Comparative Summarization of Document Collections

Umanga Bista

A thesis submitted for the degree of
Doctor of Philosophy at
The Australian National University

February 2022

Except where otherwise indicated, this thesis is my own original work.

Umanga Bista
28 February 2022

To my parents Mr. Resham Raj Bista, Mrs. Manju Devi Regmi Bista, my sister Ms. Shreya Bista and my wife Mrs. Suchana Panta.

Primary Supervisor and Chair of Advisory Panel

Lexing Xie
Professor of Computer Science, The Australian National University
Canberra, ACT, Australia

Advisor

Aditya Krishna Menon
Sr. Research Scientist, Google Research
New York City, NY, United States

Advisor

Alexander Mathews
Research Fellow, The Australian National University
Canberra, ACT, Australia

Advisor

Lizhen Qu
Lecturer, Monash University
Melbourne, Victoria, Australia

Acknowledgments

First, I would like to express my sincere gratitude to my primary supervisor **Prof. Lexing Xie** for all her guidance, encouragement and support throughout this doctoral journey. Her immense knowledge, inspiration, patience, and inimitable awareness has motivated me make this thesis a reality. I am grateful to Lexing for being an excellent and caring supervisor, and shaping me as a researcher. I couldn't have imagined having a better supervisor for my Ph.D.

I would like to thank my co-supervisor **Dr. Aditya Krishna Menon** for all guidance, support, and contributions. I first thank him for his immense help with writing the papers and this thesis, and enduring the grammatical and syntactic mistakes. His breadth of knowledge in machine learning has provided crucial guidance to the design choices and understanding of the methodology. I've learned a lot for him.

Finally, I thank my co-supervisor **Dr. Alexander Mathews** for his support, insights and contributions. He is forever calm, clear-headed and grounded, and has provided valuable insights that connect machine learning methods to the (messy) problem and data at hand. I also acknowledge his exceptional support as primary supervisor when Lexing was on leave.

I am extremely thankful to the Australian National University (ANU), Research School of Computer Science (RSCS), and Computational Media Lab (CMLab) for allowing me to be the part of the incredible research cohort. Specially, I would like to thank my colleague Minjeong Shin for all her suggestions, feedback and contributions. I thank her for closely collaborating with me on both building the data intensive systems and in designing crowd-sourced evalualtions used in this thesis. I would also like to thank Suvash Sedhain, Swapnil Mishra, Dawei Chen, Alasdair Tran, Siqi Wu, Quyu Kong, Rui Zhang, Qiongkai Xu, Aditya Mogadala, Marian-Andrei Rizoiu and Dongwoo Kim for their valuable suggestions and discussions throughout the journey.

I acknowledge the financial and technical support provided by the Australian National University (ANU) and Data to Decisions CRC for my doctoral program. This work is also supported by the ARC Discovery Project DP180101985, and by the use of the NeCTAR Research Cloud, a collaborative Australian research platform supported by the National Collaborative Research Infrastructure Strategy.

I would like to thank my relatives who have supported me in various phases especially Suman Raj Bista, Prashuma Khakural, Rajesh Subedi, Pramila Bista, Dhirendra Nidhi Joshi and Prashant Pandey. I would also like to thank my friends especially Dr. Kshitij Thapa,

Suprabha Adhikari, Pawan Parajuli, Prajdnik Awasthi, Sarvagya Pant, Prashant Kumar Thakur, Pushpa Sharma, Shivashankar Subramanian, and Suryansh Kumar for supporting me in various phases of this journey.

I would like to thank my in-laws Mr. Suresh Pant, Mrs Indira Panta, Ms Shamikshya Panta and Mr. Sarthak Panta or showing their love, support, and faith in me. Finally, I would like to thank my parents Mr. Resham Raj Bista, Mrs. Manju Devi Regmi Bista, my sister Ms. Shreya Bista and my wife Mrs. Suchana Panta for their unconditional love, support and encouragement throughout my PhD. I dedicate this thesis to them.

Abstract

Comparing documents is an important task that help us in understanding the differences between documents. Example of document comparisons include comparing laws on same related subject matter in different jurisdictions, comparing the specifications of similar product from different manufacturers. One can see that the need for comparison does not stop at individual documents, and extends to large collections of documents. For example comparing the writing styles of an author early vs late in their life, identifying linguistic and lexical patterns of different political ideologies, or discover commonalities of political arguments in disparate events. Comparing large document collections calls for automated algorithms to do so.

Every day a huge volume of documents are produced in social and news media. There has been a lot of research in summarizing individual document such as a news article, or document collections such as a collection of news articles on a related topic or event. Comparatively summarizing different document collections, or *comparative* summarization is a way of comparing document collections. It is under-explored problem in terms of methodology, datasets, evaluations and applicability in different domains. To address this, in this thesis, we make three types of contributions to comparative summarization, methodology, datasets and evaluation, and empirical measurements on a range of settings where comparative summarization can be applied.

We propose a new formulation of the problem of comparative summarization as *competing binary classifiers*. This formulation help us to develop new unsupervised and supervised methods for comparative summarization. Our methods are based on Maximum Mean Discrepancy (MMD), a metric that measures the distance between two sets of datapoints (or documents). The unsupervised methods incorporate information coverage, information diversity and discriminativeness of the prototypes based on global-model of sentence-sentence similarity, and be optimized with greedy and gradient methods. We show the efficacy of the approach in summarizing a long running news topic over time. Our supervised method improves the unsupervised methods, and can learn the importance of prototypes based on surface features (e.g., position, length, presence of cue words) and combine different text feature representations. Our supervised method meets or exceeds the state-of-the-art performance in benchmark datasets.

We design new scalable automatic and crowd-sourced extrinsic evaluations of comparative summaries when human written ground truth summaries are not available. To evaluate our

methods, we develop two new datasets on controversial news topics – ControvNews2017 and News2019+Bias datasets which we use in different experiments. We use ControvNews2017 , which consists of news articles on controversial topics to evaluate our unsupervised methods in summarizing over time. We use News2019+Bias , which again consists of news articles on controversial news topics, along with media bias labels to empirically study the applicability of methods.

Finally, we measure the *distinguishability* and *summarizability* of document collections to quantify the applicability of our methods in different domains. We measure these metrics in a newly curated News2019+Bias dataset (§3.4.1) in comparing articles *over time*, and *across ideological leanings* of media outlets. First, we observe that the summarizability is proportional to the distinguishability, and identify the groups of articles that are less or more distinguishable. Second, better distinguishability and summarizability is amenable to the choice of document representations according to the comparisons we make, either *over time*, or *across ideological leanings* of media outlets. We also apply the comparative summarization method to the task of comparing stances in the social media domain.

In sum, we address the problem of comparative summarization by developing new supervised, and unsupervised methods, scalable automatic and human extrinsic evaluation of comparative summaries, and studying the applicability of our methods in different domains. We expect our methods and evaluations to be applicable to other datasets and domains in building systems to compare document collections.

Publications, Software and Data

Some of the thesis has been published in conference proceedings. We also provide software and data developed during the thesis for future work.

Publications

Bista, U., 2019. Comparative summarisation of rich media collections. WSDM '19 Doc. Sym., 812–813. doi: 10.1145/3289600.3291603. URL http://doi.acm.org/10.1145/3289600.3291603

Bista, U.; Mathews, A. P.; Shin, M.; Menon, A. K.; and Xie, L., 2019. Comparative document summarisation via classification. AAAI 2019. AAAI Press. doi: 10.1609/aaai.v33i01.330120. URL https://doi.org/10.1609/aaai.v33i01.330120

Bista, U.; Mathews, A.; Menon, A.; and Xie, L., 2020. SupMMD: A sentence importance model for extractive summarization using maximum mean discrepancy. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 4108–4122. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.367. URL https://www.aclweb.org/anthology/2020.findings-emnlp.367

Software and Data

1. Comparative Document Summarization via Classification.
   - software & ControvNews2017 : https://github.com/computationalmedia/compsumm

2. SupMMD: A Sentence Importance Model for Extractive Summarization using Maximum Mean Discrepancy.
   - software: https://github.com/computationalmedia/supmmd

3. Applicability of Comparative summarization.
   - software & News2019+Bias dataset: https://github.com/bistaumanga/compsumm-applicability
   - https://www.bistaumanga.com.np/files/media_bias_factuality_oct2020_mbfc.csv (media-bias annotations)

Contents

List of Figures

List of Tables

Chapter 1

Introduction

> *Faced with information overload, we have no alternative but pattern-recognition.*
>
> Marshall McLuhan, *Counterblast*, 1969

Summarization is an important cognitive process that aids in reading comprehension [Dole et al., 1991]. In brief, it is a process by which we synthesize important ideas from long text documents to create a new coherent condensed representation of the original text. Summaries are commonplace in our everyday life, and appear as abstracts in books and research articles, key points in news articles, minutes in meetings etc. Roles of summaries is to provide a glimpse or synopsis of longer documents, or reduce information overload in later revisits.

With the advent of the World Wide Web, the amount of information published on news outlets and social media is beyond an individual's ability to process. New York Times alone produces more than 150 articles per day [Meyer, 2016] and more than 500 million tweets are produced every day [Krikorian, 2013]. While most of the information produced might not be relevant to every individual, it is very important for individuals to remain informed to make better decisions about their lives and society [Kovach and Rosenstiel, 2014]. This is the problem of information overload and automatic summarization mimics the human ability of summarization using computers and helps us tackle it [Salton et al., 1997]. Other applications of automatic summarization include question answering [Demner-Fushman and Lin, 2006], improving information retrieval performances [Sakai and SparckJones, 2001], comparing related documents, such as patents [Zhang et al., 2015], and condensing news articles to form TL;DR [autotl;dr, 2021].

Automatic summarization systems are classified based on the input to the automatic summarization system, how it forms the output summary, methodology used, and application focus. Our work is on multi-document extractive comparative summarization which we simply call *comparative summarization* throughout the thesis. Comparative summarization is a type of automatic summarization which facilitates the comparison of different groups of documents

in addition to reducing information overload. The term *multi-document* means that the input to the system is multiple documents, and *extractive* means that the summary is formed by picking a few representative prototype documents (or sentences) from the input documents. The alternative input to the summarization system can be a single document, e.g. summarizing a news article, a research paper, etc. Summaries may be presented as extracts as mentioned above, or abstract in case of abstractive summarization. An abstract is a summary which is formed by generating a new text covering salient information of the input document(s). From a methodology perspective, a system may be able to do summarization with or without learning from ground truth summaries, i.e., using supervised or unsupervised learning methods. We review different summarization types in detail in §2.1.1.

Given a collection of documents with predefined groups such as publication month, publication geography, or political ideology of the publication media, we may wish to compare different groups of documents to understand the differences. For example, we may wish to compare the articles on *Gun control* from right and left leaning media outlets to understand diverse perspectives on an important topic. Comparative summarization, where the goal is to select a small subset of representative documents or sentences (prototypes) from each group that maximally distinguish from other groups help us tackle such scenarios.

Comparative summarization is one of the interesting problems in Machine Learning that has wide range of use cases. In news media, it helps us in answering questions such as - What is new in *Beefban* topic in India this month compared to last month? What is the difference between the coverage on *Climate Change* in *left-leaning* and *right-leaning* news media outlets. In social media, it can summarize different stances on important topics such as *Gun rights*. Understanding different perspectives in news and social media on important topics help us to make informed decisions. In eCommerce, it can help us compare related products such as comparing mirrorless cameras *Canon M50* vs *Sony a6400* vs *Fujifilm X-t30* to make informed purchasing decisions. A derivative of comparative summarization, *update summarization*, where we summarize *Climate change* topic in January and *update* the summaries as new articles are available in February is useful in keeping informed about long-running and evolving news topics.

Comparative summarization is one way of comparing different document groups. There are other ways to compare different document groups, such as comparative topic modeling and visual comparisons, which facilitates comparisons by providing a topic (list of salient words to a document group), or visualizations such as word clouds of different groups. In this work, we work on comparative summarization, which facilitates comparisons by picking representative and discriminative sentences, or documents of each document groups, which we call prototypes. In particular, we focus on comparative summarization and its derivative update summarization in extractive and multi-document settings. We start by introducing the problem definitions.

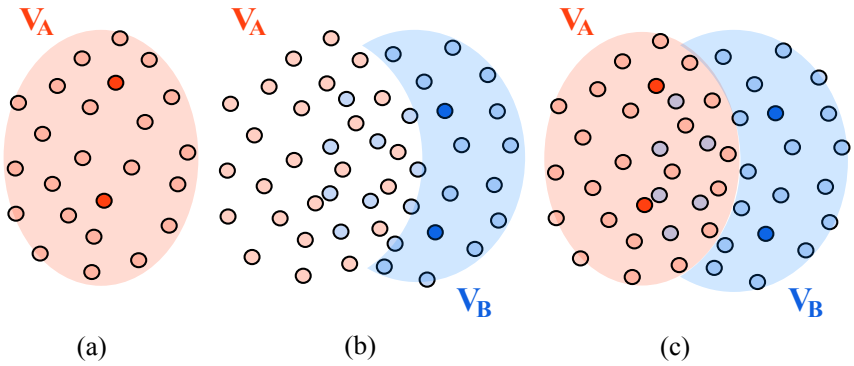


Figure 1.1: Given two sets of documents (set V_A and V_B) denoted by red and blue circles, respectively, three multi-document extractive summarization tasks are (a) Generic (b) Comparative (c) Update. Summary documents picked by summarization systems are illustrated as bold circles, and the information coverage of each task is represented by the shaded regions, the color of which corresponds to the group of documents it summarizes.

1.1 Group focused extractive multi-document summarization

Suppose we are given two groups of documents (denoted set V_A and set V_B, and illustrated as red and blue circles in Figure 1.1) on a related topic (e.g., climate change, the COVID-19 pandemic), separated by publication month or ideological bias of media outlets (e.g. left and right leaning). In the presence of such pre-defined groups, we may then identify three possible instantiations of multi-document extractive summarization (see Figure 1.1):

(i) *Generic summarization*, where the goal is to summarize a set (V_A or V_B) individually. For example, generic summarization aims to summarize all articles on *Climate change* in each month. The prototypes are representative of the document groups (V_A or V_B).

(ii) *Comparative summarization*, where the goal is to summarize a set V_B against another set V_A (and vice versa) while highlighting the differences. For example, comparative summarization identifies the key discriminative viewpoints of left and right leaning media outlets in a given topic such as *Climate change*, or compare events that happened around *Climate change* topic in January vs February (and vice versa). The prototypes are representative of V_B and discriminative against V_A (and vice versa).

(iii) *Update summarization*, where the goal is to present a user with summary of new information over time as new events develop [Dang and Owczarzak, 2008]. We note that update summarization consists of both generic summarization of set V_A *and* comparative summarization of set V_B against V_A. For example, update summarization would present a user with generic summaries on *Climate change* in January, and update the summaries with new information when new batch of articles are produced in February, and so on.

We illustrate the three summarization settings in Figure 1.1. The two groups of documents in some feature space are denoted by blue and red circles. The coverage of each summary

is denoted by the shaded regions, and prototype documents are denoted by bold circles. We call these three types of summarization group-focused summarization. They are also called multi-document and extractive because the input to the systems are multiple documents (or sentences from the documents), and we summarize by extracting a few representative documents (or sentences) which we call prototypes [1].

We note that these three instantiations of multi-document summarization is also applicable to abstractive summarization, which we briefly review in §2.1.3, but it not in the scope of this thesis. A closely related concept to update summarization is timeline summarization or time aware summarization, which we review briefly in §2.1.6. There are other types of summarization which we introduce in §2.1.1.

We note that update summarization is a derivative of comparative summarization in the sense that it is a hybrid of generic (V_A) and comparative summarization (V_B). Which means that any system we develop for comparative summarization can be immediately applied to update summarization provided that we can also perform generic summarization of V_A. In this thesis, we address both comparative and update summarization.

1.2 Research questions

While the problem of generic summarization has received much attention as we review in Chapter 2, comparative summarization is still an underexplored topic. We identify several research problems in comparative summarization topic that warrants work. Summarization by selecting representative sentences is common in extractive summarization literature. We put forward an equally interesting problem of selecting documents in comparative summarization. It would be interesting to pick entire documents in datasets with contrastive groups, such as a corpus of tweets, or news articles on some controversial topic. A few existing approaches to do comparative summarization work by extracting discriminative sentences [Li et al., 2012a; Wang et al., 2012] from a small corpus in the absence of ground truth summaries. Instead, we would like to derive a method that is applicable to a range of scenarios such as comparing document collections by picking sentences and documents, and can be potentially applied to domains other than text.

First, a system may do automatic summarization with or without learning from the ground truth summaries. We would also like our method to do comparative summarization in the absence of ground truth summaries. But, when the ground truth summaries are available, we would like our method to learn useful signals from them. These are the unsupervised

[1] Alternatively, we could also call them 'exemplar', as these two terms are often used to describe our ability to categorize objects in cognitive psychology [Ross and Makin, 1999; Storms et al., 2000]. 'Prototype' is more abstract compared to 'exemplar' which is memory based, and is commonly used to describe a representative sample of a category in exemplar based classification and data-summarization literatures [Kim et al., 2016; Bien and Tibshirani, 2011; Dornaika and Aldine, 2015; Kim et al., 2014].

and supervised learning problems in comparative summarization. We first we seek to find a method of comparative summarization in absence of ground truth summaries, and applicable to all the scenario explained in the previous paragraph. An unsupervised method would allow us to apply comparative summarization in any appropriate large scale datasets. Hence, the first research question is: **RQ1. Can we do unsupervised comparative summarization?**

In automatic summarization literature, most of the unsupervised methods make use of global model of sentence-sentence similarity to find representative sentences, deemed important, as the summaries [Erkan and Radev, 2004; Lin and Bilmes, 2011]. However, in the presence of ground truth summaries, they provide useful extra signals to decide the importance of sentences. From a machine learning modeling perspective, these signals can be learned with supervision. In generic summarization, supervision from ground truth summaries has proven to be useful in improving the qualities of the summaries. A supervised method would help us to effectively apply comparative summarization in domains where ground truth summaries are available. Hence, we seek to develop a supervised learning technique for comparative summarization (and update summarization) as second research question. **RQ2. Can we do supervised comparative summarization tasks?**

Evaluating comparative summarization systems is a long running challenge. In generic summarization standardized datasets are ubiquitous [Over, 2003; Over and Yen, 2004; Over et al., 2007; Nallapati et al., 2016b] and evaluation techniques are well documented [Lin, 2004; Nenkova et al., 2007; Hovy et al., 2006]. In contrast, evaluations of comparative summarization systems typically rely on small datasets, with a mix of ad-hoc qualitative and quantitative measures [Huang et al., 2011]. In update summarization, there is a small scale benchmark dataset available for the supervised settings [Dang and Owczarzak, 2008, 2009]. However, there is a lack of a large scale standard datasets, and evaluations protocols for unsupervised settings, i.e. when human written ground truth summaries are not available. Nevertheless, it would be useful if we could cheaply evaluate the quality of comparative summaries without ground truth summaries. Hence, we seek to answer: **RQ3. How can we evaluate comparative summaries without using ground truth summaries?**

Comparative summarization has been used to compare news collections over time [Duan and Jatowt, 2019; Huang et al., 2011], compare arguments [Rieskamp, 2022], and compare patents [Zhang et al., 2015]. The underlying assumption in these literatures is that the groups of documents are comparable (or distinguishable). No previous work has explored and quantified how accurately comparative summarization can be applied to datasets where different facets along which a document collections can be split, such as time, source/authorship, and ideological leaning. Hence, we seek to answer is: **RQ4. Can we quantify the applicability of comparative summarization to document collections with multiple types of groups?**

1.3 Thesis contributions

The core of the thesis consists of developing methodologies, evaluations, and studying the applicability of comparative summarization. To address the research questions raised in §1.2, we make several following contributions.

First of all, it is well known that datasets and benchmark datasets are the precursor of many great things happening to a research problem [Deng et al., 2009; Rajpurkar et al., 2016]. We hence develop two new datasets based on *controversial news topics* in Chapter 3 – ControvNews2017 and News2019+Bias datasets, which provide the foundation for answer RQ1, RQ3 and RQ4. Controversial news topics such as *climate change, beef ban, gun control, etc.* are interesting. Because they evolve over time allowing us to do comparisons over time, and they attract different viewpoints i.e., content produced by outlets with different ideological biases in news media (e.g. left and right leaning) can be compared.

1. Problem formulation and unsupervised methods: We formulate the problem of summarization as binary classification and comparative summarization as competing binary classification. This allows us to define new methodologies (both supervised and unsupervised) and evaluation for comparative summarization. In particular, in Chapter 4 within this classifier formulation we propose unsupervised objectives, addressing the RQ1. Our objectives are on MMD (maximum mean discrepancy): a kernel based distance measure between two sets of data points [Gretton et al., 2012a], and can be optimized by discrete or continuous optimization strategies. We show the efficacy of these methods in comparatively summarizing controversial news topics over time on ControvNews2017 dataset.

2. Supervised method: We extend our unsupervised MMD based summarization (generic and comparative) to the supervised setting, addressing the RQ2. In particular, we propose SupMMD with two stages of supervision. First, the method learns to combine different textual features using Multiple kernel learning [Cortes et al., 2010]. Second, we introduce weighted MMD (wMMD), which learns sentence saliency using ground truth summaries. The proposed method consists of a single model used in both learning and inference. It is based on a log-linear model with only a few trainable parameters and, can incorporate a diverse range of textual features. We show the effectiveness of the method in benchmark generic and update summarization tasks.

3. Unsupervised evaluation protocols: We propose a scalable extrinsic evaluation for comparative summarization using classification performance. This proposal takes advantage of our formulation of comparative summarization as competing binary classifiers. It is applicable to automatic and human based evaluations which we demonstrate. The evaluation is

also scalable - i.e., we can scale the evaluations to larger datasets in unsupervised settings, i.e. when ground truth human written summaries are not available. Hence, it addresses RQ3, and also lays the foundation for the measuring applicability to address RQ4.

4. Applicability of comparative summarization: To address RQ4, we study the application of comparative summarization in a range of tasks on social and news media with the goal of quantifying to what degree we can apply our methods. We introduce two new metrics – *distinguishibility* and *summarizability*, in order to quantify along which facets document collections can be *comparatively summarized*. In news media, we compare across ideological leanings of media outlets and over publication month. We observe that the summarizability is proportional and close to the distinguishability, meaning summaries are useful in comparing document collections. We also observe that the distinguishability is amenable to feature representations based on the type of comparisons we are doing – over months or across the ideological leanings. Finally, we compare stances in social media short text, where summarizability suffers due to sparsity of data, even though the summaries make sense in qualitative analysis.

