 2021. A Graph Reasoning Network for Multi-turn Response Selection via Customized Pre-training. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. AAAI Press pp. 13433–13442.
URL: *https://ojs.aaai.org/index.php/AAAI/article/view/17585*

Loshchilov, Ilya and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.
URL: *https://openreview.net/forum?id=Bkg6RiCqY7*

Louis, Annie, Aravind Joshi and Ani Nenkova. 2010. Discourse indicators for content selection in summarization. In *Proceedings of the SIGDIAL 2010 Conference*. Tokyo, Japan: Association for Computational Linguistics pp. 147–156.
URL: *https://aclanthology.org/W10-4327*

Lu, Yao, Yue Dong and Laurent Charlin. 2020. Multi-XScience: A Large-scale Dataset for Extreme Multi-document Summarization of Scientific Articles. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics pp. 8068–8074.
URL: *https://aclanthology.org/2020.emnlp-main.648*

Luhn, Hans Peter. 1958. "The automatic creation of literature abstracts." *IBM Journal of research and development* 2(2):159–165.

Lyle, Sue. 2000. "Narrative understanding: Developing a theoretical context for understanding how children make meaning in classroom settings." *Journal of Curriculum Studies* 32(1):45–63.

Ma, Kaixin, 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)*. New Orleans, Louisiana: Association for Computational Linguistics pp. 2039–2048.
URL: *https://aclanthology.org/N18-1185*

Maaten, Laurens van der and Geoffrey Hinton. 2008. "Visualizing data using t-SNE." *JMLR* 9(Nov):2579–2605.

Magooda, Ahmed and Diane Litman. 2020. Abstractive summarization for low resource data using domain transfer and data synthesis. In *The Thirty-Third International Flairs Conference*.

Mani, Inderjeet. 2012. "Computational modeling of narrative." *Synthesis Lectures on Human Language Technologies* 5(3):1–142.

Mani, Inderjeet, Barbara Gates and Eric Bloedorn. 1999. Improving Summaries by Revising Them. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*. College Park, Maryland, USA: Association for Computational Linguistics pp. 558–565.
URL: *https://aclanthology.org/P99-1072*

Mani, Inderjeet and Eric Bloedorn. 1997. Multi-document summarization by graph search and matching. In *Proceedings of the fourteenth national conference on artificial intelligence and ninth conference on Innovative applications of artificial intelligence*. pp. 622–628.

Mani, Inderjeet and Eric Bloedorn. 1998. Machine learning of generic and user-focused summarization. In *AAAI/IAAI*. pp. 821–826.

Marcu, Daniel. 1999. The Automatic Construction of Large-Scale Corpora for Summarization Research. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '99 New York, NY, USA: Association for Computing Machinery p. 137–144.
URL: *https://doi.org/10.1145/312624.312668*

McDonald, Ryan. 2007. A study of global inference algorithms in multi-document summarization. In *European Conference on Information Retrieval*. Springer pp. 557–564.

McKeown, Kathleen and Dragomir R. Radev. 1995. Generating Summaries of Multiple News Articles. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. SIGIR '95 New York, NY, USA: Association for Computing Machinery p. 74–82.
URL: *https://doi.org/10.1145/215206.215334*

McKeown, Kathleen R, Judith Klavans, Vasileios Hatzivassiloglou, Regina Barzilay and Eleazar Eskin. 1999. Towards Multidocument Summarization by Reformulation: Progress and Prospects. In *AAAI/IAAI*.

McKeown, Kathleen, Rebecca J Passonneau, David K Elson, Ani Nenkova and Julia Hirschberg. 2005. Do summaries help? In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. pp. 210–217.