1.4 Thesis outline

Having introduced the problem of comparative summarization, this dissertation now examines the problem of comparative summarization in detail. The thesis is structured as follows.

Chapter 2: We first review the literature on summarization, different datasets and evaluation strategies for summarization, and comparing document collections. We then provide a brief overview of kernels, maximum mean discrepancy and submodular functions, which form the mathematical foundations of the thesis.

Chapter 3: We describe the two news datasets on controversial news topics we curated. We first discuss our data collection methodology, including a scalable and fault-tolerant data collection and pre-processing architecture. We then describe two datasets, which we use for experiments in Chapter 4 and Chapter 6.

Chapter 4: We describe a novel formulation of comparison summarization as competing binary classifiers, as well as unsupervised methods for comparative summarization, and their optimization strategies. We also describe new extrinsic scalable human and automatic evaluation protocols for evaluating comparative summaries in the absence of ground truth summaries. We apply comparative summarization to comparing controversial news topics over time.

Chapter 5: We describe a supervised method for generic and comparative summarization. We introduce two kinds of supervision – learning sentence importance and combining different text features. The models meet or exceed state-of-the-art performance in benchmark datasets.

Chapter 6: We describe two new metrics – *distinguishibility* and *summarizability* that help to quantify the applicability of comparative summarization in a range of tasks in the web and social media domain. We present the observations of our large-scale measurement study, with focus on new media-bias dataset where we compare over time and across ideological leanings.

Chapter 7: We summarize the work presented in this thesis, and discuss the limitations and future directions.

This thesis contains 18 figures, 30 tables, and 267 references.

Chapter 2

Literature Review and Background

“ Study without desire spoils the memory, and it retains nothing that it takes in. ”

Leonardo da Vinci,

Our work in this thesis is primarily on comparative summarization, a way of comparing document collections using automatic summarization. In the first part of this chapter, we review the existing works in document summarization, comparing document collections, different datasets and summarization evaluation techniques. We first provide an overview of different approaches towards automatic summarization and its types, including group-focused comparative and update summarization in §2.1. Second, in §2.2 we formally define the problem of extractive summarization in presence of groups, comparative and update summarization as introduced in §1.1, which is the focus of this thesis. We then review different existing datasets (§2.3) and evaluation methods (§2.4) and identify the ones that fit in our problem settings. Finally, in §2.5 we review alternative approaches to comparing document collections other than comparative summarization.

In the later part of this chapter, we introduce the background theory that underpins the methodologies developed in this thesis. First, we introduce kernels, RKHS (reproducing kernel Hilbert spaces), and MMD and see MMD as a summarization method in §2.6. We use MMD to develop unsupervised comparative summarization methods in Chapter 4 and supervised method in 5 for comparative summarization to address research question 1 (RQ1) and research question 2 (RQ2) of the thesis. Finally, we introduce a special class of set functions called as *submodular-monotone* in §2.7, whose properties allow existing and our methods to be efficiently optimized.

2.1 Approaches to automatic summarization

Throughout this section, we review approaches towards automatic summarization, which is the broader context of our work in comparative summarization. We first introduce different types of summarization. Second, we review the evolution of summarization methodologies. Third, we briefly discuss some methodologies on extractive-vs-abstractive, single vs multi-document, and different application focuses. We will briefly discuss different group-focused summarization techniques in §2.1.6, as it is the focus of this thesis.

2.1.1 Summarization types

We can categorize summarization methods bases on input, methodology, output and application focuses, which we illustrate in Table 2.1. The categorizations can be applied independently, i.e., any summarization method can lie in the intersection of these different categorizations, e.g., Yasunaga et al. [2017] is multi-document extractive supervised generic summarization, Nallapati et al. [2016b] is single-document supervised generic abstractive summarization.

Input	Output	Method	Application focuses
Single doc (SDS)	Abstractive	Unsupervised	Query focused, viewpoint, etc.
Multi-doc (MDS)	Extractive	Supervised	Group-focused (comparative, update)

Table 2.1: Four dichotomies of document summarization problems. The categorizations by different criteria are not mutually exclusive. Our thesis focus is highlighted with gray background.

Summarization input can be a single document, e.g. a single news article, in which case it is known as Single document summarization (SDS), or multiple documents, e.g. a set of news articles on a related event, in which case it is known as Multi-document summarization (MDS). Based on the output, summarization can be extractive or abstractive. In extractive summarization, only the parts of the input documents, such as sentences are extracted and combined to create a summary. Whereas in abstractive summarization, a new text is created by rephrasing the salient parts of the input documents. The former is the more popular approach in automatic summarization, as it is simpler and tends to cause fewer syntactic and semantic mistakes than abstractive summarization [Nallapati et al., 2017]. Also, the latter suffers from the difficulties in natural language generation whereas the former follows a data driven approaches [Erkan and Radev, 2004]. From the learning perspective, summarization algorithm can be supervised or unsupervised. In supervised learning setting, we train a machine learning model to summarize documents by providing ground truth, which are often human written summaries. Whereas, in unsupervised learning setting, the model produces a summary without any supervision, often using the global model of text similarity.

From application focus perspective, summaries can be generic or focused. Generic sum-

marization produces a summary for a given document (or set of documents), without any further conditions. Alternatively, we might want the automatic summarization system to generate summaries focused on some criteria, such as user queries, or specific aspect of the documents such as causes of bushfires in a document collection about bushfires. Our work is on group focused summarization, i.e., in the presence of different groups of documents, which can be comparative and update summarization as we introduced in §1.1. Next we review how summarization methods has evolved over last six decades.

2.1.2 A brief history of approaches towards automatic summarization

Automatic summarization has been in existence since 1950s. Different approaches have been developed in the last seven decades of research from the Natural Language processing (NLP), and Information Retrieval (IR) communities. Roughly, we can classify the development into four eras – 1) using corpus statistics, 2) using linguistic and retrieval approaches, 3) using supervised and unsupervised learning, and 4) using deep learning. We now briefly discuss some approaches that are representative of each of these eras.

The first attempt to automatic summarization used corpus statistics such as word frequency distributions to model the sentence importance and extract important sentences as summaries [Luhn, 1958]. Later, Edmundson [1969] used additional indicators such as cue words, title and heading words, and sentence locations to further improve performance. These earlier methods were based on statistical features extracted from the corpus, and the summaries were generated by extracting the important sentences and combining them.

Automatic summarization became more popular amongst the NLP and IR communities when researchers started to apply more advanced computational linguistic based approaches which provide more information than mere statistical features in generating the summaries. McKeown and Radev [1995] used human curated templates to generate summaries of news articles on the events on certain domains. They made use of several linguistic features such as syntactic form of the text and, built lexicons that link information pieces and forms important phrases. Mani and Bloedorn [1997] used synonym resolution to relate sentences and phrases, and used graph matching algorithms to identify key textual units which can be synthesized to form summaries. Carbonell and Goldstein [1998] used Maximal Marginal Relevance (MMR), an information retrieval technique to extract summary sentences from multiple documents that are relevant to a user query and diverse. Goldstein et al. [1999] use weighted combination of statistical features such as TF-IDF scores, cosine similarity, etc. and linguistic features such as name of person and places, presence of proper nouns, thematic phrases, etc. to extract important sentences. Linguistic features based methods generally identify the key linguistic components by analyzing the ground-truth summaries [McKeown and Radev, 1995; Goldstein et al., 1999].

More recent summarization methods are based on unsupervised and supervised machine learning. Unsupervised methods are useful in extracting summaries from corpus which does not have human written ground-truth summaries. Radev et al. [2004] used text similarity from corpus statistics to identify sentences that are central to the topic of the documents. Erkan and Radev [2004] represent the corpus as a sentence-sentence similarity graph and use graph centrality measures such as eigen-vector centrality to score the sentence importance. Sentence scores are used to select sentences iteratively, but sentences that are redundant to previously selected ones are avoided [Erkan and Radev, 2004]. Another group of works formulates summarization as utility maximization of summary sentences, where utility measures the usefulness of summaries. Utilities such as graph cuts [Lin and Bilmes, 2010], combination of information coverage and diversity [Lin and Bilmes, 2011], and concepts coverage [Gillick and Favre, 2009] has been explored. Some of these utilities belong to a special class of set functions called *submodular-monotone* [Lin and Bilmes, 2010, 2011], which can be efficiently optimized (more details in §2.7), whereas some methods use exact optimization with integer linear programming [Gillick and Favre, 2009]. Finally, topic models can be used to model the topic vocabulary distributions of the corpus, and extract summary sentences whose vocabulary distribution best matches the corpus distributions [Haghighi and Vanderwende, 2009].

In presence of ground truth summaries available for documents, supervised methods can produce better summaries. Earlier approaches to supervised summarization includes learning to classify if a sentence belongs to a summary, with a naive Bayes classifier [Kupiec et al., 1999], or decision trees [Lin, 1999]. The learned classifier is used to rank sentences and the summary is formed by extracting the top scoring sentences. A disadvantage of such approaches is the assumption that the decision as to whether a sentence forms a summary is independent of that decision for a previous sentence. This was addressed by Conroy and O'leary [2001] using hidden Markov models (HMM), and by Shen et al. [2007] using conditional random fields (CRF), which treats summary extraction as a sequence labeling problem rather than *iid* classification. Li et al. [2009a] cast summarization as a structured prediction problem, i.e., mapping from a set of sentences to a subset of sentences. An advantage of their method is incorporating summary qualities such as information diversity, coverage and balancing different aspects of given document set directly as constraints into their structured prediction framework. Combining different submodular set functions has been employed to take advantages of efficient discrete optimizations of submodular functions [Lin and Bilmes, 2012]. Hong and Nenkova [2014] trained a function to identify important words for extracting important sentences. Another framework to extract important sentences as summaries is by using determinantal point processes (DPP), where sentence importance scores can be learned simultaneously by maximizing information diversity of the summary sentences [Kulesza and Taskar, 2012; Cho et al., 2019].

Recently, some works use deep neural networks to extract important sentences, such as

using Recurrent neural networks (RNN) for sequence labeling [Nallapati et al., 2017], using Recursive neural networks for ranking sentences Cao et al. [2015], and using graph neural networks to learn sentence saliency [Yasunaga et al., 2017]. The advantage of such deep neural methods is that the models can learn more sophisticated rules between features using non-linear, hierarchical, and sequence models.

We have mostly discussed the approaches to summarize by extracting important sentences (or documents), which is known as *extractive summarization*. An alternative is to generate abstracts, which is *abstractive summarization*. Extractive summarization has been popular in the summarization literature until recently, because it does not have to deal with the difficulties in natural language generation [Nallapati et al., 2017]. With the success of language generation by sequence-to-sequence models in Machine Translation, abstractive approaches have also appealed among summarization works. We now discuss some of such abstractive summarization models.

Rush et al. [2015] used sequence-to-sequence attention networks to generate headlines and short summaries, but the summaries often suffer due to out of vocabulary words (OOV), inability in modeling longer documents, being repetitive, and generating factually incorrect summaries. Nallapati et al. [2016b] used hierarchical attention with word level and sentence level attention to address the problems of modeling longer documents. Nallapati et al. [2016b]; See et al. [2017] used pointer generator networks [Vinyals et al., 2015], which balances between generating and copying existing words from the given documents, to address the OOV and factual incorrectness issues. Paulus et al. [2017] introduced the intra-temporal attention in encoder to address the repetitiveness issue, while See et al. [2017] used coverage loss in attention to address it. Recently, transformers based models [Liu and Lapata, 2019b] to generate summaries have been explored. Pilault et al. [2020] proposed a transformer based abstractive summarization model that summarizes based on key sentences extracted using a pointer network. These works on abstractive summarization are in supervised settings. On the other hand, Chu and Liu [2019] developed an end-to-end unsupervised abstractive model based on auto encoder. Liu and Lapata [2019b] leveraged the recent success in pretrained transformer based language models in extractive and abstractive summarization. While, Wang et al. [2020] showed that additional information from topic models can improve the quality of summaries in transformer architecture.

We just briefly overviewed how summarization methods has evolved amongst the NLP and IR communities. In this process, we discussed different supervised and unsupervised methods, and briefly introduced abstractive and extractive summarization. Next, we will briefly introduce different types of summarization in addition to the ones we just introduced.

2.1.3 Extractive and abstractive summarization

We just briefly introduced *extractive* and *abstractive* summarization settings (§2.1.2). We now discuss extractive summarization in detail as it is the focus of this thesis. Most of the methods developed in extractive summarization have two components – a sentence importance (or saliency) scorer and sentence selection. First, a sentence scorer assigns each sentence a saliency or importance score. This is followed by sentence selection, which is a discrete optimization problem of selecting a subset of important yet diverse subset of sentences from a larger set. The discrete optimization often takes the form of utility set maximization, as it is natural to model the qualities of summaries as a utility. These two components are often decoupled in supervised settings, i.e., a sentence scorer is independently learned, and discrete optimization objective is separately defined. Whereas in unsupervised settings, they might be jointly defined as a part of single discrete optimization objective, as the importance is implicitly modeled into it as information coverage and/or prototypes diversity.

Sentence saliency : Unsupervised sentence importance are based on distance from centroids [Radev et al., 2004], graph centrality [Erkan and Radev, 2004], word statistics Nenkova and Vanderwende [2005], importance as deemed by topic models Haghighi and Vanderwende [2009], concepts coverage [Gillick and Favre, 2009], submodular utility set functions maximizing coverage and diversity [Lin and Bilmes, 2011, 2010]. Such methods define sentence importance from global model of sentence-sentence similarity [Erkan and Radev, 2004; Lin and Bilmes, 2011, 2010], or corpus statistics [Radev et al., 2004; Nenkova and Vanderwende, 2005; Haghighi and Vanderwende, 2009]. Unsupervised models assume all sentences are created equal, but often in context of summarization some sentences are more important than others. For example, in news articles, the top few sentences often contain key information and forms summary [Kedzie et al., 2018].

In presence of ground truth summaries, we can employ supervised methods to learn importance signals from available ground truth summaries. Some supervised method learn sentence individually for each sentence, such as learning n-grams overlap of each sentence with summary sentences using support vector regressor [Varma et al., 2009], and graph neural networks [Yasunaga et al., 2017]. Some methods learn importance of sub-sentence level textual units such as words and phrases and combine them [Hong and Nenkova, 2014; Cao et al., 2015]. Alternatively, sentence importance can be learned with structured prediction [Li et al., 2009a; Lin and Bilmes, 2012], or from determinantal point-processes [Kulesza and Taskar, 2012; Cho et al., 2019].

Sentence selection : Selecting a few important and diverse sentences to form summary is a discrete optimization problem. Greedy algorithms, which select a set of sentences based

on the sentence scores while maintaining the diversity of sentences [Goldstein et al., 1999; Radev et al., 2004; Erkan and Radev, 2004; Nenkova and Vanderwende, 2005; Cao et al., 2015; Haghighi and Vanderwende, 2009; Hong and Nenkova, 2014; Kulesza and Taskar, 2012; Cho et al., 2019; Lin and Bilmes, 2012]. Most often, the choice of set utility function in sentence selection falls to a special category of set functions called as *submodular-monotone*, which allow us to use greedy algorithms with bounded approximations guarantees [Nemhauser et al., 1978], we cover more on this in §2.7.2. Some methods employ exact integer linear programming based method to select sentences [Gillick and Favre, 2009], or a sequence classifier using recurrent neural networks [Nallapati et al., 2017] or encoders [Liu and Lapata, 2019b].

2.1.4 Single and multi-document summarization

Automatic summarization can take single document or multiple documents as an input, in which case it is called as *single document summarization (SDS)*, or *multiple document summarization (MDS)* respectively. An example of SDS is to summarize a news article, or a scholarly article, whereas an example of MDS would be to summarize a news topic consisting of articles about some related events such as *Climate change, Nepal Earthquake 2015, Gun Control*, etc. over a period of time, or summarizing user reviews/opinions on a product. We avoid a detailed review of techniques for SDS and MDS since our work focus on MDS. We first note that most of the extractive methods detailed in §2.1.3 are applicable to both SDS and MDS. A key challenge in MDS compared to SDS is to additionally address redundancy of information in input documents, since multiple documents covers same event or a facet of a product. This is addressed by selecting a diverse set of sentences during sentence selection phase [Erkan and Radev, 2004; Lin and Bilmes, 2011; Kulesza and Taskar, 2012].

Most recent neural abstractive models that we briefly discussed in §2.1.2 are specifically designed for SDS in supervised setting [Rush et al., 2015; Nallapati et al., 2016b; Paulus et al., 2017; See et al., 2017]. Developments of neural methods in MDS has been limited by the availability of large scale MDS supervised training datasets, and difficulty in encoding the multiple documents. We will review the different available datasets in summarization in §2.3. Chu and Liu [2019] developed the unsupervised method for MDS on opinion summarization domain. First, to address the dataset issue for supervised MDS settings, recent works have curated a large scale MDS datasets in news domain [Fabbri et al., 2019], and from wikipedia [Liu and Lapata, 2019a]. Second, representing multiple documents is generally addressed by hierarchical transformers [Fabbri et al., 2019; Liu and Lapata, 2019a].

We have briefly discussed summarization types based on input (SDS or MDS), output (extractive or abstractive), and method (supervised and unsupervised). We next review different types of summarization by focus and in data domains other than text.

2.1.5 Summarization by focus

Summarization can be categorized as generic, or focused. Focus can be due to groups or by other application requirements. We introduced generic, and group focused comparative and update summarization in §1.1. Most of the approaches discussed till now are applicable to generic summarization. We will cover group-focused summarization in detail in §2.1.6 as they are important background for this thesis. We now briefly introduce focused summarization with other application requirements.

Based on focus, several summarization tasks have been proposed in the literature. We refer readers to Over et al. [2007] for details on each task. *Novelty track* of summarization was intended to build an automated bulletin like system for a user to track most recent information, which often arrives in bursts [Soboroff and Harman, 2003]. *Viewpoint summarization* task is to produce summaries matching a viewpoint description, which describes the important facets of the document set within a border topic using a short text [Over, 2003]. An example of viewpoint description is "Drugs and the treatment of schizophrenia" in a cluster of news articles about *Schizophrenia*. *Query-focused* where the focus is on query description, or question answering [Over, 2003; Over and Yen, 2004; Dang, 2006a, 2005, 2006b]. Question answering summarization goal is to answer *simple questions* such as "What are the advantages of growing plants in water or some substance other than soil?" in *Hydroponics* topic, or "Who is Stephen Hawking" in a cluster containing articles about him. Later, *complex* question answering tasks were also introduced, with queries such as "Why are wetlands important? Where are they threatened? What steps are being taken to preserve them? What frustrations and setbacks have there been?" on topic *wetlands value and protection* was introduced [Dang, 2006a, 2005, 2006b].

Approaches to query focused summarization include adding a term to incorporate the query information in generic summarization systems [Li et al., 2012a; Lin and Bilmes, 2011]. Often queries are very short text and methods suffer from information limit, thus approaches to incorporate query expansion with graph-based methods [Zhao et al., 2009], probabilistic graphical models [Daumé III and Marcu, 2006]. *Guided summarization* where the goal is to summarize the documents covering a pre-determined list of important aspects [Owczarzak and Dang, 2010] using a deeper linguistic analysis of the documents instead of relying on corpus statistics[1]. An example of possible aspects on *accidents* and *natural disasters* related topic is "what happened; date; location; reasons for accident/disaster; casualties; damages; rescue efforts/countermeasures". Some approaches include guided sentence compression [Li et al., 2013], aspect recognition [Zhang et al., 2011], and language generation [Genest and Lapalme, 2012].

[1] https://tac.nist.gov/2010/Summarization/Guided-Summ.2010.guidelines.html

2.1.6 Group focused summarization – comparative and update

We now review the literature on group focused summarization, specifically comparative summarization and its close derivative update summarization, as these are the main focus of this thesis. We introduced comparative and update summarization in §1.1 as forms of summarization that facilitates comparisons across document groups and over time. Comparative summarization is one way to compare document collections. There are other ways to compare document collections – *comparative topic modeling* and *visual comparisons*, which we will cover in §2.5 and compare to comparative summarization.

Comparative summarization : Comparative summarization is one of the least explored problems in the summarization literature. It facilitates comparisons by providing a summary that highlights the differences between document groups. This is basically a summarization problem with summaries facilitating additional *focus* on comparisons between document groups. Applications of comparative summarization include mining comparative facts [Huang et al., 2011], comparing news articles across time [Duan and Jatowt, 2019] and comparing patents [Zhang et al., 2015]. He et al. [2016] applied comparative topic models to identify comparative sentences among groups of scientific papers. Li et al. [2012a]; Wang et al. [2012] extract one or few discriminative sentences from a small multi-document corpus utilizing greedy optimizations and evaluating qualitatively. Huang et al. [2011] compares descriptions about similar concepts in closely related document pairs, leveraging an integer linear program and evaluating with few manually created ground truth summaries. In some cases, submodular generic summarization methods can be extended to comparative summarization [Li et al., 2012a] without compromising much in approximation qualities even though it breaks monotonicity [Lin and Bilmes, 2010].

Update summarization : Update summarization is a derivative of comparative summarization, where the goal is to update a user with a new summary as new documents are produced. For example, summarizing a news topic such as *climate change* in January and updating the summary in February is an example of update summarization. It is a hybrid case of generic summarization of topic in January and comparative summarization of topic in February against January. Update summarization has been explored relatively more compared to *comparative summarization*. DUC ran an update summarization competition in 2007[2], and TAC ran competitions in 2008 and 2009[3] [Dang and Owczarzak, 2008]. We use these datasets in evaluating our supervised methods in Chapter 5. The datasets have two groups of documents for each topic – set A, which is generic part and set B, which is comparative part.

[2]https://duc.nist.gov/duc2007/tasks.html
[3]https://tac.nist.gov/data/index.html

We now review some of the best performing methods in these datasets, especially how the comparative part is addressed. Since comparative summarization has additional focus on differentiating against set A, most of these methods have some extension to handle the comparativeness. Gillick et al. [2009] maximized concept coverage with exact integer linear programming for summarization, and upweighing the concepts in first sentences of set B by a factor of 3, and a factor of 2 for set A as concepts in the first sentences are deemed to be very important in their analysis. Varma et al. [2009] used support vector regression for predicting sentence saliency. They address comparativeness by using a feature called *novelty factor*, which measures how new a word is in set B compared to set A. Peyrard [2019] modeled summarization using KL divergence, which models coverage with cross entropy between vocabulary distribution of set B sentences and that of summary sentences, and diversity with entropy of vocabulary distribution of summary sentences. They additionally use cross entropy between vocabulary distribution of set A and that of summary sentences to model comparativeness.