Mihalcea, Rada and Paul Tarau. 2004. TextRank: Bringing Order into Text. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*. Barcelona, Spain: Association for Computational Linguistics pp. 404–411.
URL: *https://aclanthology.org/W04-3252*

Mikolov, Tomás, Ilya Sutskever, Kai Chen, Gregory S. Corrado and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and their Compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, ed. Christopher J. C. Burges, Léon Bottou, Zoubin Ghahramani and Kilian Q. Weinberger. pp. 3111–3119.
URL: *https://proceedings.neurips.cc/paper/2013/hash/9aa42b31882ec039965f3c4923ce901b-Abstract.html*

Moro, Gianluca and Luca Ragazzi. 2022. Semantic Self-Segmentation for Abstractive Summarization of Long Documents in Low-Resource Regimes. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*. AAAI Press pp. 11085–11093.
URL: *https://ojs.aaai.org/index.php/AAAI/article/view/21357*

Nallapati, Ramesh, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre and Bing Xiang. 2016. Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*. Berlin, Germany: Association for Computational Linguistics pp. 280–290.
URL: *https://aclanthology.org/K16-1028*

Nallapati, Ramesh, Feifei Zhai and Bowen Zhou. 2017. SummaRuNNer: A Recurrent Neural Network Based Sequence Model for Extractive Summarization of Documents. In *Proceedings of the Thirty-First AAAI*

Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA, ed. Satinder P. Singh and Shaul Markovitch. AAAI Press pp. 3075–3081.
URL: *http://aaai.org/ocs/index.php/AAAI/AAAI17/paper/view/14636*

Narayan, Shashi, Joshua Maynez, Jakub Adamek, Daniele Pighin, Blaz Bratanic and Ryan McDonald. 2020. Stepwise Extractive Summarization and Planning with Structured Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics pp. 4143–4159.
URL: *https://aclanthology.org/2020.emnlp-main.339*

Narayan, Shashi, 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*. Brussels, Belgium: Association for Computational Linguistics pp. 1797–1807.
URL: *https://aclanthology.org/D18-1206*

Nash, Cristopher. 2005. *Narrative in culture: The uses of storytelling in the sciences, philosophy and literature*. Routledge.

Nayeem, Mir Tafseer, Tanvir Ahmed Fuad and Yllias Chali. 2018. Abstractive Unsupervised Multi-Document Summarization using Paraphrastic Sentence Fusion. In *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics pp. 1191–1204.
URL: *https://aclanthology.org/C18-1102*

Nenkova, Ani. 2008. Entity-driven Rewrite for Multi-document Summarization. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I*.
URL: *https://aclanthology.org/I08-1016*

Nenkova, Ani and Lucy Vanderwende. 2005. "The impact of frequency on summarization." *Microsoft Research, Redmond, Washington, Tech. Rep. MSR-TR-2005* 101.

Noreen, Eric W. 1989. *Computer-intensive methods for testing hypotheses*. Wiley New York.

Otterbacher, Jahna C., Dragomir R. Radev and Airong Luo. 2002. Revisions that improve cohesion in multi-document summaries: a preliminary study. In *Proceedings of the ACL-02 Workshop on Automatic Summarization*. Phildadelphia, Pennsylvania, USA: Association for Computational Linguistics pp. 27–44.
URL: *https://aclanthology.org/W02-0404*

Ouyang, Jessica, Serina Chang and Kathy McKeown. 2017. Crowd-Sourced Iterative Annotation for Narrative Summarization Corpora. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Valencia, Spain: Association for Computational Linguistics pp. 46–51.
URL: *https://aclanthology.org/E17-2008*

Pan, Sinno Jialin and Qiang Yang. 2009. "A survey on transfer learning." *IEEE Transactions on knowledge and data engineering* 22(10):1345–1359.

Papalampidi, Pinelopi, 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*. Online: Association for Computational Linguistics pp. 1920–1933.
URL: *https://aclanthology.org/2020.acl-main.174*

Papalampidi, Pinelopi, Frank Keller and Mirella Lapata. 2019. Movie Plot Analysis via Turning Point Identification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics pp. 1707–1717.
URL: *https://aclanthology.org/D19-1180*

Parida, Shantipriya and Petr Motlicek. 2019. Abstract Text Summarization: A Low Resource Challenge. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics pp. 5994–5998.
URL: *https://aclanthology.org/D19-1616*

Pasunuru, Ramakanth, Mengwen Liu, Mohit Bansal, Sujith Ravi and Markus Dreyer. 2021. Efficiently Summarizing Text and Graph Encodings of Multi-Document Clusters. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics pp. 4768–4779.
URL: *https://aclanthology.org/2021.naacl-main.380*