Time-aware summarization is an emerging sub-problem, where the goal is to summarize the content produced online over time, and has been getting attention recently. Time aware summarization is similar to update summarization, i.e., summarizing a rapidly evolving online content to recommend personalized tweets [Ren et al., 2016], summarizing viewpoints in social-media [Ren et al., 2013], or continuously updating news summaries [Rücklé and Gurevych, 2017], building news bulletins [Schiffman and McKeown, 2005], and generating structured stories such as metro maps [Shahaf et al., 2012]. Another important aspect of timeline summarization is incorporating causality of events and future references [Rücklé and Gurevych, 2017; Yan et al., 2011]. In update summarization, we rather treat time in discrete periods, with a goal to summarize each time period for discriminative information.

Despite these works in update summarization, which is similar to comparative summarization, comparative summarization remains a little explored problem and lacks benchmark datasets and evaluations. Our thesis lies in the intersection of extractive, multi-document, and comparative summarization, which we simply call as *comparative summarization* from now on. In this thesis, we develop supervised and unsupervised methods for comparative summarization (and consequently on update summarization), unsupervised evaluation protocols for comparative summarization, and empirically study the applicability of comparative summarization in different datasets. We have seen a brief overview of extractive summarization and its types – generic and group-focused comparative and update summarization till now.

2.1.7 Summarization in other data domains

While, it is natural to think summarization of text documents; summarization also exists for other data domains such as images [Simon et al., 2007; Tschiatschek et al., 2014; Xu et al., 2011], where we select few representative images from an image collection, and multimodal

data such as combination of text and images [Li et al., 2012b; Kim et al., 2015a]. Video summarization, where goal is to select a subset of frames that captures salient information of the input video has also gained some interests recently, using convolutional sequence models [Rochan et al., 2018], long term short term memory networks [Mahasseni et al., 2017], and maximizing non-monotone sumbodular set functions [Mirzasoleiman et al., 2017]. Next, we will mathematically define the problem of extractive summarization.

2.2 Extractive multi-document summarization problem

Extractive generic multi-document summarization works by selecting a few representative documents from a set of all documents in a topic. A topic is an abstract concept describing an event e.g. news articles on *climate change*. Let $\mathcal{D} = \{V^t\}_{t=1}^{T}$ be T topics of documents. Each **topic** $V^t \in \mathcal{V}$ with $N_t = |V^t|$ is a set of N_t documents on a related event, where $\mathcal{V}$ is the input space which is non-empty, such as space of all news articles[4]. Let $\mathbf{X}^t \in \mathbb{R}^{N_t \times d}$ be a d-dimensional vector representation of V^t. We alternate between these two notations as per convenience.

Let $S^t \subset V^t$ be any subset of V^t, which has a potential to a summary of topic t. If the **ground truth summaries** are provided in the dataset, we denote it by $\hat{S}^t$ and $\hat{\mathbf{X}}^t$. Ground truth summaries are often written by human and known as **human abstracts**. Supervised extractive summarization algorithms require *extractive ground truth*, i.e, summary sentences from the dataset.

Definition 2.1 (Oracles)**.** The sentences in the datasets that best match (e.g. in terms of ROUGE metric §2.4.2) with the human abstracts are known as oracles.

Oracles are used to train supervised extractive summarization algorithms. We use ROUGE metrics (§2.4.2) to determine which sentences from dataset best matches the *human abstracts*. An algorithm to extract oracles is provided in §5.4.4.

Summarization as prototype selection : The problem of extractive summarization is to select summary documents (or sentences) $\bar{S}^t \subset V^t$. In case of vector representation, the summarization problem is to select summary vectors $\bar{\mathbf{X}}^t \subset \mathbf{X}^t$. The summary documents, sentences or vectors are also known as **prototypes** and contain salient information of the document collection to be summarized. Typically, not all subset of V^t are valid summaries; often we have constraints such as *cardinality constraint* on number of prototypes[5], i.e. $|\bar{S}^t| \leq M$

[4]We might alternatively refer to a topic as a set consisting of all sentences from all documents based on what granularity we want the summarization system to output, i.e. summarize by selecting documents (Chapter 4, 6) or sentences (Chapter 5).

[5]We assume M (number of prototypes) to be same across all topics and groups.

or *budget constraint* on summaries, i.e. $\sum_{s \in \bar{S}^t} \text{len}(s) \leq B$, where B is the number of words that summary cannot exceed. Hence, we restrict $\bar{S}^t \in \mathcal{S}^t$, where $\mathcal{S}^t \subseteq 2^{|V^t|}$ satisfies the either *budget* or *cardinality* constraint requirement.

Most extractive summarization approaches can be performed in a two-stage procedure as discussed in §2.1.3. First, scoring and/or ranking sentences (or documents), and secondly, selecting sentences (or documents) to be combined and presented as a summary. Formally, let $f : \mathcal{V} \mapsto \mathbb{R}$ be a function that scores each sentence (or document) based on unsupervised or supervised model of importance. With supervision, the function f may be parameterized by learnable parameters θ. Let $\mathcal{U} : \mathcal{S} \mapsto \mathbb{R}$ be a utility function, typically a non-negative function, that assigns utility value to each subset of a topic (V^t) to be summarized. Then, we formally write extractive summarization as:

$$\bar{S}^t = \underset{S^t \in \mathcal{S}^t}{\text{Argmax}}\, \mathcal{U}(S^t; V^t, f) \tag{2.1}$$

We note that this is a discrete optimization problem, which is often NP-Hard [Krause and Golovin, 2014]. We will discuss in §2.7 that for certain class of utility functions called *submodular-monotone* functions, efficient greedy algorithms are effective.

Groups within a Topic : We just provided a brief overview of *generic summarization*, where the goal is to summarize a document corpus without any specific requirement. But, the summaries can *focus* on different aspects such as a query or short description provided by a user, certain viewpoint, comparing different groups within a corpus, or updating summary when new documents are produced over time.

In this thesis, we work on the summarization system that focuses on comparative and update aspects, in presence of pre-defined **groups**, e.g. articles separated by a *publication month*, or *ideological bias of publication media*, we denote V^t as $(V_1^t, V_2^t, \ldots, V_G^t)$ and $\mathbf{X}^t$ as $(\mathbf{X}_1^t, \mathbf{X}_2^t, \ldots, \mathbf{X}_G^t)$. Then the problem of extractive summarization is to select prototypes in each *group* within the topic as $\bar{S}_g^t \subset V_g^t$. Next, we review the datasets available in summarization literature.

2.3 Summarization datasets

We provide a list of datasets that has been used in summarization literature in Table 2.2 and identify which of these datasets are applicable in our experiments. Several datasets are contributed by Document Understanding Conference (DUC[6]) and Text Analysis Conference (TAC[7]) have been widely used. These datasets are created from news articles, for both SDS and MDS, and for various applications – generic, query-focused, update, etc. The datasets

[6] https://duc.nist.gov
[7] https://tac.nist.gov

Dataset	input	domain	applications	dataset size	ground truth	output
DUC-2001 [Over, 2001]	Both	news	G	60×10	abstracts	SDS: 100 words MDS: 50, 100, 200, 400 w
DUC-2002 [Over and Liggett, 2002]	Both	news	G	59×10	abstracts/ extracts	SDS: 100 w MDS: 10, 50, 100, 200, 400 w
DUC-2003 [Over, 2003]	Both	news	G,V,QF-S,N	60×10, 30×25	abstracts	SDS: 10 w, MDS: 100 w
DUC-2004 [Over and Yen, 2004]	Both	news	G, QF-S	{50,50}×10	abstracts	SDS: 75 bytes, MDS: 665b
DUC-2005 [Dang, 2005]	MDS	news	QF-C	50×[25-50]	abstracts	250 w
DUC-2006 [Dang, 2006b]	MDS	news	QF-C	50×25	abstracts	250 w
DUC-2007 [Dang, 2007]	MDS	news	QF-C, U	45×25	abstracts	QF-C: 250 w, U: 100 w
TAC-2008 [Dang and Owczarzak, 2008]	MDS	news	U	48×20	abstracts	100 w
TAC-2009 [Dang and Owczarzak, 2009]	MDS	news	U	44×20	abstracts	100 w
TAC-2010 [Owczarzak and Dang, 2010]	MDS	news	Gu + U	44×20	abstracts	100 w
TAC-2011 [Owczarzak and Dang, 2011]	MDS	news	Gu + U	44×20	abstracts	100 w
TAC-2014 [Cohen et al., 2014]	MDS	scientific	G	20×11	abstracts	250 w
CNN+DM [Nallapati et al., 2016b]	SDS	news	G	310K	highlights	50 w
GigaWord [Rush et al., 2015]	SDS	news	G	4M+	title + 1st sent	headline (8w)
XSum [Narayan et al., 2018a]	SDS	news	G	220K	preface	1 sentence (23w)
NYTimes [Sandhaus, 2008]	SDS	news	G	650K	teaser	38 w
NewsRoom [Grusky et al., 2018]	SDS	news	G	1.3M	abstracts	27w
Arxiv [Cohan et al., 2018]	SDS	scientific	G	215K	abstracts	220 w
PubMed [Cohan et al., 2018]	SDS	scientific	G	133K	abstracts	203 w
Reddit-TIFU [Kim et al., 2019]	SDS	social-media	G	123K	tl;dr	short: 9w, long: 23w
Webis-TLDR-17 [Völske et al., 2017]	SDS	social-media	G	2.3M comments 1.6M posts	tl;dr abstracts	comments: 22 w posts: 34 w
Webis-Snippet-20 [Chen et al., 2020c]	SDS	web	QF-S	3.5M+	snippet	59 w
BigPatent [Sharma et al., 2019b]	SDS	patent	G	1.3M	abstracts	116w
BillSum [Kornilova and Eidelman, 2019]	SDS	legal	G	33K	abstracts	-
Aeslc [Zhang and Tetreault, 2019]	SDS	email	G	18K	subject	4w
WikiHow [Koupaee and Wang, 2018]	SDS	task-guide	G	230K	abstracts	62w
AusLegal [Galgani et al., 2012b]	MDS	legal	G	2814	catchphrases	catchphrases
Opinosis[Ganesan et al., 2010]	MDS	reviews	G	51×100	abstracts	17 w
MultiNews [Yasunaga et al., 2017]	MDS	news	G	56K×[2-10]	abstracts	260w

Table 2.2: A list of Summarization datasets in English language. We provide the *input* (SDS or MDS), *applications*, *dataset size*, *ground truth* creation method and *output size*. The *applications* are Generic, Viewpoint, Query-Focused-Simple, Query-Focused-Complex, Update, Novelty, and Guided. The *output size* is the average size of ground truth summaries. *Dataset size* for MDS datasets are in the form $T \times N$, (number of topics times number of documents per topic).

generally have few hundred articles in each year and have human multiple written abstracts for each article / topic of documents. In OPINIOSES, the authors employ crowd to create ground-truth summaries for user reviews [Ganesan et al., 2010].

Recent large news datasets, such as MULTINEWS [Yasunaga et al., 2017], and NEWSROOM [Grusky et al., 2018] make use of professionally written summaries by authors and editors. Scientific papers come with abstracts written by authors, and hence were used to create ARXIV and PUBMED [Cohan et al., 2018]. Similaryly, this was done for patent documents in BIGPATENT [Sharma et al., 2019b]. In *reddit*, authors of the posts and comments often provide a summary as 'TL;DR', hence it was leveraged to create WEBIS-TLDR-17 [Völske et al., 2017], and REDDIT-TIFU[Kim et al., 2019]. For congressional bills discussions, Congressional research service (CRS) provide professionally written summaries, which was used to create BILLSUM [Sharma et al., 2019b]. In online website *W*ikiHow, *h*ow-to questions are answered in paragraphs of steps with each paragraph starting with a short summary sentence, which was used to create WIKIHOW dataset [Kornilova and Eidelman, 2019].

While it is costly to employ human to write summaries on a new dataset, most news articles already come with human written highlights (CNN/DM, Nallapati et al. [2016b]), teasers (NYTIMES, Sandhaus [2008]), and headlines (XSUM Narayan et al. [2018a]). Also, news is often written in such a way that the first few sentences form a summary [Kedzie et al., 2018], and hence can be used as summaries as in GIGAWORD dataset [Rush et al., 2015]. This is also known as inverted pyramid in journalism[8]. Galgani et al. [2012a] used catchphrases of the legal case reports as ground truth summaries to create AUSLEGAL dataset. Chen et al. [2020c] employed web search engine to create a large scale snippets dataset WEBIS-SNIPPET-20, i.e. the ground truth are the summaries returned by search engine for the query term and source document is the linked webpage. The AESLC is a dataset to generate an email subject, which are much shorter summaries [Zhang and Tetreault, 2019].

As is evident from Table 5.1, most of the datasets in the summarization literature are applicable to generic (both SDS and MDS) summarization. TAC-2008 and TAC-2009 are applicable to update summarization, and hence we use this dataset in evaluating supervised methods developed in Chapter §5. We additionally use the DUC-2003 and DUC-2004 datasets as well, as these are widely used in the multi-document summarization literature, and the summaries on this dataset from various methods are made openly available by Hong et al. [2014], making it easy to compare the methods we develop with existing works. In evaluating unsupervised methods, we instead develop a large scale controversial news dataset in Chapter 3. The dataset we develop allows us to compare news articles over time (Chapter 4) and across ideologies of news outlets (Chapter 6). We next review different methods to evaluate summaries.

[8] https://en.wikipedia.org/wiki/Inverted_pyramid_(journalism)

	Intrinsic	Extrinsic
Auto	Content similarity [Saggion et al., 2002] ROUGE [Lin, 2004] BE [Hovy et al., 2006]	Classification [Bien and Tibshirani, 2011; Kim et al., 2016] Retrieval relevance [Goldstein et al., 1999]
Human	SEE [Over, 2001] Linguistic Quality [Over, 2001] Pyramid [Nenkova et al., 2007]	Question answering [Morris et al., 1992; Over et al., 2007] Human classification [Mani et al., 1999; Kim et al., 2016] Relevance assessment [Mani et al., 1999]

Table 2.3: Classification of summarization evaluation strategies based on how we evaluate summaries (human and automatic), and what we evaluate (intrinsic and extrinsic qualities).

2.4 Summarization evaluation

In this section, we first provide a brief overview of different summarization evaluation strategies. Summarization evaluations can be categorized as *intrinsic* and *extrinsic* based on what summaries qualities we evaluate, or *automatic* and *human* evaluations based on how we evaluate the summaries. First, we review different automatic evaluations, and then provide a detailed overview of ROUGE, which is widely used in the summarization literature. We use ROUGE in evaluating our supervised method developed in Chapter 5. Finally, we review different human evaluation strategies of summaries. In Chapter 4, we design extrinsic automatic and human evaluations for comparative summarization addressing the second research question of the thesis.

2.4.1 Automatic evaluations

Summarization evaluation can be *intrinsic* or *extrinsic*, as in the case of other natural language processing (NLP) systems [Jones and Galliers, 1995]. *Intrinsic* evaluation is when we evaluate the inherent quality of summaries rather than the usefulness of summary on some secondary tasks. *Extrinsic* evaluation evaluates the usefulness of summaries in some secondary tasks. Both extrinsic and intrinsic evaluations can be automatic, or human based, which we summarize in Table 2.3. Here, we discuss the extrinsic and intrinsic automatic evaluations, and we will discuss human evaluations in §2.4.3.

Intrinsic evaluation assesses summaries for qualities such as coverage, or linguistic qualities. The most commonly used evaluation in the summarization literature, ROUGE, is an intrinsic automatic evaluation [Lin, 2004]. ROUGE has been a de-facto standard for automatically evaluating coverage of summaries in the DUC, TAC competitions and in recent datasets. Alternatively, we can make use of short fragments instead of n-grams called *Basic elements (BE)* [Hovy et al., 2006]. Basic elements are defined as the head of major syntactic constituents (noun, verb, adjective or adverbial phrases), expressed as a single item or, a relation between head-BE and a single dependent expressed as a head-modifier-relation triplet [Hovy et al.,

2006]. However, we only use ROUGE in Chapter 5, as it is standard in evaluating under the experimental settings we use.

Automatic intrinsic evaluations such as ROUGE require ground truth summaries, which are costly to prepare in larger datasets. Hence, extrinsic evaluation such as using classification performance provides a surrogate way to measure the usefulness of prototypes in prototype selection, or summaries [Kim et al., 2016; Bien and Tibshirani, 2011]. Another way to evaluate summaries is by accessing the sentence relevance on a query-focused summarization task, where the summary sentences have to be relevant to the user query [Goldstein et al., 1999]. In Chapter 4, we will develop an *extrinsic* evaluation of comparative summarization using classification accuracy (§4.6.3). We next review ROUGE in details.

2.4.2 Automatic evaluation with ROUGE

ROUGE (Recall Oriented Understudy for Gisting Evaluation) has been a very popular automatic evaluation metric of summaries [Lin, 2004]. The evaluation works by comparing number of overlapping units such as n-grams, word sequences, and word pairs between summaries generated by summarization algorithms and abstracts written by human [Lin, 2004]. ROUGE has been shown to correlate with human judgements [Lin, 2004; Graham, 2015], which were lacking in earlier attempts in content-overlap based automatic evaluation [Saggion et al., 2002]. We will now briefly describe ROUGE-N and ROUGE-SU4, which we use in evaluating summaries in Chapter 5.

Let S^t be the *human abstracts* and $\bar{S}^t$ be a *candidate summary* from a summarization algorithm to be evaluated. Since ROUGE works by comparing the number of overlapping units, let $\text{units}(\cdot)$ be a function that gives a set of units, e.g. n-grams, skip grams, etc. for each text piece. Then we define ROUGE recall, precision and F1 as:

$$\text{ROUGE}_{\text{recall}}(\bar{S}^t, S^t) = \frac{|\text{units}(\bar{S}^t) \cap \text{units}(S^t)|}{|\text{units}(S^t)|} \tag{2.2}$$

$$\text{ROUGE}_{\text{precision}}(\bar{S}^t, S^t) = \frac{|\text{units}(\bar{S}^t) \cap \text{units}(S^t)|}{|\text{units}(\bar{S}^t)|} \tag{2.3}$$

$$\text{ROUGE}_{\text{F1}}(\bar{S}^t, S^t) = \frac{2 \times \text{ROUGE}_{\text{recall}}(\bar{S}^t, S^t) \times \text{ROUGE}_{\text{precision}}(\bar{S}^t, S^t)}{\text{ROUGE}_{\text{recall}}(\bar{S}^t, S^t) + \text{ROUGE}_{\text{precision}}(\bar{S}^t, S^t)} \tag{2.4}$$

If the units are n-grams, we get ROUGE-N metrics. Skip-bigram is a bigram in a sentence with gaps [Lin, 2004]. For example, in sentence "Australia's capital is Canberra", we have $\binom{4}{2}$ bigrams, "australia's capital", "australia's is", "australia's canberra", "capital is", "capital canberra", "is canberra". The advantage of using skip-bigrams is that it is sensitive to the word order even though it does not exactly match the consecutive words [Lin, 2004]. It is often preferable to use the skip-distance of 4 while generating the bigram units, i.e. the bigrams

are generated if the two words appear within the distance of 4 words and also include the unigram units, as skip-bigrams would give zero match to the reverse sequence of exact same words. Altogether, this forms ROUGE-SU4, which is often used in automatic summarization evaluation [Dang and Owczarzak, 2008, 2009].

ROUGE recall measures the amount of units in the ground truth summary retained by the system generated summary. Recall is generally preferred over precision, making ROUGE a recall oriented metric compared to BLEU (Bilingual evaluation understudy) [Papineni et al., 2002], which is a precision oriented metric for evaluating quality of machine translated text against human generated ground truth [Lin, 2004].

Often summarization datasets have multiple reference summaries. For example DUC and TAC datasets §2.3 have four human written abstracts for each document (SDS) or topic (MDS). There are various ways to combine the ROUGE scores in presence of multiple references, we refer readers to Lin [2004] for the details. The authors of ROUGE suggested taking maximum of scores from various summaries [Lin, 2004]. But, it is standard practice to take the average of ROUGE scores with respect to individual reference summaries in benchmarks [Dang and Owczarzak, 2009; Hong et al., 2014], and hence we use this in our evaluation in Chapter 5.

2.4.3 Human evaluation of summaries

The summarization competitions by DUC and TAC used both automatic and human evaluation protocols to assess the quality of the summaries [Over et al., 2007]. *Extrinsic* evaluations such as decision audit in meetings summarization [Murray et al., 2009], question answering based on summaries [Morris et al., 1992], relevance assessments of summaries [Mani et al., 1999], recovering original categories from summaries [Mani et al., 1999], analyzing time required to classify from prototypes [Kim et al., 2016], responsiveness measuring information need [Over et al., 2007], etc. have been proposed with human assessments.

For intrinsic evaluations, linguistic qualities such as informativeness, coherence, grammatical correctness, referential clarity, etc. are generally evaluated using human evaluation [Over, 2001; Over and Liggett, 2002; Mani, 2001]. Intrinsic human based coverage evaluations such as Summary Evaluation Environment (SEE) [Over, 2001; Lin and Hovy, 2003] was used in earlier DUC competitions (DUC2001-DUC2004); and was later replaced with pyramid evaluations, where human annotated summary content-units (SCU) are arranged in pyramids using corpus-wide statistics [Nenkova et al., 2007]. A comprehensive set of protocols for human evaluations of summaries was designed by Dang [2005] for the DUC-2005 competition, and similar protocols were used by Chu and Liu [2019].

More recent works in the summarization literature use crowd-sourced human evaluations (extrinsic and intrinsic). Crowd sourced human evaluations are often used in summarization literature for two main reasons:

1. Complementing automatic evaluations : While ROUGE score is good at capturing coverage of extractive systems and shown to correlate with human judgements [Lin, 2004; Graham, 2015], it does not guarantee an increase in quality and readability of the output and over-penalizes abstractive systems [Kryscinski et al., 2019; Zhang et al., 2020]. Furthermore, abstractive summarization systems suffer from the problems of the natural language generation such as syntactic and semantic correctness [Nallapati et al., 2017]. Extractive summarization systems might suffer from the problem of coherence and referential clarity (e.g. unresolved pronouns). Hence, intrinsic human evaluations assessing the linguistic quality of the summaries in terms informativeness, fluency, coherence, non-redundancy, succinctness, grammatical correctness, and referential clarity, etc. are important to complement the ROUGE evaluation.

2. Assessing the usefulness of the summaries : Extrinsic evaluations can show that summaries are actually useful in real-world tasks and may help to make comparisons of the systems beyond intrinsic evaluations [Over et al., 2007]. For example, in query-focused summarization, the summaries should be relevant to the query text or question, hence question-answering can be used as an extrinsic human evaluation [Morris et al., 1992]. Narayan et al. [2018b] curated a set of questions covering important content for human evaluations with the intuition that summaries should cover the salient content. Summarization of a large corpus containing multiple categories should represent all categories, allowing human judges to classify accurately and in timely manner with just prototypes [Bien and Tibshirani, 2011; Kim et al., 2016]. In comparative summarization, the summaries should reveal the differences to the human, and we will use this intuition to build extrinsic automatic and human evaluations in Chapter 4.

Recent human evaluations : We now study a variety of recent works that use human evaluation and summarize the protocols used in Table 2.4. We only list the common quality measures avoiding the less frequent ones and group similar concepts together, like informativeness and relevance together, fluency and readability together, and succinctness and conciseness together. For extrinsic evaluations, question answering paradigm by Narayan et al. [2018b] are generally used, which asks participants a set of curated questions, with a goal of quantifying the degree to which summaries retain the salient information [Liu and Lapata, 2019b]. For intrinsic evaluations, several inherent linguistic qualities as shown in Table 2.4 are accessed. The comparison between the systems is generally done by presenting the participants with summaries from two systems and asking them to choose the better one and using best-worst scaling [Louviere et al., 2015] to analyze the preferences of human judgements [Liu and Lapata, 2019a]. Alternatively, Zhang et al. [2020] compared the system summaries with human and analyze with paired t-test. Despite having a variety of methods for summarization

	Intrinsic						*Extrinsic*
	Info/Rel	**Fl/Re**	**Su/Con**	**NR**	**Gr**	**Coh**	**QA**
Lebanoff et al. [2018]	✓	✓		✓			
Zhang et al. [2019]	✓	✓					
Liu and Lapata [2019a]	✓	✓	✓				✓
Yasunaga et al. [2017]	✓	✓		✓			
Zhou et al. [2018]	✓			✓			
Narayan et al. [2017]	✓	✓					
Cheng and Lapata [2016]	✓	✓					
Liu and Lapata [2019b]	✓	✓	✓				✓
Paulus et al. [2017]	✓	✓					
Narayan et al. [2018a]	✓	✓					✓
Hsu et al. [2018]	✓	✓	✓				
Sharma et al. [2019a]	✓				✓	✓	
Xu and Durrett [2019]					✓		
Parveen et al. [2015]						✓	
Narayan et al. [2018b]	✓	✓					✓
Zhang et al. [2020]		✓				✓	
Zhang and Tetreault [2019]	✓	✓					
Grusky et al. [2018]	✓	✓				✓	
Kim et al. [2019]	✓						

Table 2.4: Summary of Human evaluation in recent summarization papers. The intrinsic qualities are: 1) Informativeness / Relevance, 2) Fluency / Readability, 3) Succinctness/ Conciseness, 4) Non-Redundancy, 5) Grammaticality, 6) Coherence, and Question-Answering as an extrinsic quality.

evaluation, Kryscinski et al. [2019] identified that current evaluation protocols do not evaluate the factual consistencies and such factual inconsistencies are substantial in their small pilot study.

2.5 Comparing document collections

In presence of multiple *groups* within a corpus as introduced in §2.2, one can develop several systems to facilitate comparisons. The goal of such systems is to identify the patterns that are common and/or different among the different document groups. We identify four types of comparison systems that have been frequently studied in the literature – comparative topic models, comparative document analysis, visual comparisons and comparative summarization. The key difference between these systems is the output - topics, list of comparative phrases, visualization, or a summary respectively. We use comparative summarization as a way of comparing document collections as summaries provide a much detailed outlook to the differences between document groups compared to the topics or visualizations. We have already discussed comparative summarization in §2.1.6, we now briefly review the other three approaches.

Comparative topic modeling : Topic modeling has been widely used in text analysis to reveal hidden topics in a document corpus. A *topic* is a probability distribution over vocabulary (please note the difference in term 'topic' from the one introduced in §2.2, where topic describes an abstract concept describing an event). Probabilistic Latent Semantic Analysis (pLSA), which is a probabilistic factorization of document-word co-occurances into document-topic and topic-word distributions, and Latent Dirichlet Allocation (LDA), which is a bayesian version of pLSA with *dirichlet* priors for the two distributions are common topic models [Blei et al., 2003].

Comparative topic models find the *topics* that are *similar* and *different* between the *groups*. Approaches include non-negative matrix factorization (NNMF) [Kim et al., 2015b], LDA like joint-modeling of documents and group, i.e. response variable in supervised setting [Mcauliffe and Blei, 2008; Paul, 2009; Lacoste-Julien et al., 2009; Hua et al., 2020], and hierarchical modeling with transformed Pitman-Yor process (TPYP) to incoorporate prior knowledge such as vocabulary variations in different groups [Chen et al., 2014]. Such topic models have been used to compare scientific conferences [Kim et al., 2015b], descriptions of similar business [Kim et al., 2015b], blogs based on political ideology [Chen et al., 2014], comparative summarization of scientific papers [He et al., 2016], discover cultural differences in blogs and forums, discovering scientific topics across multiple disciplines, and comparing editorial differences between multiple media sources [Paul, 2009]. The main difference between comparative topic modeling and comparative summarization is the unit of output, which is a list of comparative topics here, but a list of comparative summary sentences in comparative summarization.

Comparative document analysis : Identifying a list of comparative phrases between the document groups using graph based methods is *Comparative Document Analysis* [Ren et al., 2017]. In this system, a list of phrases highlights the differences is the unit of output.

Visual comparisons : *Visual comparisons* facilitate comparisons by visually presenting the information that reveals the differences and similarities between document groups. One way is to visualize the comparative topics based on the topic-word distributions [Paul, 2009; Chen et al., 2014; Kim et al., 2015b]. Shin et al. [2019] presented a graph-visualization system to visualize the differences across document groups focusing on entities and relationships. Visual comparison systems using word clouds, where words are arranged in 2D layout revealing the differences between document groups has been explored [Diakopoulos et al., 2015; Le and Lauw, 2016]. Alternatively, joint modeling of topic models and visual mappings can also facilitate comparative analysis [Oelke et al., 2014; Le and Akoglu, 2019]. In such models, topics are laid in a 2D layout, rather than a single large cloud of words together as in word cloud systems. Thus, such systems can reveal the nuances in similarities and differences with topics. The unit of output in these systems is a visualization.

2.6 Preliminaries: Kernels, RKHS, and MMD

In this section, we first provide a brief overview of Hilbert spaces, reproducing kernel Hilbert spaces (RKHS) and Maximum mean discrepancy (MMD). These form important background for Chapter 4 and Chapter 5, in which we develop unsupervised and supervised methods for comparative summarization, addressing $RQ1$ and $RQ3$ of the thesis. We then briefly review MMD as a prototype selection method as introduced by Kim et al. [2016], allowing us to use it as a method of extractive summarization. For a detailed overview, we refer readers to Daners [2008]; Muandet et al. [2017]; Sejdinovic and Gretton [2012] and Gretton et al. [2012a].

2.6.1 Topological spaces and Hilbert spaces

A space is a set with some additional structure. There is a hierarchy of different spaces in functional analysis. We briefly introduce some spaces here and refer readers to Daners [2008] for a more formal introduction. The top of the hierarchy is a *topological space* $\mathcal{X}$, which adds the concept of a neighborhood to the elements in the set . Intuitively, a subset of points within a small neighborhood of $x \in \mathcal{X}$ stays together on applying a continuous function. A *metric space* $(\mathcal{X}, \mathrm{d})$ is a topological space with a distance metric $\mathrm{d} : \mathcal{X} \times \mathcal{X} \mapsto [0, \infty)$. A metric space is a space where we can measure the distance between points/ A *vector space* $\mathcal{V}$ (or a linear space) over $\mathbb{R}$ is a space of set of objects that is closed under addition ($+ : \mathcal{V} \times \mathcal{V} \mapsto \mathcal{V}$), and scalar multiplication ($\cdot : \mathbb{R} \times \mathcal{V} \mapsto \mathcal{V}$)[9].

We now define a normed, inner product, function, and Hilbert space, which gradually add more structure to a metric and a vector space. Hilbert spaces form the basis of RKHS which we will introduce in §2.6.2.

Definition 2.2 (Normed space)**.** A non-negative function $\|\cdot\|_{\mathcal{V}} : \mathcal{V} \mapsto [0, \infty)$ is a norm if and only if $\forall_{f,g \in \mathcal{V}}\ \forall_{\alpha \in \mathbb{R}}$:

(i) $\|f\|_{\mathcal{V}} \geq 0$ and $\|f\|_{\mathcal{V}} = 0$ iff $f = 0$ (positive definiteness).

(ii) $\|f + g\|_{\mathcal{V}} \leq \|f\|_{\mathcal{V}} + \|g\|_{\mathcal{V}}$ (triangle inequality).

(iii) $\|\alpha f\|_{\mathcal{V}} = |\alpha| \|f\|_{\mathcal{V}}$ (positive homogeneity).

A normed space $(\mathcal{V}, \|\cdot\|_{\mathcal{V}})$ is a vector space in which norm is defined.

The norm induces a distance metric, i.e., $\forall_{f,g \in \mathcal{V}}\ \mathrm{d}(f, g) = \|f - g\|_{\mathcal{V}}$. A normed vector space is thus a space where we can measure the length of a vector.

Definition 2.3 (Inner product space)**.** A function $\langle \cdot, \cdot \rangle_{\mathcal{V}} : \mathcal{V} \times \mathcal{V} \mapsto \mathbb{R}$ is an inner product on $\mathcal{V}$ if and only if $\forall_{\alpha,\beta \in \mathbb{R}}\ \forall_{f,g,h \in \mathcal{V}}$:

(i) $\langle \alpha f + \beta g, h \rangle_{\mathcal{V}} = \alpha \langle f, h \rangle_{\mathcal{V}} + \beta \langle g, h \rangle_{\mathcal{V}}$ (bilinearity).

[9] A Vector space is defined over any field F, but we only focus on $\mathbb{R}$, which is called as a real vector space.

(ii) $\langle f, g\rangle_{\mathcal{V}} = \langle g, f\rangle_{\mathcal{V}}$ (symmetry).

(iii) $\langle f, f\rangle_{\mathcal{V}} \geq 0$ and $\langle f, f\rangle_{\mathcal{V}} = 0$ iff $f = 0$ (positive definiteness).

An Inner product space $\mathcal{V}, \langle\cdot,\cdot\rangle_{\mathcal{V}}$ is a normed space an inner product.

An inner product space is a space where we can measure the angle between vectors. An inner product defines a norm $\|f\|_{\mathcal{V}} = \sqrt{\langle f, f\rangle_{\mathcal{V}}}$. An additional property of norm is Cauchy-Schwarz inequality, i.e., $|\langle f, g\rangle_{\mathcal{V}}| \leq \|f\|_{\mathcal{V}}\|g\|_{\mathcal{V}}$.

Definition 2.4 (Hilbert Space)**.** A Hilbert space $\mathcal{H}$ is a complete[10] inner product space.

Intuitively, a Hilbert space is a space where we can measure distances (and lengths), angles, and take limits (the limit point of a converging sequence will be in the Hilbert space). A Hilbert space generalizes the finite Euclidean vector space $\mathbb{R}^d$ to infinite dimensions.

Definition 2.5 (Function space)**.** A function space $\mathcal{F}$ is a space of set of functions $f \in \mathcal{F} : \mathcal{X} \mapsto \mathbb{R}$ with some structure. The functions $\mathcal{X} \mapsto \mathbb{R}$ can have a structure of vector space over $\mathbb{R}$ where the operations (addition and scalar multiplication) are defined pointwise, i.e., $\forall_{f,g\in\mathcal{F}}\ \forall_{x\in\mathcal{X}}\ \forall_{\alpha\in\mathbb{R}}$

(i) $(f + g)(x) = f(x) + g(x)$.

(ii) $(c \cdot f)(x) = c \cdot f(x)$.

We next introduce RKHS, which is a Hilbert space of functions with some more structure.

2.6.2 Kernels, RKHS, and Kernel mean embeddings

An RKHS (reproducing kernel Hilbert space) is a Hilbert space of functions with a *reproducing kernel* which we define in this section. We then see some useful properties of RKHS. Finally, we define the kernel mean embeddings, and characteristic kernels which is useful in introducing MMD in §2.6.3. For a detailed overview, we refer readers to Aronszajn [1950]; Sejdinovic and Gretton [2012].

Definition 2.6 (Kernel [Mercer, 1909])**.** Let $\mathcal{X}$ be a non-empty set[11]. A function $\mathrm{k} : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is called a *kernel*, or a *positive definite kernel* if it is symmetric, i.e. $\forall_{x,y\in\mathcal{X}}\ \mathrm{k}(x, y) = \mathrm{k}(y, x)$, and the Gram matrix[12] is positive semi-definite, i.e. $\forall_{x_i,x_j\in\mathcal{X}}\ \forall_{n\in\mathbb{N}}\ \forall_{u_1,u_2,..u_n\in\mathbb{R}}$, $\sum_{i,j=1}^{n} u_i u_j \mathrm{k}(x_i, x_j) \geq 0$.

[10] A sequence of elements from normed vector space, i.e., $\{f_n\}_{n=1}^{\infty} \in (\mathcal{V}, \|\cdot\|_{\mathcal{V}})$ is **Cauchy sequence** if $\forall_{\epsilon>0}\ \exists_{N\in\mathbb{N}} : \forall_{n,m\geq N}\ \|f_n - f_m\|_{\mathcal{V}} < \epsilon$. Cauchy sequences are always bounded, i.e., $\forall_{n\in\mathbb{N}}\ \exists_{M<\infty}\ : \|f_n\|_{\mathcal{V}} \leq M$. A normed vector space $\mathcal{V}$ is **complete** if every Cauchy sequence in $\mathcal{V}$ converges to an element in $\mathcal{V}$, i.e., a sequence of points that get closer to each other converges to some point in that space.

[11] $\mathcal{X}$ is typically a domain of observations, e.g. documents, images. It might or might not have some structure like a topological space, metric space.

[12] Gram matrix $\mathbf{K} \in \mathbb{R}^{n\times n}$ is defined by $k_{ij} = \mathrm{k}(x_i, x_j)$, and is positive definite i.e., $\forall_{\mathbf{u}\in\mathbb{R}^n}\ \mathbf{u}^T\mathbf{K}\mathbf{u} \geq 0$.

Examples of kernels on $\mathbb{R}^d$ include the polynomial kernel $\mathrm{k}(x, y) = \langle x, y\rangle^p, p \geq 0$ and the Gaussian RBF kernel $\exp(-\gamma\|x - y\|^2), \gamma \geq 0$.

Definition 2.7 (Reproducing kernel [Aronszajn, 1950])**.** Let $\mathcal{H}$ be a Hilbert space of functions over $\mathcal{X}$. A kernel $\mathrm{k} : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$ is a reproducing kernel of $\mathcal{H}$ if:

(i) $\forall_{x\in\mathcal{X}}\ \mathrm{k}(\cdot, x) : \mathcal{X} \mapsto \mathbb{R} \in \mathcal{H}$.

(ii) $\forall_{x\in\mathcal{X}}\ \forall_{f\in\mathcal{H}}\ f(x) = \langle \mathrm{k}(x, \cdot), f(\cdot)\rangle_{\mathcal{H}}$ (reproducing property).

Reproducing kernels are positive definite [Sejdinovic and Gretton, 2012].

Definition 2.8 (Reproducing kernel Hilbert spaces (RKHS) [Aronszajn, 1950])**.** A Hilbert space $\mathcal{H}$ of functions on $\mathcal{X}$ is RKHS if there exists a reproducing kernel k.

A RKHS is fully characterized by a positive definite kernel, i.e., a positive definite kernel uniquely determines a RKHS and vice versa [Muandet et al., 2017]. For a given RKHS $\mathcal{H}$, $\mathrm{k}(\cdot, x) = \phi(x)$, $\phi : \mathcal{X} \mapsto \mathcal{H}$ is known as the canonical feature map, or Aronszajn's map [Sejdinovic and Gretton, 2012; Aronszajn, 1950]. Hence, for every function in RKHS ($f : \mathcal{X} \mapsto \mathbb{R} \in \mathcal{H}$), there is a feature map ϕ such that $f(x) = \langle f, \phi(x)\rangle_{\mathcal{H}}$.

Learning with RKHS : Having introduced the notion of RKHS, now we see the usefulness of RKHS in machine learning. This primarily comes from the *kernel trick*, by which we can use kernels to learn functions in RKHS without explicitly mapping to the RKHS. Second, the learned functions in RKHS are well-behaved and generalize well to unseen datasets [Muandet et al., 2017].

Theorem 2.1 (Kernels as inner products [Aronszajn, 1950])**.** For a positive definite kernel k, there exists an RKHS $\mathcal{H}$ and a feature map $\phi : \mathcal{X} \mapsto \mathcal{H}$ such that:

$$\forall_{x,y\in\mathcal{X}}\ \mathrm{k}(x, y) = \langle \phi(x),\ \phi(y)\rangle_{\mathcal{H}} = \langle \mathrm{k}(\cdot, x), \mathrm{k}(\cdot, y)\rangle_{\mathcal{H}}.$$

Many machine learning algorithms such as support vector machine (SVM) [Cortes and Vapnik, 1995] and principal component analysis (PCA) [Pearson, 1901] only need to access the data through the inner product $\langle \mathbf{x}_i, \mathbf{x}_j\rangle$. In such methods, Theorem 2.1 allows us to avoid explicit mapping from $\mathcal{X} \mapsto \mathcal{H}$ when we extend linear algorithms to non-linear via higher (possibly infinite) dimensional non-linear feature mappings. This is known as the *kernel trick* in machine learning. An example of such kernel is Gaussian RBF kernel $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma\|\mathbf{x}_i - \mathbf{x}_j\|_2^2)$. Next we see why learned functions f in RKHS are well-behaved.

Theorem 2.2 (Aronszajn [1950])**.** In RKHS, all function evaluations are bounded, i.e.

$$\forall_{x\in\mathcal{X}}\ \forall_{f\in\mathcal{H}}\ \ |f(x)| \leq \sqrt{\mathrm{k}(x, x)}\|f\|_{\mathcal{H}}.$$

This means that the functions $f \in \mathcal{H}$ learned with RKHS norm regularization ($\|f\|_{\mathcal{H}}$) are smooth and well-behaved as function evaluations are bounded by the function norm, i.e., regularization with RKHS norm controls the model variations improving generalization to unseen test data [Muandet et al., 2017].

Kernel mean embeddings and characteristic kernel : Now we define kernel mean embedding, which is a mapping of a probability distribution (or a set of data-points) to an RKHS. Then, we define the reproducing property of the expectation in RKHS and characteristic kernels which are useful in developing MMD later in §2.6.3. For details, we refer readers to Smola et al. [2007]; Muandet et al. [2017].

Definition 2.9 (Kernel mean embeddings [Smola et al., 2007])**.** For a distribution p, and kernel with feature map $\phi\colon \mathcal{X} \mapsto \mathcal{H}$, the *kernel mean embedding* is

$$\mu_p = \mathbb{E}_{x\sim p}\left[\phi(x)\right].$$

Theorem 2.3 (Reproducing peoperty of expectation in RKHS [Smola et al., 2007])**.** For kernels with $\mathbb{E}_p[\sqrt{\mathrm{k}(x,x)}] < \infty$, and $f \in \mathcal{H}$, $\mathbb{E}_p[f(x)] = \left\langle f,\ \mathbb{E}_p[\phi(x)]\right\rangle_{\mathcal{H}}$.

Definition 2.10 (Characteristic kernel and RKHS [Muandet et al., 2017])**.** A kernel k is characteristic if the map $\mu : p \mapsto \mu_p$ is injective. The RKHS $\mathcal{H}$ is characteristic if its reproducing kernel is characteristic.

A characteristic kernel ensures $\|\mu_p - \mu_q\|_{\mathcal{H}} = 0$ if and only if $p = q$, i.e., no information is lost in mapping the distribution into the RKHS. An RKHS with characteristic kernel should contain a sufficiently rich class of functions to represent all higher order moments of p [Muandet et al., 2017]. The linear kernel $k(\mathbf{x},\mathbf{y}) = \langle \mathbf{x},\mathbf{y}\rangle$, and polynomial kernel $k(\mathbf{x},\mathbf{y}) = (\langle \mathbf{x},\mathbf{y}\rangle + c)^r, r > 0$ are not characteristic. Examples of characteristic kernels for $\mathbb{R}^d$ are: Gaussian kernel: $k(\mathbf{x},\mathbf{y}) = \exp(-\gamma\|\mathbf{x}-\mathbf{y}\|_2^2),\ \gamma > 0$, Laplace kernel: $k(\mathbf{x},\mathbf{y}) = \exp(-\gamma\|\mathbf{x}-\mathbf{y}\|_1),\ \gamma > 0$, and Exponential kernel: $k(\mathbf{x},\mathbf{y}) = \exp(\gamma\langle \mathbf{x},\mathbf{y}\rangle),\ \gamma > 0$.

2.6.3 Maximum Mean Discrepancy (MMD)

Here we introduce MMD, which is a kernel-based measure of the distance between two distributions (or two sets of data-points). A small MMD value indicates that the distributions are similar. MMD forms the important background for the thesis in Chapter 4 and Chapter 5 where we develop unsupervised and supervised methods for comparative summarization. More formally:

Definition 2.11 (Maximum mean discrepancy (MMD) [Gretton et al., 2012a])**.** Let $\mathcal{F}$ be the class of functions $h : \mathcal{X} \mapsto \mathbb{R}$ within the unit ball in an RKHS, i.e. $h \in \mathcal{H}, \|h\|_{\mathcal{H}} \leq 1$, where $\mathcal{X}$

is a topological space. Let p, q be two probability measures on $\mathcal{X}$. Then, the MMD between distributions p, q is the *maximal difference* in expectations of functions from $\mathcal{F}$ under:

$$\mathrm{MMD}_{\mathcal{F}}(p,q) = \sup_{h\in\mathcal{F}} \left\{ \mathop{\mathbb{E}}_{x\sim p}[h(x)] - \mathop{\mathbb{E}}_{y\sim q}[h(y)] \right\}. \tag{2.5}$$

MMD belongs to a class of distance measures between probability distributions called as *Integral probability metric (IPM)* [Sriperumbudur et al., 2009b]. It is the function class $\mathcal{F}$ that characterizes IPM. Restricting $\mathcal{F}$ to unit ball in RKHS give us MMD, but other choices of $\mathcal{F}$ give us other distance metrics such as Wasserstein distance, total variation distance, etc [Sriperumbudur et al., 2009b].