Pasunuru, Ramakanth and Mohit Bansal. 2018. Multi-Reward Reinforced Summarization with Saliency and Entailment. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. New Orleans, Louisiana: Association for Computational Linguistics pp. 646–653.
URL: *https://aclanthology.org/N18-2102*

Pennington, Jeffrey, Richard Socher and Christopher Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Doha, Qatar: Association for Computational Linguistics pp. 1532–1543.
URL: *https://aclanthology.org/D14-1162*

Piper, Andrew, Richard Jean So and David Bamman. 2021. Narrative Theory for Computational Narrative Understanding. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics pp. 298–311.
URL: *https://aclanthology.org/2021.emnlp-main.26*

Polkinghorne, Donald E. 1988. *Narrative knowing and the human sciences*. Suny Press.

Pollock, Joseph J and Antonio Zamora. 1975. "Automatic abstracting research at chemical abstracts service." *Journal of Chemical Information and Computer Sciences* 15(4):226–232.

Prince, Gerald. 1973. *A Grammar of Stories: An Introduction*.

Qiu, Yao, Jinchao Zhang and Jie Zhou. 2021. Different Strokes for Different Folks: Investigating Appropriate Further Pre-training Approaches for Diverse Dialogue Tasks. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics pp. 2318–2327.
URL: *https://aclanthology.org/2021.emnlp-main.178*

Radev, Dragomir R., Hongyan Jing and Malgorzata Budzikowska. 2000. Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. In *NAACL-ANLP 2000 Workshop: Automatic Summarization*.
URL: *https://aclanthology.org/W00-0403*

Radev, Dragomir R, Hongyan Jing, Małgorzata Styś and Daniel Tam. 2004. "Centroid-based summarization of multiple documents." *Information Processing & Management* 40(6):919–938.

Radford, Alec, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever et al. 2019. "Language models are unsupervised multitask learners." *OpenAI blog* 1(8):9.

Raffel, Colin, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li and Peter J. Liu. 2020. "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer." *J. Mach. Learn. Res.* 21:140:1–140:67.
URL: *http://jmlr.org/papers/v21/20-074.html*

Rajpurkar, Pranav, Jian Zhang, Konstantin Lopyrev and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Austin, Texas: Association for Computational Linguistics pp. 2383–2392.
URL: *https://aclanthology.org/D16-1264*

Ram, Ori, Yuval Kirstain, Jonathan Berant, Amir Globerson and Omer Levy. 2021. Few-Shot Question Answering by Pretraining Span Selection. 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)*. Online: Association for Computational Linguistics pp. 3066–3079.
URL: *https://aclanthology.org/2021.acl-long.239*

Rameshkumar, Revanth and Peter Bailey. 2020. Storytelling with Dialogue: A Critical Role Dungeons and Dragons Dataset. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics pp. 5121–5134.
URL: *https://aclanthology.org/2020.acl-main.459*

Richardson, Matthew, Christopher J.C. Burges and Erin Renshaw. 2013. MCTest: A Challenge Dataset for the Open-Domain Machine Comprehension of Text. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Seattle, Washington, USA: Association for Computational Linguistics pp. 193–203.
URL: *https://aclanthology.org/D13-1020*

Riedhammer, Korbinian, Benoit Favre and Dilek Hakkani-Tür. 2010. "Long story short–global unsupervised models for keyphrase based meeting summarization." *Speech Communication* 52(10):801–815.

Rogers, Anna, Olga Kovaleva, Matthew Downey and Anna Rumshisky. 2020. Getting Closer to AI Complete Question Answering: A Set of Prerequisite Real Tasks. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*. AAAI Press pp. 8722–8731.
URL: *https://aaai.org/ojs/index.php/AAAI/article/view/6398*

Rosen, Harold. 1985. *Stories and meanings*. National Association for the Teaching of English.

Rush, Alexander M., 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*. Lisbon, Portugal: Association for Computational Linguistics pp. 379–389.
URL: *https://aclanthology.org/D15-1044*

Schank, Roger C and Robert P Abelson. 2013. *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. Psychology press.

Schumaker, Jean B., Pegi H. Denton and Donald D. Deshler. 1984. *The Paraphrasing Strategy*. University of Kansas.