Theorem 2.4 (Gretton et al. [2012a])**.** For unit ball RKHS $\mathcal{F}$ with mean embeddings defined as Definition 2.9 and, $\mathbb{E}_{x\sim p}[\sqrt{\mathrm{k}(x,x)}] < \infty$, $\mathbb{E}_{y\sim q}[\sqrt{\mathrm{k}(q,q)}] < \infty$, MMD (2.5) is equivalent to the length of difference in kernel mean embeddings:

$$\mathrm{MMD}_{\mathcal{F}}(p,q) = \left\|\mu_p - \mu_q\right\|_{\mathcal{H}}. \tag{2.6}$$

Proof. We provide this proof because, a proof of Lemma 5.1 in Chapter 5 follows this proof. To prove the result, we first use Theorem 2.3, and the definition of *Dual norm* [Daners, 2008, p. 85] in third step:

$$\begin{aligned}
\mathrm{MMD}_{\mathcal{F}}(p,q) &= \sup_{\|h\|_{\mathcal{H}}\leq 1} \left\{ \mathop{\mathbb{E}}_{x\sim p}[h(x)] - \mathop{\mathbb{E}}_{y\sim q}[h(y)] \right\} \\
&= \sup_{\|h\|_{\mathcal{H}}\leq 1} \left\{ \left\langle h, \mathbb{E}_p[\phi(x)] \right\rangle - \left\langle h, \mathbb{E}_q[\phi(y)] \right\rangle \right\} \\
&= \sup_{\|h\|_{\mathcal{H}}\leq 1} \left\{ \left\langle h, \mathbb{E}_p[\phi(x) - \mathbb{E}_q[\phi(y)]] \right\rangle \right\} \\
&= \left\| \mathbb{E}_p\phi(x) - \mathbb{E}_q\phi(y) \right\|_{\mathcal{H}} \\
&= \left\| \mu_p - \mu_q \right\|_{\mathcal{H}}
\end{aligned}$$

■

A desirable property of function class $\mathcal{F}$ is that they must be rich enough so that MMD vanishes if and only if $p = q$. If $\mathcal{F}$ is characteristic (see Definition 2.10), i.e., $\mathrm{MMD}_{\mathcal{F}}(p,q)$ is 0 if and only if $p = q$, i.e., MMD is a valid distance metric. This is because the characteristic kernels uniquely map each probability distribution to the RKHS. MMD with the Gaussian kernel is equivalent to comparing all moments between two distributions [Li et al., 2015].

Equation (2.6) involves explicitly evaluating the arbitrarily high-dimensional features. Instead, the *kernel trick* allows efficient computation of $\mathrm{MMD}^2_{\mathcal{F}}(p,q)$ by evaluating just pairwise

kernels. Supposing $\mathcal{F}$ has induced kernel k, we have

$$\mathrm{MMD}^2_{\mathcal{F}}(p,q) = \mathop{\mathbb{E}}_{x,x'\sim p}\left[\mathrm{k}(x,x')\right] + \mathop{\mathbb{E}}_{y,y'\sim q}\left[\mathrm{k}(y,y')\right] - 2\mathop{\mathbb{E}}_{x\sim p,y\sim q}\left[\mathrm{k}(x,y)\right]. \tag{2.7}$$

s Given finite samples $X \sim p^N$ and $Y \sim q^M$, an empirical estimate (biased) of the MMD^2, denoted as $\mathrm{MMD}^2_{\mathcal{F}}(X,Y)$, can be computed as:

$$\mathrm{MMD}^2_{\mathcal{F}}(X,Y) = \frac{1}{N^2}\sum_{x,x'}\mathrm{k}(x,x') + \frac{1}{M^2}\sum_{y,y'}\mathrm{k}(y,y') - \frac{2}{N\cdot M}\sum_{x,y}\mathrm{k}(x,y). \tag{2.8}$$

MMD is shown to be equivalent to classifiability by Sriperumbudur et al. [2009a], hence we use it as a surrogate for classification accuracy in Chapter 4 to develop an unsupervised method for comparative summarization (RQ1). In Chapter 5, we introduce weighted MMD (wMMD), which can account for different sample weights and use it in developing methods for supervised extractive summarization (RQ3). We now see the usefulness of MMD in extractive summarization (or prototype selection).

2.6.4 Prototype selection using MMD

Kim et al. [2016] showed that MMD can be used to select prototypes (a small representative subset of data collection) in their method *MMD-Critic*. The intuition is that the prototypes represent the set of all datapoints, hence an optimal set of prototypes choice we seek to obtain should have a minimum MMD with the datapoints.

$$\operatorname*{Argmax}_{\bar{\mathbf{X}}} \; -\mathrm{MMD}^2(\bar{\mathbf{X}},\mathbf{X}). \tag{2.9}$$

From Equation (2.8), as we can ignore the first term because it does not affect the maximization, the MMD between N data-points ($\mathbf{X}$) and M prototypes ($\bar{\mathbf{X}}$) we seek would be:

$$\mathcal{U}_{mmd}(\bar{\mathbf{X}}) = -\frac{1}{M^2}\sum_{\bar{\mathbf{x}},\bar{\mathbf{x}}'}\mathrm{k}(\bar{\mathbf{x}},\bar{\mathbf{x}}') + \frac{2}{N\cdot M}\sum_{\mathbf{x},\bar{\mathbf{x}}}\mathrm{k}(\mathbf{x},\bar{\mathbf{x}}). \tag{2.10}$$

Two desirable properties of prototypes in extractive summarization are coverage and non-redundancy. The first term of Equation (2.10) is equivalent to minimizing redundancy between prototypes (or maximizing diversity), and second term is maximizing the information coverage of prototypes, which are the two desirable properties of extractive generic summaries. Hence, MMD is useful as a summarization (prototype selection) method.

Selecting prototypes by the discrete maximization in (2.9) is equivalent to generic summarization of each group of documents. Discrete optimization problems such as (2.9) are generally difficult to optimize. But, the objective (2.10) belongs to a special class of set func-

tions called *submodular-monotone* [Kim et al., 2016], and hence efficient greedy algorithms give reasonably good solutions [Nemhauser et al., 1978]. We next introduce *submodular-monotone* functions and their optimization strategies which are useful in optimizing objectives such as $\mathcal{U}_{mmd}$.

2.7 Preliminaries: Submodular and monotone functions

Here, we provide a brief overview of a class of set functions that exhibit a diminishing return property, called *submodular* functions. Submodular functions are well studied problems in computer science, and have applications in automatic summarization [Lin and Bilmes, 2011; Tschiatschek et al., 2014; Mirzasoleiman et al., 2016, 2017], influence maximization in social media [Kempe et al., 2003], electrical networks optimizations [Narayanan, 1997], active learning [Wei et al., 2015], sensor placements in water distribution networks [Krause et al., 2008], among others. Submodular functions naturally model extractive summarization and prototype selection, and can be efficiently optimized using greedy algorithms. Some of the methods we develop in this thesis are *submodular-monotone*, hence this introduction forms an important background material for the thesis.

First we introduce the concepts of submodularity and monotonicity. Second, we review some methods in extractive summarization literatures that are submodular and/or monotone. Then we review the conditions on kernel matrix for submodularity and monotonicity of $\mathcal{U}_{mmd}$ by Kim et al. [2016], and provide less restrictive conditions. Finally, we introduce a theorem by Nemhauser et al. [1978] which allows efficient greedy algorithms to be employed for discrete optimizations of submodular and/or monotone set functions, and see how we can use a greedy algorithm for a couple of summarization objectives.

2.7.1 Submodular and monotone set functions

Suppose we wish to maximize a utility set function $\mathcal{U} : 2^{|V|} \mapsto \mathbb{R}$, that assigns each subset $S \subseteq V$ a value $\mathcal{U}(S)$, where V is the ground set. An example of V is a set of all documents and S is a set of summary documents. A set function $\mathcal{U}$ is called *normalized* if $\mathcal{U}(\phi) = 0$.

Definition 2.12 (Discrete derivative [Krause and Golovin, 2014])**.** For $S \subset V$ and $\forall s \in V \setminus S$, the *marginal gain* of adding element s to an existing set S ($S + s = S \cup \{s\}$) is known as the *discrete derivative*, and is defined by

$$\Delta_{\mathcal{U}}(s|S) = \mathcal{U}(S + s) - \mathcal{U}(S).$$

Definition 2.13 (Monotonicity [Krause and Golovin, 2014])**.** $\mathcal{U}$ is *monotone* (monotonically in-

creasing) if and only if the discrete derivatives are non-negative , i.e.

$$\Delta_{\mathcal{U}}(s|S) \geq 0.$$

Definition 2.14 (Modular functions)**.** $\mathcal{U}$ is modular if and only if the marginal gain is always constant, i.e., for $S \subseteq T \subset V, s \in V \setminus T$,

$$\Delta_{\mathcal{U}}(s|S) = \Delta_{\mathcal{U}}(s|T)$$

Modular functions are also known as linear functions.

Definition 2.15 (Submodularity [Krause and Golovin, 2014])**.** $\mathcal{U}$ is submodular if and only if the marginal gain satisfies diminishing returns, i.e. for $S \subseteq T \subset V, s \in V \setminus T$,

$$\Delta_{\mathcal{U}}(s|S) \geq \Delta_{\mathcal{U}}(s|T)$$

Equivalently, for every $S, T \subseteq V$, $\mathcal{U}$ is submodular if,

$$\mathcal{U}(S \cap T) + \mathcal{U}(S \cup T) \leq \mathcal{U}(S) + \mathcal{U}(T)$$

If a utility function $\mathcal{U}$ is both submodular and monotone, then it is called *submodular-monotone*. The above definition show that submodular functions exhibit a diminishing return property, i.e., the marginal gain of adding an item s to a set is larger if we are adding to a smaller set. This idea is immediately applicable in extractive summarization: the gain in coverage of the summary is higher if we are adding a sentence to a set containing fewer sentences than to a set with many sentences. Next, we review several methods in the generic extractive summarization literatures leveraging the idea of submodularity and monotonicity.

2.7.2 Submodularity in summarization

Many extractive generic summarization objectives in the literature are naturally submodular [Lin and Bilmes, 2010, 2011; Li et al., 2012a; Mitrovic et al., 2018; Mirzasoleiman et al., 2016]. Submodularity naturally rises in the objectives promoting information coverage and diversity of the prototypes [Lin and Bilmes, 2011; Tschiatschek et al., 2014]. We now review several such objectives from the generic summarization literature. To our best knowledge, existing works do not generally use submodular objectives for comparative summarization.

Let V be the set of all N sentences (or documents) in a collection, and suppose we seek to find summary M prototype sentences $\bar{S}$. Let $\mathbf{K} = \mathbb{R}^{N \times N}$ with $k_{ij} = \mathrm{k(i,j)}$ be a kernel matrix (or Gram matrix). We also use notation k_{ij} for similarity between the sentences i and j in sentence-sentence graph. Let $\mathbf{K}_{TS}$ be the sub-matrix of K with rows corresponding to

$T \subset V$, and columns corresponding to $\bar{S}$. In such setting, some *submodular* and/or *monotone* summarization methods are:

1. FacLoc: Facility Location, $\mathcal{U}_{FL}(\bar{S}) = \sum_{v \in V} \max_{s \in \bar{S}} k_{vs}$, is often used in allocating facilities [Cornuéjols et al., 1983]. It is *submodular-monotone* and promotes the information coverage in summarization [Lin et al., 2009; Tschiatschek et al., 2014]. It is shown to be equivalent to maximizing the classifying ability of nearest-neighbor classifier due to prototypes [Wei et al., 2015].

2. SumCov: Sum coverage, $\mathcal{U}_{SC}(\bar{S}) = \sum_{v \in V} \sum_{s \in \bar{S}} k_{vs}$, is *submodular-monotone*[13], and promotes the information coverage [Tschiatschek et al., 2014].

3. TruncCov: Truncated coverage, $\mathcal{U}_{TC}(\bar{S}) = \sum_{v \in V} \min(\sum_{s \in \bar{S}} k_{vs}, \alpha \cdot \sum_{u \in V} k_{vu})$ with $0 \leq \alpha \leq 1$, promotes the coverage of entire documents compared to SumCov by encouraging the sentences not yet saturated with a higher chance of being covered [Lin and Bilmes, 2011]. It is *submodular-monotone*.

4. Div: Penalty based diversity, $\mathcal{U}_{div}(\bar{S}) = -\sum_{s,t \in S} k_{st}$, promotes the diversity between prototypes. It is submodular and non-monotone[14] [Simon et al., 2007; Tschiatschek et al., 2014].

5. Cut: Graph cut, $\mathcal{U}_{cut}(\bar{S}) = \sum_{v \in V \setminus S} \sum_{s \in \bar{S}} k_{vs} = \sum_{v \in V} \sum_{s \in \bar{S}} k_{vs} - \sum_{s,t \in S} k_{st}$ is submodular and non-monotone. Nevertheless, greedy algorithm can be used in practice as it respects the conditions of Theorem 2.6 when $M \ll N$ [Lin et al., 2009]. It incorporates coverage and some diversity, and is a combination of SumCov and Div.

6. CutDiv: Cut with additional Div, $\mathcal{U}_{mcut}(\bar{S}) = \sum_{v \in V} \sum_{s \in \bar{S}} k_{vs} - \eta \cdot \sum_{s,t \in S} k_{st}$, with $\eta \geq 1$, is submodular but not monotone, and can be optimized similarly to Cut [Lin et al., 2009]. It is shown to be similar to Maximal marginal relevance (MMR) method [Carbonell and Goldstein, 1998] by Lin et al. [2009].

7. ClusterDiv: Cluster based diversity. Let $P_1 \dots P_K$ be K clusters of the datapoints. $\mathcal{U}_{clu}(\bar{S}) = \sum_{j=1}^{K} g(\bar{S} \cap P_j)$, where $g(\cdot)$ is submodular-monotone, is a diversity promoting submoduar and monotone method [Lin and Bilmes, 2011].

8. LogDet: $\mathcal{U}_{ldet}(\bar{S}) = \log\det \mathbf{K}_{S,S}$ is submodular, and promotes the diversity of prototypes [Kim et al., 2016]. It is also known as entropy regularizer, and is monotone if the smallest eigenvalue of $\mathbf{K}_{SS} \geq 1$ [Sharma et al., 2015].

9. Feats: Feature based functions. Let $\mathcal{W}$ be the vocabulary of features, e.g., bag-of-words, and b_{vw} is feature w for sentence v. $\mathcal{U}_{feats}(\bar{S}) = \sum_{w \in \mathcal{W}} g(\sum v \in V b_{vw})$, where $g(\cdot)$ is submodular-

[13] SumCov is in-fact linear (or modular) function in $\bar{S}$, and greedy algorithm yields an exact solution.
[14] Div is monotonically decreasing, instead of our desired criteria of monotonically increasing.

monotone, is a feature coverage function. It is *submodular-monotone*, and scalable as it avoids $\mathcal{O}(n^2)$ sentence-sentence similarity matrix [Kirchhoff and Bilmes, 2014].

The above utility set functions promotes either coverage, or diversity, or both. The goal is to seek the prototypes $\bar{S}$ that maximizes such utility functions. Another such objective that is *submodular-monotone* is MMD, which we discuss next in detail.

2.7.3 Submodularity and monotonicity of MMD

MMD is *submodular-monotone* under certain conditions on the entries of kernel matrix [Kim et al., 2016, Corollary 3]. We first revisit these conditions, and provide a less restrictive conditions for selecting a few prototypes from a large collection. We use similar ideas to show our unsupervised comparative summarization methods in Chapter 4 are *submodular-monotone*.

Let $\mathbb{1}_{[a]}$ be an indicator function that takes value of 1 if condition a is true and 0 otherwise. Let V denote the set of all document indices we seek to summarize, and $\bar{S} \subset V$ be the set of indices of prototypes $\bar{\mathbf{X}}$. Kim et al. [2016] first showed that the MMD objective $\mathcal{U}_{mmd}(\bar{S})$ is a linear function of kernel matrix, then they derive the conditions on kernel matrix entries to be *submodular-monotone* for linear forms. Finally, they apply the conditions for $\mathcal{U}_{mmd}(\bar{S})$.

Lemma 2.1 (Linear forms of $\mathcal{U}_{mmd}(\bar{S})$ [Kim et al., 2016])**.** $\mathcal{U}_{mmd}(\bar{S})$ with $m = |\bar{S}| \leq M$ in Equation (2.10) is a linear function of kernel matrix $\mathbf{K}$ as: $\mathcal{U}_{mmd}(\bar{S}) = \langle \mathbf{A}(\bar{S}), \mathbf{K} \rangle$, where

$$A_{ij}(\bar{S}) = \frac{2}{N \cdot m} \mathbb{1}_{[j \in \bar{S}]} - \frac{1}{m^2} \mathbb{1}_{[i \in \bar{S}]} \mathbb{1}_{[j \in \bar{S}]}.$$

Theorem 2.5 (Submoularity and monotonicity of Linear forms [Kim et al., 2016])**.** Let $\mathbf{H} \in \mathbb{R}^{N \times N}_{\geq 0}$ be non-negative and bounded (non necessarily symmetric like $\mathbf{K}$) with upper bound h_*. Let $\mathbf{E} \in \{0,1\}^{N \times N}$ with $e_{ij} = 1$ if $h_{ij} = h_*$ and 0 otherwise, and $\mathbf{E}' = 1 - \mathbf{E}$. Let $a(\bar{S}) = \langle \mathbf{A}(\bar{S}), \mathbf{E} \rangle$, and $b(\bar{S}) = \langle \mathbf{A}(\bar{S}), \mathbf{E}' \rangle$ with $m = |S|$, then $\forall_{s,t \in V}$ define the functions:

$$\alpha(N,m) = \frac{a(\bar{S}+s) - a(\bar{S})}{b(\bar{S})}, \;\; \beta(N,m) = \frac{a(\bar{S}+s) + a(\bar{S}+t) - a(\bar{S} \cup \{s,t\}) - a(\bar{S})}{b(\bar{S} \cup \{s,t\}) + b(\bar{S})}. \tag{2.11}$$

for $m \leq M$, the maximal cardinality of the prototypes $\bar{S}$ we seek. Then, $\langle \mathbf{A}(\bar{S}), \mathbf{H} \rangle$ is:

(i) *monotone* if $h_{ij} \leq h_* \alpha(N,m)$,

(ii) *submodular* if $h_{ij} \leq h_* \beta(N,m)$.

Kim et al. [2016] used Theorem 2.5 to show $\mathcal{U}_{mmd}$ is *submodular-monotone* for a kernel matrix that is non-negative, has equal diagonal terms $k_{ii} = k_*$ and the off-diagonal terms satisfy $0 \leq k_{ij} \leq \frac{k_*}{N^3+2N^2-2N-3}$ [Kim et al., 2016, corollary 3]. The condition holds for $M = N$, hence the bound is function of N. We instead provide a bound which is a function of M, the maximum cardinality of the prototypes we seek. The bound is less restrictive on the entries of kernel matrix for selecting a few prototypes, which is more practical in summarization.

Corollary 2.1 (Submodularity and monotonicity of $\mathcal{U}_{mmd}$)**.** Consider a kernel matrix that is non-negative, has equal diagonal terms $k_{ii} = k_*$. If the off-diagonal terms satisfy $k_{ij} \leq \frac{k_*}{(M+1)(M^2+3M+1)}$, $\mathcal{U}_{mmd}$ is *submodular-monotone*.

Proof. For kernel matrix satisfying the stated conditions, we have $\mathbf{E} = \mathbf{I}$. Then, following the notation from Theorem 2.5 definitions, $a(\bar{S}) = \frac{2}{N \cdot m} m - \frac{1}{m^2} m = \frac{2}{N} - \frac{1}{m}$ and $b(\bar{S}) = \frac{2}{N \cdot m}(N \cdot m - m) - \frac{1}{m^2}(m^2 - m) = \frac{2(N-1)}{N} - \frac{m-1}{m}$. Let $\epsilon = \frac{2}{N}$, then $a(\bar{S}) = \epsilon - \frac{1}{m}$ and $b(\bar{S}) = \frac{m+1}{m} - \epsilon$. Recall, $m = |S|$, so $a(\bar{S} + s) = \epsilon - \frac{1}{m+1}$ and so on. Then, applying these to Equation (2.11), we get $\alpha(N, m) = \frac{1}{(m+1)(m+1-\epsilon m)} > \frac{1}{(m+1)^2}$. Hence, by Theorem 2.5, an upper bound on the kernel entries $k_{ij} \leq \frac{k_*}{(M+1)^2}$ is sufficient for *monotonicity*. Similarly, we get $\beta(N, m) = \frac{1}{(m+1)(m^2+3m+1-\epsilon m(m+2))} > \frac{1}{(m+1)(m^2+3m+1)}$. Since, $(M^2 + 3M + 1) \geq (M + 1) \forall_{M>0}$ a combined upper bound on the kernel entries $k_{ij} \leq \frac{k_*}{(M+1)(M^2+3M+1)}$ is sufficient to guarantee both *monotonicity* and *submodularity*. ■

In Chapter 4, we develop unsupervised methods for comparative summarization using MMD, and use a similar proof to show that methods are *submodular-monotone*. Next we discuss efficient optimization strategies for such submodular and/or monotone objectives.