See, Abigail, Peter J. Liu and Christopher D. Manning. 2017. Get To The Point: Summarization with Pointer-Generator Networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics pp. 1073–1083.
URL: *https://aclanthology.org/P17-1099*

Shen, Dou, Jian-Tao Sun, Hua Li, Qiang Yang and Zheng Chen. 2007. Document summarization using conditional random fields. In *IJCAI*. Vol. 7 pp. 2862–2867.

Shen, Wei, Jianyong Wang and Jiawei Han. 2014. "Entity linking with a knowledge base: Issues, techniques, and solutions." *IEEE Transactions on Knowledge and Data Engineering* 27(2):443–460.

Sipe, Lawrence R. 2001. "A palimpsest of stories: Young children's construction of intertextual links among fairytale variants." *Reading Research and Instruction* 40(4):333–352.
URL: *https://doi.org/10.1080/19388070109558354*

Socher, Richard, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng and Christopher Potts. 2013. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. Seattle, Washington, USA: Association for Computational Linguistics pp. 1631–1642.
URL: *https://aclanthology.org/D13-1170*

Song, Kaiqiang, Lin Zhao and Fei Liu. 2018. Structure-Infused Copy Mechanisms for Abstractive Summarization. In *Proceedings of the 27th International Conference on Computational Linguistics*. Santa Fe, New Mexico, USA: Association for Computational Linguistics pp. 1717–1729.
URL: *https://aclanthology.org/C18-1146*

Sun, Kai, Dian Yu, Jianshu Chen, Dong Yu, Yejin Choi and Claire Cardie. 2019. "DREAM: A Challenge Data Set and Models for Dialogue-Based Reading Comprehension." *Transactions of the Association for Computational Linguistics* 7:217–231.
URL: *https://aclanthology.org/Q19-1014*

Tan, Jiwei, Xiaojun Wan and Jianguo Xiao. 2017. Abstractive Document Summarization with a Graph-Based Attentional Neural Model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics pp. 1171–1181.
URL: *https://aclanthology.org/P17-1108*

Tapaswi, Makarand, Martin Bäuml and Rainer Stiefelhagen. 2015. "Aligning plot synopses to videos for story-based retrieval." *International Journal of Multimedia Information Retrieval* 4(1):3–16.
URL: *http://www.cs.toronto.edu/ makarand/papers/IJMIR*$_{plot - retrieval.pdf}$

Tapaswi, Makarand, Yukun Zhu, Rainer Stiefelhagen, Antonio Torralba, Raquel Urtasun and Sanja Fidler. 2016. MovieQA: Understanding Stories in Movies through Question-Answering. In *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. IEEE Computer Society pp. 4631–4640.
URL: *https://doi.org/10.1109/CVPR.2016.501*

Tay, Yi, Mostafa Dehghani, Vinh Q. Tran, Xavier Garcia, Jason Wei, Xuezhi Wang, Hyung Won Chung, Dara Bahri, Tal Schuster, Steven Zheng, Denny Zhou, Neil Houlsby and Donald Metzler. 2023. UL2: Unifying Language Learning Paradigms. In *The Eleventh International Conference on Learning Representations*.
URL: *https://openreview.net/forum?id=6ruVLB727MC*

Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, ed. Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan and Roman Garnett. pp. 5998–6008.
URL: *https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html*

Velickovic, Petar, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò and Yoshua Bengio. 2018. Graph Attention Networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.
URL: *https://openreview.net/forum?id=rJXMpikCZ*

Wan, Xiaojun and Jianwu Yang. 2008. Multi-document summarization using cluster-based link analysis. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. pp. 299–306.

Wang, Danqing, Pengfei Liu, Yining Zheng, Xipeng Qiu and Xuanjing Huang. 2020. Heterogeneous Graph Neural Networks for Extractive Document Summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics pp. 6209–6219.
URL: *https://aclanthology.org/2020.acl-main.553*

Wang, Hong, Xin Wang, Wenhan Xiong, Mo Yu, Xiaoxiao Guo, Shiyu Chang and William Yang Wang. 2019. Self-Supervised Learning for Contextualized Extractive Summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics pp. 2221–2227.
URL: *https://aclanthology.org/P19-1214*

Wang, Lu and Claire Cardie. 2013. Domain-Independent Abstract Generation for Focused Meeting Summarization. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Sofia, Bulgaria: Association for Computational Linguistics pp. 1395–1405.
URL: *https://aclanthology.org/P13-1137*

Wang, Lu 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*. San Diego, California: Association for Computational Linguistics pp. 47–57.
URL: *https://aclanthology.org/N16-1007*

Weston, Jason, Sumit Chopra and Antoine Bordes. 2015. Memory Networks. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, ed. Yoshua Bengio and Yann LeCun.
URL: *http://arxiv.org/abs/1410.3916*

Wolf, Thomas, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest and Alexander Rush. 2020. Transformers: State-of-the-Art Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics pp. 38–45.
URL: *https://aclanthology.org/2020.emnlp-demos.6*

Wu, Chien-Sheng, Steven C.H. Hoi, Richard Socher and Caiming Xiong. 2020. TOD-BERT: Pre-trained Natural Language Understanding for Task-Oriented Dialogue. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics pp. 917–929.
URL: *https://aclanthology.org/2020.emnlp-main.66*

Wubben, Sander, Antal van den Bosch and Emiel Krahmer. 2012. Sentence Simplification by Monolingual Machine Translation. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Jeju Island, Korea: Association for Computational Linguistics pp. 1015–1024.
URL: *https://aclanthology.org/P12-1107*

Xiao, Wen, Iz Beltagy, Giuseppe Carenini and Arman Cohan. 2022. PRIMERA: Pyramid-based Masked Sentence Pre-training for Multi-document Summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics pp. 5245–5263.
URL: *https://aclanthology.org/2022.acl-long.360*

Xiong, Yu, Qingqiu Huang, Lingfeng Guo, Hang Zhou, Bolei Zhou and Dahua Lin. 2019. A Graph-Based Framework to Bridge Movies and Synopses. In *2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019*. IEEE pp. 4591–4600.
URL: *https://doi.org/10.1109/ICCV.2019.00469*

Xu, Jiacheng, Zhe Gan, Yu Cheng and Jingjing Liu. 2020. Discourse-Aware Neural Extractive Text Summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics pp. 5021–5031.
URL: *https://aclanthology.org/2020.acl-main.451*

Xu, Ying, Dakuo Wang, Mo Yu, Daniel Ritchie, Bingsheng Yao, Tongshuang Wu, Zheng Zhang, Toby Li, Nora Bradford, Branda Sun, Tran Hoang, Yisi Sang, Yufang Hou, Xiaojuan Ma, Diyi Yang, Nanyun Peng, Zhou Yu and Mark Warschauer. 2022. Fantastic Questions and Where to Find Them: FairytaleQA – An Authentic Dataset for Narrative Comprehension. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Dublin, Ireland: Association for Computational Linguistics pp. 447–460.
URL: *https://aclanthology.org/2022.acl-long.34*

Yan, Xiaohui, Jiafeng Guo, Yanyan Lan and Xueqi Cheng. 2013. A biterm topic model for short texts. In *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, ed. Daniel Schwabe, Virgílio A. F. Almeida, Hartmut Glaser, Ricardo Baeza-Yates and Sue B. Moon. International World Wide Web Conferences Steering Committee / ACM pp. 1445–1456.
URL: *https://doi.org/10.1145/2488388.2488514*

Yang, Xianjun, Yan Li, Xinlu Zhang, Haifeng Chen and Wei Cheng. 2023. "Exploring the limits of chatgpt for query or aspect-based text summarization." *ArXiv preprint* abs/2302.08081.
URL: *https://arxiv.org/abs/2302.08081*

Yano, Yasukata, Michael H. Long and Steven Ross. 1994. "The Effects of Simplified and Elaborated Texts on Foreign Language Reading Comprehension." *Language Learning* 44(2):189–219.
URL: *https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-1770.1994.tb01100.x*

Yasunaga, Michihiro, Rui Zhang, Kshitijh Meelu, Ayush Pareek, Krishnan Srinivasan and Dragomir Radev. 2017. Graph-based Neural Multi-Document Summarization. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*. Vancouver, Canada: Association for Computational Linguistics pp. 452–462.
URL: *https://aclanthology.org/K17-1045*

Yeh, Jen-Yuan, Hao-Ren Ke, Wei-Pang Yang and I-Heng Meng. 2005. "Text summarization using a trainable summarizer and latent semantic analysis." *Information processing & management* 41(1):75–95.