2.7.4 Maximizing *submodular-monotone* functions

Maximization of submodular functions has been well studied in the literature [Krause and Golovin, 2014; Nemhauser et al., 1978]. The maximization seeks to find an optimal subset S^*, and takes the form:

$$\bar{S}^* = \underset{S \in \mathcal{S}}{\text{Argmax}}\, \mathcal{U}(S). \tag{2.12}$$

Here, $S \in \mathcal{S} : \mathcal{S} \subset 2^{|V|}$ are the constraints we impose on feasible solutions. We introduced a couple of constraints that naturally arise in extractive summarization and prototype selection in §2.2: *cardinality* constraint and *budget* constraint. Cardinality constraints ($|S| \leq M$) seek to find a subset within some predefined size. Budget constraints ($\sum_{s \in \bar{S}} c(s) \leq B$) seeks to find a subset with some cost associated with each item $c(\cdot)$ within some predefined budget B. Submodular maximization problems are often NP-Hard [Krause and Golovin, 2014], but for maximizing *submodular-monotone* functions under cardinality constraints, an efficient greedy algorithm provides a reasonably good solution [Nemhauser et al., 1978]. The greedy algorithm works by starting with an empty set, i.e., $S^* = \{\}$, and iteratively adding an item with the best marginal gain (or discrete derivative) to it, i.e., $S^* \leftarrow S^* \cup \{\underset{s}{\text{Argmax}}\, \Delta_{\mathcal{U}}(s|S^*)\}$. A greedy algorithm for such utility set functions is provided in Chapter 4 (see Algorithm 1).

Theorem 2.6 (Nemhauser et al. [1978])**.** For a non-negative, normalized *submodular-monotone* set function $\mathcal{U} : 2^{|V|} \mapsto \mathbb{R}_{\geq 0}$, the solution S^* obtained by greedy algorithm (e.g., Algorithm §1) achieves at-least $1 - \frac{1}{e}$ of the optimal solution, i.e., $\mathcal{U}(\bar{S}^*) = (1 - \frac{1}{e}) \cdot \max_{|S| \leq m} \mathcal{U}(\bar{S})$.

This means that the greedy solution is no worse than $1 - \frac{1}{e} \approx 0.63$ of the optimal solution under *cardinality* constraint. Lin and Bilmes [2010] showed this approximation still holds with high probability even for non-monotone submodular objectives and with *budget* constraints, and use them in different document summarization objectives. For budget constraint, Lin and Bilmes [2010] proposed dividing the discrete derivative in greedy algorithm by $c(s)^r$, where r is a hyper-parameter, a scaling factor that trades-off cost and benefit of the item s in greedy step. For a detailed overview on *submodular-monotone* functions and their maximization, we refer readers to Krause and Golovin [2014].

Applying greedy maximization : We now see how greedy algorithm can be applied to the objectives such as in §2.7.2 for summarizing document collections. Recall from §2.7.4 that the central to greedy maximization algorithm (Algorithm 1) is the iterative prototype selection, where we iteratively add a single prototype that maximizes the marginal gain (or discrete derivative) as defined in Definition 2.12. We can compute the discrete derivative using the Definition 2.12 which evaluates the utility set functions $\mathcal{U}(\bar{S} + s)$ and $\mathcal{U}(\bar{S})$. But, finding the analytic expressions, offers computational advantages. For the example, we now show the discrete derivatives of FacLoc and LogDet.

Example 2.1 (Discrete derivative of FacLoc)**.** Recall that the facility location utility function is $\mathcal{U}_{FL}(\bar{S}) = \sum_{v \in V} \max_{s \in \bar{S}} k_{vs}$. The discrete derivative is:

$$\begin{aligned}\Delta_{\mathcal{U}_{FL}(s|\bar{S})} &= \sum_{v \in V} (\max\nolimits_{s \in \bar{S}+s} k_{vs} - \max\nolimits_{s \in \bar{S}} k_{vs}) \\ &= \sum_{v \in V} (\max(r_v, k_{vs}) - r_v)\end{aligned}$$

where we track the variables $r_v = \max_{s \in \bar{S}} k_{vs}$ in each stage of iterative selection.

Example 2.2 (Discrete derivative of LogDet)**.** Recall that the log determinant $\mathcal{U}_{ldet}(\bar{S}) = \log\det \mathbf{K}_{\bar{S},\bar{S}}$. The discrete derivative is:

$$\begin{aligned}\Delta_{\mathcal{U}_{ldet}(s|\bar{S})} &= \log\det \mathbf{K}_{\bar{S}+s,\bar{S}+s} - \log\det \mathbf{K}_{\bar{S},\bar{S}} \\ &= \log\left(\frac{\det \mathbf{K}_{\bar{S}+s,\bar{S}+s}}{\det \mathbf{K}_{\bar{S},\bar{S}}}\right) \\ &= \log(k_{s,s} - \mathbf{K}_{s,\bar{S}}\mathbf{K}_{\bar{S},\bar{S}}^{-1}\mathbf{K}_{\bar{S},s})\end{aligned}$$

where the last step follows from Schur's determinant identity [Zhang, 2006].

In Chapter 4, we show the discrete derivative for $\mathcal{U}_{mmd}$ (Equation 4.5). Once we have the expressions for discrete derivatives, we can employ the greedy algorithm to select prototypes.

2.8 Summary

In this chapter, we first reviewed the literatures in document summarization, and comparing document collections. We also review different types of document summarization problem, different datasets and evaluations. We then identify where our comparative summarization problem lies, and the datasets and evaluations applicable to our problem setting. Second, we review some background theory in MMD and submoduar functions. We use this theory to build the objectives for comparative summarization in Chapter 4 and Chapter 5, along with the optimization strategies of the objectives.

Chapter 3

Datasets on Controversial News Topics

> *You can have data without information, but you cannot have information without data.*
>
> Daniel Keys Moran,

In this chapter, we describe the data collection methodology and the properties of the news datasets we curated. The datasets help us in answering the research question we identified in Chapter 1. The dataset we collected consists of tweets, news articles and images on several controversial topics between July-2017 and Sep-2019. From this larger dataset, we curate two datasets of news articles – ControvNews2017 and News2019+Bias and made them publicly available. The first news dataset, ControvNews2017 is on controversial news topics, which we used to evaluate the unsupervised comparative summarization methods on the task of comparatively summarizing over time in Chapter 4. The second dataset, News2019+Bias is also on controversial news topics, but additionally provides ideological bias labels of the media outlets, enabling us to quantify and study the applicability of comparative summarization method in Chapter 6. Part of this work is published in Bista et al. [2019].

3.1 Introduction

Controversial news topics attract opposing viewpoints and stances from politicians, the media and the public and are of social significance. Examples of such topics include *gun control in US*, *addressing climate change*, *lockdowns in COVID-19 global pandemic*. There are much discussion about them in media outlets and social media on an ongoing basis, which leads to the need to summarize the high-volume content stream. Recent work on controversial topics [Garimella et al., 2018; Smith et al., 2013] focused on the social network and interaction around controversial topics, but did not explicitly consider the content of news articles on these topics.

To this end, we curate a set of news articles on long-running controversial topics using tweets which link to news articles. We initially wanted to collect the news articles on controversial topics, so that we can compare over time across ideological leanings of the media outlets. We choose several long-running controversial topics with significant news coverage between 2017 and 2019, such as *Climate change*, *Gun control*, etc. To find articles relevant to these topics we use keywords to filter the Twitter stream, and adopt a snowball strategy to add additional keywords [Verkamp and Gupta, 2013]. The articles linked in these tweets are then de-duplicated and filtered for spam. Article timestamps correspond to the creation time of the first tweet linking to it. We also download the images embedded in news articles and tweets. Additionally, we annotate the articles and tweets with knowledge-base concepts using DBPedia Spotlight [Mendes et al., 2011]. We design a scalable and fault-tolerant data collection architecture and store the news articles and tweets in Elasticsearch, a searchable store. The architecture may be of interest to researchers and engineers who want to develop a similar continuous and large scale data collection system.

From this larger datasets, we curate two publicly available datasets of news articles in which we apply comparative summarization. The dataset we curate is a subset of the larger dataset we collected, with additional cleaning and processing being done to address noise and spams in larger dataset. Exploring the evolution of news in controversial topics such as *climate change* is a natural application of comparative summarization. Hence, we first curate ControvNews2017 dataset of news articles from three news topics, which we split over time so that they can be used for evaluating comparative summarization over time methods such as those we develop in Chapter 4. An equally interesting problem is comparing news articles across ideological leanings of media outlets, such as comparing coverage on climate change between *left* and *right-leaning* media outlets. Second, we curate a News2019+Bias dataset of five news topics over nine months with groups defined by publication month and ideological leanings of media outlets (see §6.4). Such dataset help us to study the different ways in which news collections can be compared, as we see in Chapter 6. Overall, with these datasets, comparative summarization could help to better understand the role of media in such evolving news topics, and coverage of news across the political spectrum. To the best of our knowledge, such a dataset was lacking in the literature, and may be of interest to researchers who want to compare different document groups using different methods of comparisons as detailed in §2.5.

The rest of the chapter is outlined as follows. In §3.2, we first describe the dataset collection methodology. In particular, we first describe our strategy of topic and query curation for fetching data from twitter. We further describe the continuous and fault-resilient data collection and processing architecture. Second, in §3.3, we describe a subset of data that we use in Chapter 4, i.e., ControvNews2017 dataset. The dataset helps us to comparatively summarize the controversial news topics over time and evaluate our unsupervised algorithms

Topic	Queries	#Tweets	#WebPages
Beef Ban	beef ban, beefban	300k	5k
Capital Punish	death penalty, deathpenalty, capital punishment	23m	196k
Gun Control	gun control, guncontrol, gunsense, gunsafety, gun laws, gun violence	60m	667k
Climate change	climate change, climatechange	61m	1.7m
Refugees	refugee, refugees	46m	1m
Illegal immigration	illegal immigration, asylum, illegal immigrant	45m	606k
AI	artificial Intelligence, ArtificialIntelligence, AI, SelfDrivingCar Robotics, MachineLearning, ML, DeepLearning machine learning, Data Science	14m	988k
LGBT	gay marriage, LGBT, gay rights, LGBTQ, transpeople	70m	1.2m
AusPol	AusVotes19, auspol2019, ausvotes, auspol, AusPol QandA, nswpol, vicpol, qldpol,aabill,stopadani,kidsoffnauru	13m	169k

Table 3.1: Controversial Topic Dataset Statistics (July 2017-Sep 2019). It is to be noted that more tweets does not necessarily correspond to more web pages (and vice-versa). A fraction of the web pages we collect are actually the news articles.

(research question 1 (RQ1)). Finally, in §3.4, we describe how we annotate the news articles with ideological leanings of the media outlets to create News2019+Bias dataset, which we use in measurements studies presented in Chapter 6, thus answering the research question 4 (RQ4).

3.2 Data collection methodology

Here we describe the data collection methodology including the topic and query curation strategy, and the continuous data collection pipeline.

3.2.1 Topic curation

We curate an initial list of 10 topics in June 2017 that are satisfying the criteria of having non-trivial news coverage and being controversial. The topics are *Beef Ban*, *Capital Punishment Gun Control*, *Climate change, Illegal immigration, Refugees, Gay marriage, Animal testing, Cyclists on road* and *Marijuana*. We later removed *Animal testing*, *Marijuana* and *Cyclists on road* as these topics contained a significant portion of non articles within the web pages, such as link to cycling products, and cosmetic products. And we added *Auspol* (Australian politics) in late 2018, and *Artificial intelligence* in early 2019.

We focus on controversial topics since they are likely to be discussed over a long time period, and are also likely to attract multiple viewpoints in news coverage due to differing ideological leanings of media outlets. Controversy might also attract differing viewpoints according to the geography of content produced [AlAfnan, 2020]. Controversy is an important topic for research in social media and online political discourse, and is also important in real-

world applications such as intelligence and business strategy development.

To obtain various opinions on contemporary social problems, we choose Twitter as a source since it is frequently used for reporting and sharing related news articles. Garimella et al. [2018] use a similar approach for collecting a Twitter dataset on controversial topics. The authors consider Twitter hashtags as queries and a similarity function to retrieve similar hashtags. In contrast, we obtain embedded news articles from Twitter posts to collect a dataset and use a different expansion approach to retrieve related keywords [Verkamp and Gupta, 2013].

3.2.2 Query curation and articles extraction

Query curation. We use a hashtags expansion approach [Verkamp and Gupta, 2013] to curate relevant queries for each topic. We first manually select a single query for each topic, then use it to collect Twitter posts for two weeks. These posts are used as an initial data set that we create a query set based on. We extract the 10 most common hashtags that appear in the initial dataset. These hashtags are used to query the same dataset again, and then we re-extract the 10 most common hashtags from the query result. We continue this iteration several times until the hashtags used for query and the re-extracted hashtags are the same. All of the topics finish generating a query set after 4~5 iterations.

Based on the query set generated using the hashtags expansion, we perform additional manual filtering. Location hashtags such as #Florida or #Alabama are removed to prevent detailed locations being discussed. Some hashtags like #cow, #beef, #PJNET, and #2A are excluded since they are not directly related to the topics or are too general. As a result, *Beef Ban* topic is defined by just two query terms, while *Capital Punishment* and *Gun Control* include more diverse hashtags in the query set. Table 3.1 summaries the query set used for each topic.

Article extraction. After generating a query set for each topic, we fetch the Twitter stream that includes any of the hashtags in the query set. Twitter posts frequently include embedded news articles related to the post. We focus on the news articles in this work since they generally include more coherent stories than the corresponding Twitter post dataset. We extract the embedded news articles by visiting the article URL and downloading the content from it using newspaper package[1]. Here, we also do basic filtering to remove non-articles such as checking for non-empty title and content. In doing so, we can collect news articles that are mentioned and shared in ongoing social media which can be a measure of how important and engaging the news is.

[1] https://newspaper.readthedocs.io/en/latest/

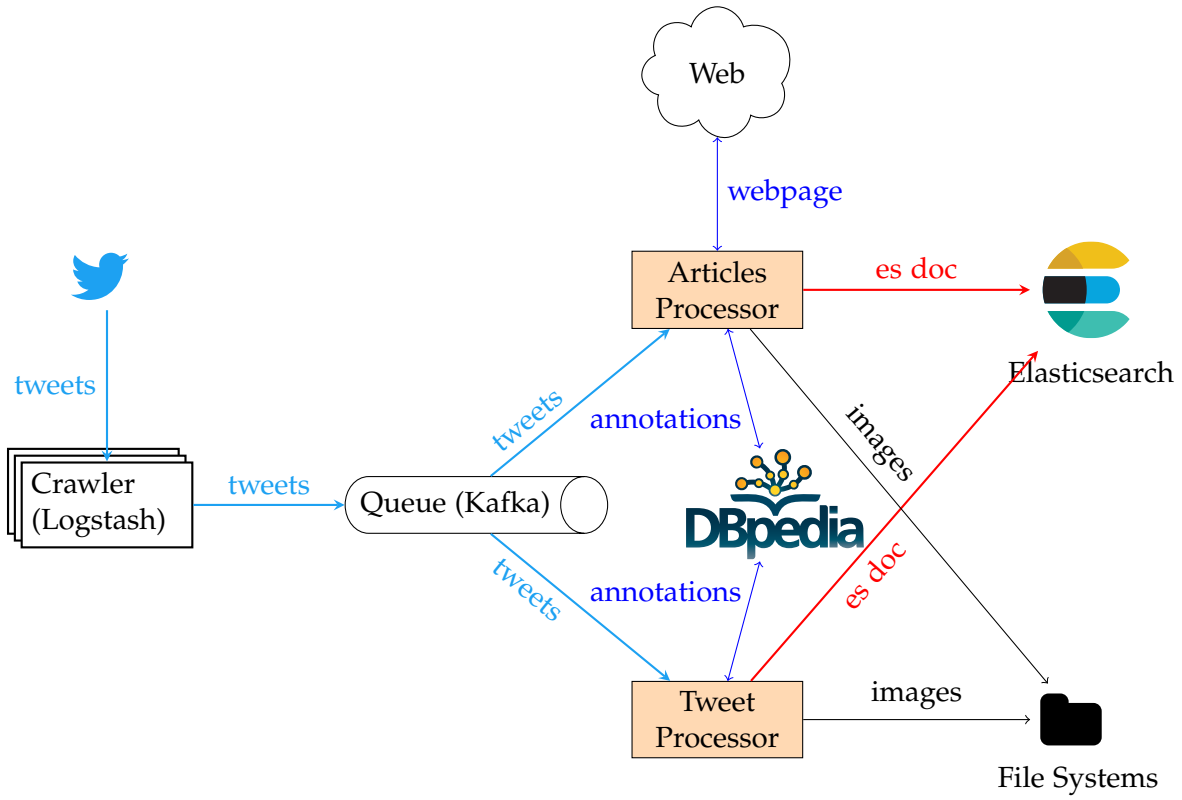


Figure 3.1: Data collection and preprocessing pipeline.

3.2.3 Data collection and pre-processing pipeline

We design a scalable, continuous and fault-tolerant system for collecting the datasets. We seek to collect the tweets, news articles mentioned in the tweets, images within the tweets and the articles, and annotate the text (tweets and news articles) with knowledge based concepts to link the text to real world entities. The data we seek to collect operate in two layers. First, we seek to collect a continuous stream of tweets from the Twitter. Second, we want to pre-process tweets to download the embedded images and web pages (potentially articles), and annotate the text with knowledge based entities. We then require the processor to index annotated tweets and articles in a search store that facilitates ease of retrieval and analysis of the text data. The processor also saves the images in the file system, and stores the meta-data and feature representations using the inception model [Szegedy et al., 2017] to the search store.

The overall data collection and pre-processing system is illustrated in Fig 3.1. We use the Logstash tool[2] to collect data from the Twitter streaming API[3]. The collection and pre-processing layers operate at different speeds due to the nature of processing and web access

[2]https://www.elastic.co/logstash
[3]https://developer.twitter.com/en/docs/tutorials/consuming-streaming-data

required. Additionally, we want our system be fault-tolerant to failures in either of the two layers. Hence, the two layers need to be decoupled from each other. We use Apache Kafka[4] to decouple the collection and processing layers [Kreps et al., 2011]. Kafka is a distributed, replicated and persistent queue which is often used to decouple the different parts of big data systems and in fault-tolerant streaming systems [Kleppmann, 2017]. Additionally, since Kafka is a persistent queue, a failure in downstream processing systems would not empty the queue, so we could just restart the processor from the last processed position.

We use DBPedia spotlight [Mendes et al., 2011] to annotate the text (tweets and articles) with the DBPedia (a knowledge base) concepts. DBPedia concepts can be thought of as an additional modality in our datasets, and as we shall see in Chapter 5, using DBPedia concepts in addition to text features improves the performance of summarization methods. We use Elasticsearch to index the tweets and articles[5], allowing us to perform a full-text search, and do various exploratory analysis. Elasticsearch is a full-text distributed and fault-tolerant search engine, which allows us to store, search, and analyze the documents [Gormley and Tong, 2015]. The different components – crawlers (logstash), queue (kafka brokers), processors (python services), DBPedia spotlight services, and search stores (elasticsearch nodes) were deployed in different virtual machines[6] and are scalable.

3.2.4 Article cleaning

We clean the data in Elastisearch search engine by filtering spam articles and removing duplicate articles mentioned in multiple tweets. To increase the relevance, we remove non-informative texts such as "Subscribe to our channel", "Please sign up" or "All Rights Reserved" that repeatedly appear with the news content. We also remove the articles that have sentences fewer than certain threshold, number of words fewer than certain threshold, and title having less than three words.

3.3 ControvNews2017 dataset

In Chapter 4, we use the ControvNews2017 dataset of news articles on three topics that appeared in a 14-month period (June 2017 – July 2018). We make this dataset publicly available[7]. Within each topic we comparatively summarize news articles in different time periods to identify what has changed in that topic between the summarization periods. To ensure our method works on a range of topics we chose substantially different long-running topics. First topic is *Beef Ban*, which is about the controversy over the slaughter and sale of beef on

[4] https://kafka.apache.org
[5] https://www.elastic.co/elasticsearch/
[6] https://nectar.org.au/research-cloud/
[7] https://github.com/computationalmedia/compsumm

religious grounds (1543 articles) is localized to a particular region, mainly the Indian subcontinent. The remaining topics are *Gun Control* – restrictions on carrying, using, or purchasing firearms (6494 articles) and *Capital Punishment* – use of the death penalty (7905 articles) are topical in various regions around the world. Figure 3.2 shows the number of new articles on each topic over time. Such a dataset allows us to compare news articles over time, and we use this dataset to evaluate the unsupervised comparative summarization methods we develop in Chapter 4.

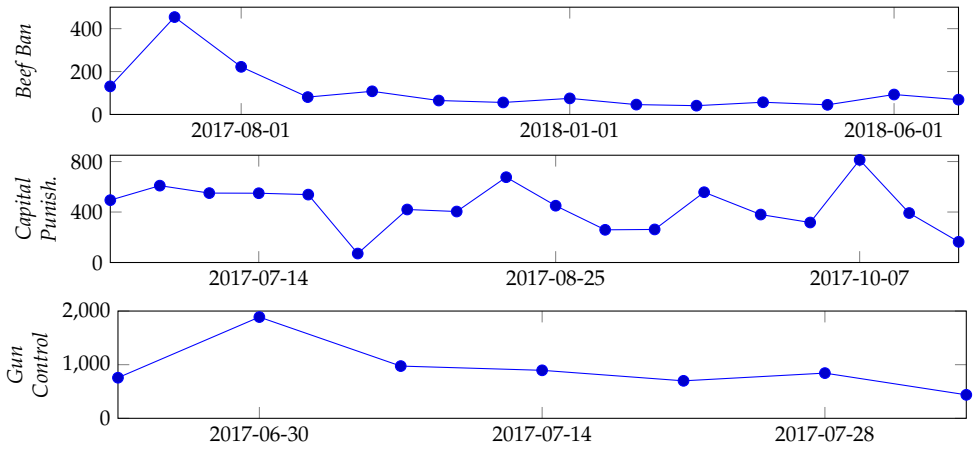


Figure 3.2: Data volume over time for each topic for ControvNews2017 dataset (§3.3).

3.4 News2019+Bias dataset

We can use comparative summarization to compare news articles in ways other than over time. We now describe a new News2019+Bias dataset that we curated, which enable us to perform different kinds of comparisons, and has a larger set of topics with diverse dynamics. We also desire to have newer data and leverage the potential of the continuous data collection system presented just before. We make this dataset publicly available[8]. The dataset consists of articles from 5 controversial topics: *Climate change*, *Gun control*, *Refugees*, *Illegal immigrants*, and *LGBT* in the period January 2019 to Sep 2019. We obtain the ideological leaning labels for the media outlets from the media-bias-fact-check [Zandt, 2015][9]. Then, we annotate the news articles from the media outlets present in the media-bias-fact-check with the ideological leanings of the outlets that produce the articles. This provides us with a document collection of 130K+ articles over 5 topics, and with two set of labels – *publication month* and *ideological leaning label*, allowing us to measure and study the *distinguishability* and *summarizability* over

[8]https://github.com/bistaumanga/compsumm-applicability
[9]https://mediabiasfactcheck.com

different groups of document collections in Chapter 6.