Yin, Wenpeng, Dragomir Radev and Caiming Xiong. 2021. DocNLI: A Large-scale Dataset for Document-level Natural Language Inference. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. Online: Association for Computational Linguistics pp. 4913–4922.
URL: *https://aclanthology.org/2021.findings-acl.435*

Yin, Wenpeng and Yulong Pei. 2015. Optimizing Sentence Modeling and Selection for Document Summarization. In *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence, IJCAI 2015, Buenos Aires, Argentina, July 25-31, 2015*, ed. Qiang Yang and Michael J. Wooldridge. AAAI Press pp. 1383–1389.
URL: *http://ijcai.org/Abstract/15/199*

Yu, Tiezheng, Zihan Liu and Pascale Fung. 2021. AdaptSum: Towards Low-Resource Domain Adaptation for Abstractive Summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics pp. 5892–5904.
URL: *https://aclanthology.org/2021.naacl-main.471*

Zajic, David M, Bonnie Dorr, Jimmy Lin and Richard Schwartz. 2006. Sentence compression as a component of a multi-document summarization system. In *Proceedings of the 2006 document understanding workshop, New York*.

Zhang, Jianmin, Jiwei Tan and Xiaojun Wan. 2018. Adapting Neural Single-Document Summarization Model for Abstractive Multi-Document Summarization: A Pilot Study. In *Proceedings of the 11th International Conference on Natural Language Generation*. Tilburg University, The Netherlands: Association for Computational Linguistics pp. 381–390.
URL: *https://aclanthology.org/W18-6545*

Zhang, Jingqing, Yao Zhao, Mohammad Saleh and Peter J. Liu. 2020*a*. PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*. Vol. 119 of *Proceedings of Machine Learning Research* PMLR pp. 11328–11339.
URL: *http://proceedings.mlr.press/v119/zhang20ae.html*

Zhang, Jingqing, Yao Zhao, Mohammad Saleh and Peter J. Liu. 2020*b*. PEGASUS: Pre-training with Extracted Gap-sentences for Abstractive Summarization. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*. Vol. 119 of *Proceedings of Machine Learning Research* PMLR pp. 11328–11339.
URL: *http://proceedings.mlr.press/v119/zhang20ae.html*

Zhang, Leilan and Qiang Zhou. 2019. Automatically Annotate TV Series Subtitles for Dialogue Corpus Construction. In *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE pp. 1029–1035.
URL: *https://ieeexplore.ieee.org/abstract/document/9023129/*

Zhang, Shiyue, Asli Celikyilmaz, Jianfeng Gao and Mohit Bansal. 2021. EmailSum: Abstractive Email Thread Summarization. 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)*. Online: Association for Computational Linguistics pp. 6895–6909.
URL: *https://aclanthology.org/2021.acl-long.537*

Zhang, Tianyi, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown and Tatsunori B Hashimoto. 2023. "Benchmarking large language models for news summarization." *ArXiv preprint* abs/2301.13848.
URL: *https://arxiv.org/abs/2301.13848*

Zhang, Xingxing, Furu Wei and Ming Zhou. 2019. HIBERT: Document Level Pre-training of Hierarchical Bidirectional Transformers for Document Summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics pp. 5059–5069.
URL: *https://aclanthology.org/P19-1499*

Zhang, Yang, Yunqing Xia, Yi Liu and Wenmin Wang. 2015. Clustering Sentences with Density Peaks for Multi-document Summarization. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Denver, Colorado: Association for Computational Linguistics pp. 1262–1267.
URL: *https://aclanthology.org/N15-1136*

Zhang, Yusen, Ansong Ni, Tao Yu, Rui Zhang, Chenguang Zhu, Budhaditya Deb, Asli Celikyilmaz, Ahmed Hassan Awadallah and Dragomir Radev. 2021. An Exploratory Study on Long Dialogue Summarization: What Works and What's Next. In *Findings of the Association for Computational Linguistics: EMNLP 2021*. Punta Cana, Dominican Republic: Association for Computational Linguistics pp. 4426–4433.
URL: *https://aclanthology.org/2021.findings-emnlp.377*

Zhang, Zhuosheng and Hai Zhao. 2021. Structural Pre-training for Dialogue Comprehension. 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)*. Online: Association for Computational Linguistics pp. 5134–5145.
URL: *https://aclanthology.org/2021.acl-long.399*

Zhao, Chao, Faeze Brahman, Kaiqiang Song, Wenlin Yao, Dian Yu and Snigdha Chaturvedi. 2022. NarraSum: A Large-Scale Dataset for Abstractive Narrative Summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*. Abu Dhabi, United Arab Emirates: Association for Computational Linguistics pp. 182–197.
URL: *https://aclanthology.org/2022.findings-emnlp.14*

Zhao, Chao, Tenghao Huang, Somnath Basu Roy Chowdhury, Muthu Kumar Chandrasekaran, Kathleen McKeown and Snigdha Chaturvedi. 2022. Read Top News First: A Document Reordering Approach for Multi-Document News Summarization. In *Findings of the Association for Computational Linguistics: ACL 2022*. Dublin, Ireland: Association for Computational Linguistics pp. 613–621.
URL: *https://aclanthology.org/2022.findings-acl.51*

Zhao, Chao, Wenlin Yao, Dian Yu, Kaiqiang Song, Dong Yu and Jianshu Chen. 2022. Learning-by-Narrating: Narrative Pre-Training for Zero-Shot Dialogue Comprehension. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Dublin, Ireland: Association for Computational Linguistics pp. 212–218.
URL: *https://aclanthology.org/2022.acl-short.23*

Zheng, Hao and Mirella Lapata. 2019. Sentence Centrality Revisited for Unsupervised Summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics pp. 6236–6247.
URL: *https://aclanthology.org/P19-1628*

Zheng, Xin, Aixin Sun, Jing Li and Karthik Muthuswamy. 2019. Subtopic-driven Multi-Document Summarization. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Hong Kong, China: Association for Computational Linguistics pp. 3153–3162.
URL: *https://aclanthology.org/D19-1311*

Zhong, Ming, 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*. Online: Association for Computational Linguistics pp. 5905–5921.
URL: *https://aclanthology.org/2021.naacl-main.472*

Zhong, Ming, Pengfei Liu, Danqing Wang, Xipeng Qiu and Xuanjing Huang. 2019. Searching for Effective Neural Extractive Summarization: What Works and What's Next. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Florence, Italy: Association for Computational Linguistics pp. 1049–1058.
URL: *https://aclanthology.org/P19-1100*

Zhong, Ming, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu and Xuanjing Huang. 2020. Extractive Summarization as Text Matching. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics pp. 6197–6208.
URL: *https://aclanthology.org/2020.acl-main.552*

Zhou, Liang, Miruna Ticrea and Eduard Hovy. 2004. Multi-Document Biography Summarization. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*. Barcelona, Spain: Association for Computational Linguistics pp. 434–441.
URL: *https://aclanthology.org/W04-3256*

Zhou, Qingyu, Nan Yang, Furu Wei, Shaohan Huang, Ming Zhou and Tiejun Zhao. 2018. Neural Document Summarization by Jointly Learning to Score and Select Sentences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Melbourne, Australia: Association for Computational Linguistics pp. 654–663.
URL: *https://aclanthology.org/P18-1061*

Zhu, Chenguang, Ziyi Yang, Robert Gmyr, Michael Zeng and Xuedong Huang. 2021. Leveraging lead bias for zero-shot abstractive news summarization. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*. pp. 1462–1471.

Zou, Yanyan, Xingxing Zhang, Wei Lu, Furu Wei and Ming Zhou. 2020. Pre-training for Abstractive Document Summarization by Reinstating Source Text. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Online: Association for Computational Linguistics pp. 3646–3660.
URL: *https://aclanthology.org/2020.emnlp-main.297*