3.4.1 Ideology of media outlets

In order to annotate the articles with the political leaning labels, we first need to know the political leaning labels of the media outlets that produce the articles. The Media-bias-fact-check website [Zandt, 2015] provides the ratings on *ideological bias* and *factual reporting* for over 2300 media outlets. Annotating the ideological leaning of the articles as the ideological leaning of the media outlet is an example of distant supervision [Mintz et al., 2009]. Labels obtained from the Media-bias-fact-check website have been used to identify the leaning of media outlets [Patricia Aires et al., 2019; Stefanov et al., 2020].

Ideology/Factual	**RC**	**R**	**LC**	**L**	**ER**	**EL**	**C**	**Total**
	42	41	75	63	80	10	91	402
High	151	16	384	124			306	981
Low	1	18	1		151	7		178
Mixed	50	229	65	108	34	8	7	501
MostlyFactual	41	22	28	42			24	157
Veryhigh	1		9				42	52
VeryLow	1	6			27	2		36
Total	287	332	562	337	292	27	470	2307

Table 3.2: Summary of *Ideological leanings* and *Factuality* of the media outlets from *media-bias-fact-check* [Zandt, 2015] website.

We scraped bias labels and factual labels from images and text embedded inside the description page of each media outlet. An example of description page is shown in Fig 3.3. Amongst the various kind of information present in the description, we scraped the ideological leaning bias label and factual level for each media-outlet. The ideological leaning bias labels are: Extreme right (ER), Right (R), Right center (RC), Center (C), Left center (LC), Left (L), and Extreme left (EL). The factual labels are: High, Low, Mixed, Mostly factual, Very high and Very low. In some cases, the bias label and factual label were absent, in which we fall back to the navigation page name for bias label. We manually fixed the url of some websites, due to missing url or parsing errors. Altogether, we obtain ideology labels for 2307 websites, among which 1905 have factual labels, an additional 31 websites have missing ideology labels. We present the cross table counts of outlets with different ideology labels and factual labels in Table 3.2. The ideological and factual labels of the all media outlets is available online[10].

An observation in Table 3.2 is that *Center, Left center*, and *Right center* leaning outlets are likely to be highly factual, whereas *Extreme right* leaning outlets are likely to have low factual

[10]https://www.bistaumanga.com.np/files/media_bias_factuality_oct2020_mbfc.csv

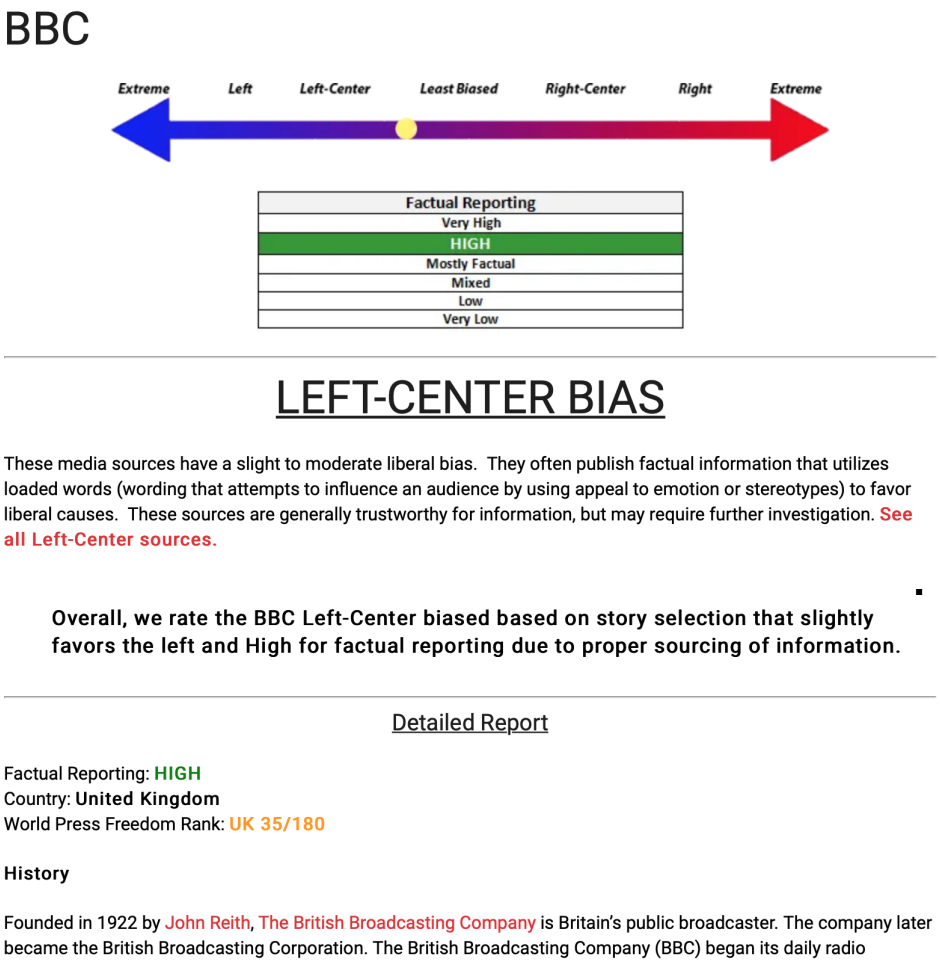


Figure 3.3: An example of description in Media-bias-fact-check website for BBC.

correctness.

3.4.2 Annotating the ideological leanings of the news articles

First, we take a subset of controversial news datasets between January 2019 and September 2019 from 5 topics – *Climate change*, *Gun control*, *Refugees*, *Illegal immigrants*, and *LGBT*. We then obtain the media outlet that produced them from the url by extracting the top-level-domain using *tldextract* package[11].

Finally, we annotate the articles with the ideological leaning labels of the media outlets that produced them, if the top-level-domain of the article exists in the media-bias labels that we created in §3.4.1. We then ignore the articles from outlets that have less than 10 articles in each topic, and ignore the articles that have less than three sentences and whose titles have

[11] https://pypi.org/project/tldextract/

topic	EL	L	LC	C	RC	R	ER	filtered	mbfc	articles
LGBT	19	9322	10422	2930	1940	2643	973	28249	33923	130694
IllegalImmi	9	2132	7667	3304	2111	5047	2865	23135	28067	69103
Refugees	10	1853	8065	3170	2080	2180	1216	18574	22813	89594
GunControl	17	2363	6386	2894	1810	4479	1283	19232	23710	59590
ClimateChange	17	6350	21931	7918	4942	3945	1284	46387	56724	223523

Table 3.3: Summary of News2019+Bias dataset (§3.4).

less than three words. Finally, there are only a few articles from *Extreme left* outlets are very low, hence, we ignore these articles.

After completing these preprocessing and filtering steps, we have altogether 130K+ articles from 5 topics as shown in Table 3.3. We show the number of articles in each ideological leaning for each topic. We also show the number of articles obtained from Elasticsearch (column articles), number of articles whose outlet is present in media-bias-fact-check (column mbfc), and finally the articles not meeting the number of sentences and number of title words' threshold (column filtered). The dataset is highly imbalanced with *Extreme right* having the fewest number of articles and *Left center* having the highest number of articles in each topic. Next, we see how we can use the dataset in comparative summarization and other tasks.

3.4.3 Comparison settings with News2019+Bias dataset

We now briefly see how we can compare different groups in one of the dataset we curated. For this demonstration, we pick the *LGBT* topic, and provide the count breakdown of articles in each month and ideology. We filter out the articles from extreme left, as the counts were too low to facilitate any comparisons. The count breakdown of the *Climate change* topic is provided in Table 6.1, and other remaining topics are provided in Appendix A.3.

	ER	R	RC	C	LC	L	Total
2019-01	87	236	153	240	906	914	2536
2019-02	117	316	244	311	1181	974	3143
2019-03	111	291	225	322	1178	1057	3184
2019-04	86	309	207	331	1168	1037	3138
2019-05	113	296	224	364	1311	1073	3381
2019-06	165	464	380	662	2054	1517	5242
2019-07	116	307	204	301	1159	975	3062
2019-08	96	217	177	218	815	949	2472
2019-09	82	207	126	181	650	826	2072
Total	973	2643	1940	2930	10422	9322	28230

Table 3.4: Counts over months and across ideologies for LGBT topic within News2019+Bias dataset.

Suppose we are given a document collection with groups defined in multiple ways – *publication month*, and *ideological leaning* of media outlets as in Table 3.4. We can compare the articles

across ideologies such as Extreme right vs Center, or over time such as June vs September as demonstrated in the table. We could even compare ideologies within each month or months within an ideology. For example, in the table we highlight the potential candidate for comparisons – comparing Right vs Left from January, and comparing March vs May within the center aligned media outlets. In Chapter 4, we evaluate our unsupervised methods by comparing articles over time. Whereas in Chapter 6, we compare both ideologies within each month, and months within a same ideology. We compare articles across ideologies in each month because, we do not want the changing vocabulary (e.g. from emerging named entities) within each topic over different months to confound the comparison between ideologies. Similarly, in comparing over time we do not want the changing viewpoints within each topic across different ideologies to confound the comparison. A more detailed experimental setup is provided within each of those chapters.

3.5 Summary

In this chapter, we describe the data collection methodology, and the datasets we curated to assist us in evaluating the comparative summarization methods in Chapter 4, and studying the applicability of methods in Chapter 6. Two large scale datasets we curated, ControversialNews2017 and News2019+Bias datasets can be used to evaluate the systems comparing document collections. We also document the data scalable, continuous and fault-tolerant data collection architecture, along with the design choices we made in building this system, which may be interest to researchers looking to build similar systems.

Chapter 4

Comparative Document Summarization as Classification

> *Information is the oil of the 21st century, and analytics is the combustion engine.*
>
> Peter Sondergaard, *Gartner Research*

In this chapter, we present our work on unsupervised methods and evaluations for comparative summarization addressing research question 1 (RQ1) and research question 3 (RQ3) of the thesis. In Section 4.3.2, we provide a novel formulation of comparative summarization as a problem of competing Using the classifiers formulation, in Section 4.4.2 we propose novel objectives based on Maximum Mean Discrepancy (MMD) for unsupervised comparative summarization. In Section 4.5 we present the greedy and gradient based optimization strategies for the objectives we proposed. In §4.6.1, we propose new automatic evaluation strategies for comparative summarization and conduct a pilot study to validate the proposed automatic evaluation. We then show the effectiveness of our methods in the task of comparatively summarizing the controversial news topics over time on a newly created ControvNews2017 dataset (§3.3). We make the software and datasets used in this chapter's experiments publicly available[1]. Part of this work is published in Bista et al. [2019] and Bista [2019].

4.1 Introduction

In this chapter, we consider unsupervised *comparative summarization*. Comparative summarization is extractive summarization in the comparative setting (§1.1). Given *groups* of document collections, the aim of comparative summarization is to select documents that represent each group, but also highlight differences *between* groups. This is in contrast to generic document summarization which aims to represent each group by independently optimizing for

[1] https://github.com/computationalmedia/compsumm

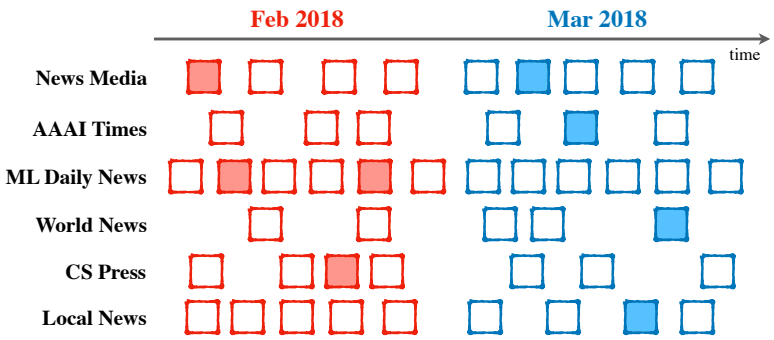


Figure 4.1: An illustrative example of comparative summarization. Squares are news articles, rows denote different news outlets, and the x-axis denotes time. The shaded articles are chosen to represent AI-related news during Feb and March 2018, respectively. They aim to summarize topics in each month, and also highlight differences *between* the two months.

information coverage and diversity, without considering other groups. As a concrete example, given thousands of news articles per month on a certain topic, groups can be formed by publication time, by source, or by political leaning. Comparative summarization systems can then help answer user questions such as: what is new on the topic of climate change this week, what is different between the coverage in NYTimes and BBC, or what are the key articles covering the carbon tax and the Paris Agreement? In this work, we focus on highlighting changes within a long-running news topic over time; see Figure 4.1 for an illustration.

Existing methods for extractive summarization use a variety of formulations such as structured prediction [Li et al., 2009a], optimization of submodular functions [Lin and Bilmes, 2011], dataset interpretability [Kim et al., 2016], and dataset selection via submodular optimization [Mirzasoleiman et al., 2016; Wei et al., 2015; Mitrovic et al., 2018]. Moreover, recent formulations of comparative summarization use discriminative sentence selection [Wang et al., 2012; Li et al., 2012a], or highlight differences in common concepts across documents [Huang et al., 2011]. However, the connections and distinctions between these approaches has yet to be clearly articulated. To evaluate summaries, traditional approaches employ automatic metrics such as ROUGE [Lin, 2004] on manually constructed summaries [Lin and Hovy, 2003; Nenkova et al., 2007]. This is difficult to employ for new tasks and new datasets, and does not scale.

Our approach to comparative summarization is based on a novel formulation of the problem in terms of two *competing classification tasks*. Specifically, we formulate the problem as finding summaries for each group such that a powerful classifier can distinguish them from documents belonging to *other* groups, but cannot distinguish them from documents belonging to the *same* group. We show how this framework encompasses an existing nearest neighbor objective for summarization, and propose two new *submodular-monotone* objectives based on the maximum mean discrepancy [Gretton et al., 2012a] – *mmd-diff* which emphasizes classification

accuracy and *mmd-div* which emphasizes summary diversity – as well as new gradient-based optimization strategies for these objectives.

A key advantage of our discriminative problem setting is that it allows summarization to be evaluated as a classification task. To this end, we design extrinsic automatic and crowd-sourced evaluations for comparative summaries, which we apply on the ControvNews2017 dataset of three ongoing controversial news topics that we introduced in §3.3. We observe that the new objectives with gradient optimization are top-performing in 14 out of 24 settings (across news topics, summary size, and classifiers) (§4.6.4). We design a new crowd-sourced article classification task for human evaluation. We find that workers are on average 7% more accurate in classifying articles using summaries generated by *mmd-diff* with gradient-based optimization than all alternatives. Interestingly, our results contrast with the body of work on dataset selection and summarization that favor discrete greedy optimization of submodular objectives due to approximation guarantees. We hypothesize that the comparative summarization problem is particularly amenable to gradient-based optimization due to the small number of prototypes needed. Moreover, gradient-based approaches can further improve solutions found by greedy approaches.

In summary, the main contributions of this chapter are:

- A new formulation of comparative document summarization in terms of competing binary classifiers (§4.3.2), two new objectives based on this formulation (§4.4.2), and their corresponding greedy (§4.5.1) and gradient-based optimization strategies (§4.5.2).
- The design of a scalable automatic (§4.6.3) and human evaluation (§4.6.5) methodology for comparative summarization models, with results showing that the new objectives out-perform existing methods.

4.2 Related works

The broader context of this work is extractive summarization. Approaches to this problem include incorporating diversity measures from information-retrieval [Carbonell and Goldstein, 1998], structured SVM regularized by constraints for diversity, coverage, and balance [Li et al., 2009a], or topic models for summarization [Haghighi and Vanderwende, 2009]. Time-aware summarization is an emerging subproblem, where the current focus is on modeling continuity [Ren et al., 2016] or continuously updating summaries [Rücklé and Gurevych, 2017], rather than formulating comparisons. Li et al. [2012a]; Wang et al. [2012] present methods to extract one or few discriminative sentences from a small multi-document corpus utilizing greedy optimization and evaluating qualitatively. Huang et al. [2011] compares descriptions about similar concepts in closely related document pairs, leveraging an integer linear program and evaluating with few manually created ground truth summaries. While these works exist in

the domain of comparative summarization, they are either specific to a data domain or have evaluations which are hard to scale up. In this chapter we present approaches to comparative summarization with intuition from competing binary classifiers, leading to different objectives and evaluation. We demonstrate and evaluate the application of these approaches to multiple data domains such as images and text.

Submodular functions have been the preferred form of discrete objectives for summarizing text [Lin and Bilmes, 2011], images [Simon et al., 2007] and data subset selection [Wei et al., 2015; Mitrovic et al., 2018], since they can be optimized greedily with tightly-bounded guarantees. The topic of interpreting datasets and models use similar strategies [Kim et al., 2016; Bien and Tibshirani, 2011]. This work re-investigates classic continuous optimization for comparative summarization, and puts it back on the map as a competitive strategy.

4.3 Comparative summarization as classification

We first formally describe the problem of comparative summarization in the extractive setting, and then cast it as competing binary classifiers.

4.3.1 Problem statement

Formally, the comparative summarization problem is defined on G groups of document collections $\{\mathbf{X}_1, \ldots, \mathbf{X}_G\}$, where a group may, for example, correspond to news articles about a specific topic published in a certain month. We write the document collection for group g as

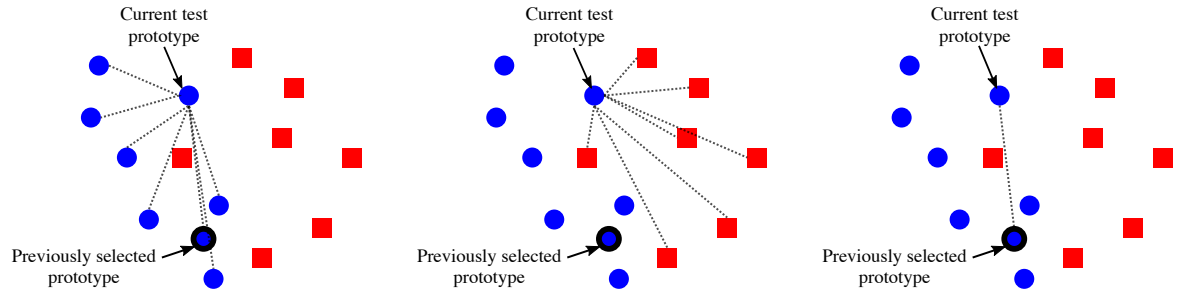
$$\mathbf{X}_g = \{\mathbf{x}_{g,1}, \mathbf{x}_{g,2}, \ldots, \mathbf{x}_{g,N_g}\},$$

where N_g is the total number of documents in group g. We represent individual documents as vector $\mathbf{x}_{g,i} \in \mathbb{R}^d$ (see §4.6).

Our goal is to summarize each document collection $\mathbf{X}_g$ with a set of *summary documents* or *prototypes* $\bar{\mathbf{X}}_g \subset \mathbf{X}_g$, written

$$\bar{\mathbf{X}}_g = \{\bar{\mathbf{x}}_{g,1}, \bar{\mathbf{x}}_{g,2}, \ldots, \bar{\mathbf{x}}_{g,M}\}.$$

For simplicity, we assume the number of prototypes $M = M_g$ is the same for each group. The selected prototypes should represent the documents in the group achieving coverage (Figure 4.2a) and diversity (Figure 4.2c), while simultaneously discriminating documents from other groups (Figure 4.2b). For example, if we have news articles on the *Climate Change* topic, then they may discuss the *Paris agreement* in February, *coral bleaching* in March, and *rising sea levels* in both months. A comparative summary should include documents about the *Paris agreement* in February and *coral bleaching* in March, but avoid documents on *rising sea levels* as they are common to both time ranges and hence do not discriminate. It is common to summa-



(a) Coverage is the average similarity between the test prototype document and all other documents in the test documents' group.

(b) Discriminativeness is the average dissimilarity between the test prototype document and all documents **not** in the test document's group.

(c) Diversity is the average dissimilarity between the test prototype document and all selected prototypes in the test document's group.

Figure 4.2: Illustration of coverage, discriminativeness and diversity criteria for selecting prototypes. The two document groups are shown as blue circles and red squares. The dotted lines represent comparisons between pairs of documents.

rize a group of documents by picking representative sentences [Erkan and Radev, 2004; Lin and Bilmes, 2011; Nallapati et al., 2017; Yasunaga et al., 2017]. Selecting an entire document from a corpus has also been used in information retrieval [Gelbukh et al., 2003; Raiber and Kurland, 2010], and summarizing image collections [Xu et al., 2011; Tschiatschek et al., 2014]. We posit that a system to pick representative documents from a corpus has interesting use cases in text summarization, such as summarizing evolving topics as mentioned above, and summarizing bias in news and social-media which we present in Chapter 6.

4.3.2 A Binary classification perspective

We now cast comparative summarization as a binary classification problem. To do so, let us re-interpret the two defining characteristics of prototypes $\bar{\mathbf{X}}_g$ for the gth group:

(i) they must represent the documents belonging to that group. Intuitively, this means that each $\bar{\mathbf{x}}_{g,i} \in \bar{\mathbf{X}}_g$ must be *indistinguishable* from all $\mathbf{x}_{g,j} \in \mathbf{X}_g$.

(ii) they must discriminate against documents from all other groups. Intuitively, this means that each $\bar{\mathbf{x}}_{g,i} \in \bar{\mathbf{X}}_g$ must be *distinguishable* from $\mathbf{x}_{\neg g,j} \in \mathbf{X}_{\neg g}$, where $\mathbf{X}_{\neg g}$ denotes the set of all documents belonging to all groups except g.

This lets us relate prototype selection to the familiar binary classification problem: for a good set of prototypes,

(a) there *cannot exist* a classifier that can accurately discriminate between them and documents from that group. For example, even a powerful classifier should not be able to discriminate

prototype documents about the Great Barrier Reef from other documents about the Great Barrier Reef.

(b) there *must exist* a classifier that can accurately discriminate them against documents from all other groups. For example, a reasonable classifier should be able to discriminate prototypes about the Great Barrier Reef from documents about emission targets.

Consequently, we can think of prototype selection in terms of two competing binary classification objectives: one distinguishing $\bar{\mathbf{X}}_g$ from $\mathbf{X}_g$, and another distinguishing $\bar{\mathbf{X}}_g$ from $\mathbf{X}_{\neg g}$. In abstract, this suggests a multi-objective optimization problem of the form

$$\max_{\bar{\mathbf{X}}_g} \left(-\text{Acc}(\bar{\mathbf{X}}_g, \mathbf{X}_g),\ \text{Acc}(\bar{\mathbf{X}}_g, \mathbf{X}_{\neg g}) \right), \tag{4.1}$$

where $\text{Acc}(\mathbf{X}, \mathbf{Y})$ estimates the accuracy of the best possible classifier for distinguishing between the datasets $\mathbf{X}$ and $\mathbf{Y}$. Making this idea practical requires committing to a particular means of balancing the two competing objectives. More interestingly, one also needs to find a tractable way to estimate $\text{Acc}(\cdot, \cdot)$: explicitly searching over rich classifiers such as deep neural networks, would lead to a computationally challenging nested optimization problem.

In the following we discuss a set of objective functions that avoid such nested optimization. We also discuss two simple optimization strategies for these objectives in §4.5.

4.4 Unsupervised methods for comparative summarization

We now present our novel unsupervised models for comparative summarization addressing RQ1. First, we connect our classifier formulation with an existing method for summarization based on nearest neighbor [Wei et al., 2015]. Then, we define a couple of objectives based on Maximum Mean Discrepancy (MMD) (§2.6). Finally, we show how the objectives relate to some existing summarization strategies.

4.4.1 Prototype selection via nearest-neighbor

One existing prototype selection method involves approximating the intragroup $\text{Acc}(\cdot, \cdot)$ term in Equation 4.1 using nearest-neighbor classifiers, while ignoring the intergroup accuracy term. Specifically, a formulation of prototype selection in Wei et al. [2015] maximizes the total similarity of every point to its nearest prototype from the same group:

$$\mathcal{U}_{nn}(\bar{\mathbf{X}}_g) = \sum_{i=1}^{N_g} \max_{m \in \{1,\ldots,M\}} \text{Sim}(\bar{\mathbf{x}}_{g,m}, \mathbf{x}_{g,i}). \tag{4.2}$$

Here, Sim is any similarity function, with admissible choices including a negative distance, or *positive definite kernel* functions (§2.6.2).

The nearest-neighbor objective was articulated in Wei et al. [2015] and earlier in Bien and Tibshirani [2011], and used for classification tasks. It also appears in several summarization literatures as facility location method [Lin et al., 2009; Tschiatschek et al., 2014] (see §2.7.2 $\mathcal{U}_{FL}$). The nearest neighbor utility function is simple and intuitive. However, it only considers the most similar prototype for each datapoint which misses our second desirable property of prototypes: that they explicitly distinguish between different groups. Moreover, the nearest neighbor utility function can be challenging to optimize because of the max function. The rest of this section introduces three other utilities that address these concerns.

4.4.2 Prototype selection via MMD

We introduced MMD in §2.6.3 as a measure of distance between sets of datapoints, and reviewed how existing work has used it in prototype selection (generic summarization) in §2.6.4. One can think of MMD as implicitly computing a (kernelized) *nearest centroid classifier* to distinguish between $\mathbf{X}$ and $\mathbf{Y}$: MMD is small when this classifier has high expected error. Furthermore, for a characteristic kernel, MMD is equivalent to the difficulty in classifiability of a kernel based classifier as shown by Sriperumbudur et al. [2009a]. Thus, MMD can be seen as an efficient approximation to classification accuracy $\mathrm{Acc}(\cdot,\cdot)$. This intuition leads to a practical utility function that approximates Equation 4.1 by taking the difference of two MMD terms:

$$\mathcal{U}_{diff}(\bar{\mathbf{X}}_g) = -\mathrm{MMD}^2(\bar{\mathbf{X}}_g, \mathbf{X}_g) + \lambda \cdot \mathrm{MMD}^2(\bar{\mathbf{X}}_g, \mathbf{X}_{\neg g}). \tag{4.3}$$

The hyper-parameter λ with $0 \leq \lambda < 1$ trades off how well the prototype represents its group, against how well it distinguishes between groups (Figure 4.2b). Intuitively, when the term $\mathrm{MMD}^2(\bar{\mathbf{X}}_g, \mathbf{X}_{\neg g})$ is large then the prototypes $\bar{\mathbf{X}}_g$ are dissimilar from documents $\mathbf{X}_{\neg g}$ of other groups. Similarly, when $\mathrm{MMD}^2(\bar{\mathbf{X}}_g, \mathbf{X}_g)$ is small then the prototypes are similar to documents of that group. Maximizing $-\mathrm{MMD}^2$ gives prototypes that are both close to the empirical samples (as seen by the $\mathbb{E}_{\mathbf{x},\mathbf{y}}$ term in Equation (2.7) and illustrated by Figure 4.2a) and far from one another (as seen by the $\mathbb{E}_{\mathbf{y},\mathbf{y}'}$ term and illustrated by Figure 4.2c).

While the objective of Equation (4.3) provides the core of our approach, we also present a variant that increases the diversity of prototypes chosen for each group. A closer examination of the difference of MMD^2 in Equation (4.3) – by expanding both using Equation 2.8 – reveals two separate prototype diversity terms $-\mathbb{E}_{\bar{\mathbf{x}}_g,\bar{\mathbf{x}}'_g}[\mathrm{k}(\bar{\mathbf{x}}_g, \bar{\mathbf{x}}'_g)]$ and $\lambda\mathbb{E}_{\bar{\mathbf{x}}_g,\bar{\mathbf{x}}'_g}[\mathrm{k}(\bar{\mathbf{x}}_g, \bar{\mathbf{x}}'_g)]$. The latter counteracts the former and decreases prototype diversity. On the expanded form of $\lambda\mathrm{MMD}^2(\bar{\mathbf{X}}_g, \mathbf{X}_{\neg g})$, we remove the terms not involving $\bar{\mathbf{x}}_g$, as they are constants and have no effect on the solution, and also remove the conflicting diversity term $\lambda\mathbb{E}_{\bar{\mathbf{x}}_g,\bar{\mathbf{x}}'_g}[\mathrm{k}(\bar{\mathbf{x}}_g, \bar{\mathbf{x}}'_g)]$. This

gives a new objective:

$$\mathcal{U}_{div}(\bar{\mathbf{X}}_g) = -\text{MMD}^2(\bar{\mathbf{X}}_g, \mathbf{X}_g) - 2\lambda \mathbb{E}_{\bar{\mathbf{x}}_g, \mathbf{x}_{\neg g}}[\text{k}(\bar{\mathbf{x}}_g, \mathbf{x}_{\neg g})]. \tag{4.4}$$

Maximizing $-\lambda \mathbb{E}_{\bar{\mathbf{x}}_g, \mathbf{x}_{\neg g}}[\text{k}(\bar{\mathbf{x}}_g, \mathbf{x}_{\neg g})]$ encourages prototypes in group g to be far from data points in other groups.

One can envision another variant that explicitly optimizes the diversity between *prototypes of different groups*, rather than between prototypes of group g against data points in other groups. This is computationally more efficient, and reflects similar intuitions. However, it did not outperform $\mathcal{U}_{diff}$, $\mathcal{U}_{div}$ in summarization tasks. Next, we explain how $\mathcal{U}_{diff}$ and $\mathcal{U}_{div}$ relate to existing summarization methods.

4.4.3 Comparisons with related methods

MMD-critic: Kim et al. [2016] proposed *MMD-critic*, a two stage method for selecting prototypes. The first stage selects the prototypes $\bar{\mathbf{X}}$ for a single group of documents $\mathbf{X}$ by maximizing $-\text{MMD}^2(\bar{\mathbf{X}}_g, \mathbf{X}_g)$, i.e., $\mathcal{U}_{mmd}(\bar{\mathbf{X}}_g, \mathbf{X}_g)$ as we discussed in §2.6.4. The first term in Equation (4.3) builds on this formulation, applying this idea independently for each group. The second term in Equation (4.3) is crucial to encourage prototypes that *only* represent their own group and none of the other groups.

The second stage of *MMD-critic* contains the *model criticisms*, which have to be optimized sequentially after obtaining prototypes. As shown in §4.6, *MMD-critic* under-performs in comparison tasks by a significant margin. An additional discussion on the criticisms part of *MMD-critic* is provided in Appendix A.1.

Graph cuts: Graph cut is a well known submodular function that can be used for extractive summarization [Lin et al., 2009; Lin and Bilmes, 2010]. We introduced two variants Cut ($\mathcal{U}_{cut}$), and CutDiv ($\mathcal{U}_{mcut}$) in §2.7.2. Both of these objectives are comparable with the generic summarization part of MMD based objectives (Equation (2.10)). Equation (2.10) does not have the hyper-parameter η that controls the diversity of prototypes, but instead it automatically determines the diversity term weighting from the data.

Measuring innovativeness using classifiers: A similar classifier based approach was employed by Savov et al. [2020] to identify innovative papers, where they use classifier's confidence to find scientific papers that proposed novel ideas. If the classifier trained on scientific papers from various years predicts a given paper to be published several years later than this paper's actual publication date, then the paper can be deemed as an innovative (one that started certain topic or was first to propose effective solution which was later used widely by the community). Our method differs because, we identify the instances (or prototypes) that maximizes the classification objective (4.1), while Savov et al. [2020] train a classifier and use

confidence scores on an unseen paper to decide if it is innovative.

Random walk based methods: Methods based on random walk such as Lexrank, which is a classical pagerank based algorithm for generic summarization are common [Erkan and Radev, 2004]. However, such method for comparative summarization were lacking in the literature until recently. Additionally, performance of such method can be subject to the choice of graphs. In §5.5, we find that one of our classifier based mmd method performs on par with Lexrank on generic summarization task. A recent work by Rieskamp [2022] compares our classification based supervised MMD method (SupMMD method we introduce in §5.3.2) against the Lexrank based method modified for a related contrastive argument summarization task. In that work, author found our method is suitable for the task as it is better in representing the gist of the contrastive arguments.

4.5 Optimizing utility functions

There are two general strategies for optimizing the utility functions outlined in §4.4 to generate summaries that are a subset of the original dataset: greedy and gradient optimization. We now describe how we can apply these optimization strategies on our proposed objectives.

4.5.1 Greedy optimization

In optimizing the utility functions $\mathcal{U}_{diff}$ and $\mathcal{U}_{div}$, the first strategy involves directly choosing M prototypes for each group. Obtaining the exact solution to this discrete optimization problem is intractable; however, approximations such as greedy selection can work well in practice, and may also have theoretical guarantees as discussed in §2.7.4. Specifically, given a utility function $\mathcal{U}$, the greedy algorithm (Algorithm 1) works by iteratively picking the $\mathbf{x}_g$ that provides the largest discrete derivative or marginal gain ($\Delta_{\mathcal{U}}(\mathbf{x}_g|\bar{X}_g)$ as defined in Definition 2.12) one at a time for each group. First, we note the nearest-neighbour objective $\mathcal{U}_{nn}$ is *submodular-monotone* as shown by Wei et al. [2015], hence greedy algorithms provide solutions with guaranteed lower bounds. The Discrete derivative of $\mathcal{U}_{nn}$ in Equation (4.2) is same as that of facility location $\mathcal{U}_{FL}$ in 2.7.4. We now show the submodularity and monotonicity of $\mathcal{U}_{diff}$ and $\mathcal{U}_{div}$ using Theorem 2.5.

We use set notations in the proofs as in §2.7.3 for convenience. Let V_g be the set of N_g documents in group g (corresponding vector notation $\mathbf{X}_g$), and $V_{\neg g}$ be the set of $N_{\neg g}$ documents in other groups (vector notation $\mathbf{X}_{\neg g}$). Let $\bar{S}_g$ be the prototype summary we seek in summarizing group g against all other groups $\neg g$. Similar to the *submodularity* and *monotonicity* of $\mathcal{U}_{mmd}$ in Corollary 2.1, we first show $\mathcal{U}_{diff}(\bar{S})$ is a linear function of kernel matrix $\mathbf{K}$.

Lemma 4.1 (Linear forms of $\mathcal{U}_{diff}(\bar{S})$)**.** $\mathcal{U}_{diff}(\bar{S}_g)$ with $m = |\bar{S}_g|$ in Equation (4.3) is a linear

Algorithm 1 Greedy algorithm to maximize objective $\mathcal{U}(\cdot)$

Require: $\{\mathbf{X}_{1:G}\}$: Groups of documents,
G: number of groups, M: number of prototypes per group
procedure GREEDYMAX($\mathbf{X}, G, M$)
  $(\forall g \in 1 \dots G)\, \bar{\mathbf{X}}_g \leftarrow \{\}$
  for m from 1 to M **do**
    for g from 1 to G **do**
      $\bar{\mathbf{x}}_{g,m} \leftarrow \underset{\mathbf{x}_g \in \mathbf{X}_g \setminus \bar{\mathbf{X}}_g}{\text{Argmax}}\, \Delta_{\mathcal{U}}(\mathbf{x}_g | \bar{\mathbf{X}}_g)$
      $\bar{\mathbf{X}}_g \leftarrow \bar{\mathbf{X}}_g \cup \{\bar{\mathbf{x}}_{g,m}\}$
  $\bar{\mathbf{X}} \leftarrow \cup_{g=1}^{G} \mathbf{X}_g$
  return $\bar{\bar{X}}$

function of kernel matrix $\mathbf{K}$: $\mathcal{U}_{diff}(\bar{S}_g) = \langle \mathbf{A}(\bar{S}_g), \mathbf{K} \rangle$, where,

$$A_{ij}(\bar{S}_g) = \frac{2}{N_g \cdot m} \mathbb{1}_{[i \in V_g][j \in \bar{S}_g]} - \frac{1-\lambda}{m^2} \mathbb{1}_{[i \in \bar{S}_g]} \mathbb{1}_{[j \in \bar{S}_g]} - \frac{-2\lambda}{N_{\neg g} \cdot m} \mathbb{1}_{[i \in V_{\neg g}][j \in \bar{S}_g]}.$$

Similarly, we can show the linear form of $\mathcal{U}_{div}$ as well. We now apply the Theorem 2.5 to this linear form to get the upper bounds on kernel matrix for $\mathcal{U}_{diff}(\bar{S}_g)$.

Corollary 4.1 (Submodularity and monotonicity of $\mathcal{U}_{diff}$)**.** For a kernel matrix that is non-negative, has equal diagonal terms $k_{ii} = k_*$. If the off-diagonal terms satisfy $k_{ij} \leq \frac{k_*}{(M+1)(M^2+3M+1)}$, $\mathcal{U}_{diff}$ is *submodular-monotone*.

Proof. The proof is similar to the proof of Corollary 2.1, we apply Lemma 4.1 in Theorem 2.5. For a kernel matrix that satisfies the stated conditions, we have $\mathbf{E} = \mathbf{I}$. We can show $a(\bar{S}_g) = \epsilon - \frac{1-\lambda}{m}$ and $b(\bar{S}) = (1-\lambda)\frac{m+1}{m} - \epsilon$, where $\epsilon = \frac{2}{N_g}$. When we apply these to Theorem 2.5 and for $0 \leq \lambda < 1$, we get $\alpha(N, m) = \frac{1-\lambda}{(m+1)((1-\lambda)(m+1)-\epsilon m)} \geq \frac{1}{(m+1)^2}$, and $\beta(N, m) = \frac{1}{(m+1)(m^2+3m+1-\epsilon m(m+2))} > \frac{1}{(m+1)(m^2+3m+1)}$. Since, $(M^2 + 3M + 1) \geq (M+1) \forall_{M>0}$ a combined upper bound on the kernel entries $k_{ij} \leq \frac{k_*}{(M+1)(M^2+3M+1)}$ is sufficient for both *monotonicity* and *submodularity*. ■

Corollary 4.2 (Submodularity and monotonicity of $\mathcal{U}_{div}$)**.** For a kernel matrix that is non-negative, has equal diagonal terms $k_{ii} = k_*$. If the off-diagonal terms satisfy $k_{ij} \leq \frac{k_*}{(M+1)(M^2+3M+1)}$, $\mathcal{U}_{div}$ is *submodular-monotone*.

Proof. The proof of Corollary 4.2 is similar to the proof of Corollary 4.1. ■

Since our objectives $\mathcal{U}_{diff}$ and $\mathcal{U}_{div}$ are *submodular-monotone*, we can apply greedy algorithms to get solutions with bounded guarantees. Recall from §2.7.4 that pre-computing an-

alytic discrete derivative of the objectives have computational advantages in discrete maximization, we now compute the discrete derivative of our MMD based objectives.

Discrete Derivatives: Let $|\bar{\mathbf{X}}_g| = m$, then the discrete derivatives of $\mathcal{U}_{mmd}(\bar{\mathbf{X}}_g, \mathbf{X}_g)$ is:

$$\Delta_{\mathcal{U}_{mmd}}(\mathbf{x}_g \mid \bar{\mathbf{X}}_g) = \frac{1}{m+1}\Bigg(\frac{2}{N_g}\sum_{\mathbf{x}_i \in \mathbf{X}_g} \mathrm{k}(\mathbf{x}_i, \mathbf{x}_g) - \frac{2}{N_g \cdot m}\sum_{\mathbf{x}_i \in \mathbf{X}_g, \bar{\mathbf{x}}_j \in \bar{\mathbf{X}}_g} \mathrm{k}(\mathbf{x}_i, \bar{\mathbf{x}}_j) + \\ + \frac{2m+1}{m^2(m+1)}\sum_{\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_j \in \bar{\mathbf{X}}_g} \mathrm{k}(\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_j) - \frac{2}{(m+1)}\sum_{\bar{\mathbf{x}}_i \in \bar{\mathbf{X}}_g} \mathrm{k}(\bar{\mathbf{x}}_i, \mathbf{x}_g) - \frac{1}{m+1}\mathrm{k}(\mathbf{x}_g, \mathbf{x}_g)\Bigg). \quad (4.5)$$

Discrete derivatives of $\mathcal{U}_{diff}$ and $\mathcal{U}_{div}$ can be built upon the discrete derivative of $\mathcal{U}_{mmd}$. $\mathcal{U}_{diff}$ is the difference between two $\mathcal{U}_{mmd}$ terms, hence we directly apply above equation to get its discrete derivative. The discrete derivative of λ term in $\mathcal{U}_{div}$ is given by first two terms of equation (4.5), and we ignore the remaining terms. After deriving the terms for discrete derivatives, we can apply them to Algorithm 1.

4.5.2 Gradient optimization

In optimizing the utility functions $\mathcal{U}_{diff}$ and $\mathcal{U}_{div}$, the second strategy is to re-cast the problem to allow for continuous optimization in the feature space, e.g. using standard gradient descent. A disadvantage of greedy approach is that it requires explicit computing of similarity matrix, which has quadratic memory requirements, making it infeasible for larger datasets. Gradient optimization help us to get around this problem. To generate prototypes, the solutions to this optimization can then be *snapped* to the nearest data points as a post-processing step.

Concretely, rather than searching for optimal prototypes $\bar{\mathbf{X}}_g$ directly, we seek the "meta-prototypes" $\bar{\mathbf{A}}_g = \{\bar{\mathbf{a}}_{g,1}, \ldots, \bar{\mathbf{a}}_{g,M}\}$, drawn from the same space as the document embeddings. We now modify $\mathcal{U}_{diff}$ (Equation 4.3) to incorporate "meta-prototypes". Note that $\mathcal{U}_{div}$ can be similarly modified, but $\mathcal{U}_{nn}$ cannot, since the max function is not differentiable. The "meta-prototypes" for $\mathcal{U}_{diff}$ are chosen to optimize

$$\max_{\bar{\mathbf{A}}_g}(-\mathrm{MMD}^2(\bar{\mathbf{A}}_g, \mathbf{X}_g) + \lambda \cdot \mathrm{MMD}^2(\bar{\mathbf{A}}_g, \mathbf{X}_{\neg g})). \quad (4.6)$$

The only difference to Equation 4.3 is that we do *not* enforce that $\bar{\mathbf{A}}_g \subset \mathbf{X}_g$. This subtle, but significant, difference allows Equation 4.6 to be optimized using gradient-following methods. We use L-BFGS [Byrd et al., 1995] with analytical gradient equations. The selected meta-prototypes $\bar{\mathbf{A}}_g$ are then snapped to the nearest document in the group: to construct the ith prototype for the gth group, we find

$$\bar{\mathbf{x}}_{g,i} = \operatorname*{argmin}_{\mathbf{x}_{g,j} \in \mathbf{X}_g} \|\bar{\mathbf{a}}_{g,i} - \mathbf{x}_{g,j}\|_2^2. \quad (4.7)$$

On a problem often tackled with discrete greedy optimization, one may wonder if gradient-based methods can be competitive; we answer this in the affirmative in our experiments.

Gradients: The equation and gradient of $MMD^2(\bar{\mathbf{A}}_g, \mathbf{X}_g)$ for the RBF kernel are determined to be as:

$$\mathrm{MMD}^2(\bar{\mathbf{A}}_g, \mathbf{X}_g) = -\frac{2}{M \cdot N_g} \sum_{i=1}^{N_g} \sum_{j=1}^{M_g} \mathrm{k}(\bar{\boldsymbol{a}}_{g,j}, \mathbf{x}_{g,i}) + \frac{1}{M_g^2} \sum_{i,j=1}^{M_g} \mathrm{k}(\bar{\boldsymbol{a}}_{g,i}, \bar{\boldsymbol{a}}_{g,j}) \tag{4.8}$$

$$\forall_{l \in 1 \ldots M_g} \nabla_{\bar{\boldsymbol{a}}_{g,l}} \mathrm{MMD}^2(\bar{\mathbf{A}}_g, \mathbf{X}_g) = \frac{4\gamma}{M_g} \Bigg(-\frac{1}{N_g} \sum_{i=1}^{N_g} \mathrm{k}(\bar{\boldsymbol{a}}_{g,l}, \mathbf{x}_i)(\mathbf{x}_i - \bar{\boldsymbol{a}}_{g,l}) + \frac{1}{M_g^2} \sum_{i=1}^{M_g} \mathrm{k}(\bar{\boldsymbol{a}}_{g,i}, \bar{\boldsymbol{a}}_{g,l})(\bar{\boldsymbol{a}}_{g,i} - \bar{\boldsymbol{a}}_{g,l}) \Bigg). \tag{4.9}$$

$\mathrm{MMD}^2(\mathbf{A}_g, \mathbf{X}_{\neg g})$ can also be computed similarly, by replacing $\mathbf{x}_{g,i}$ with $\mathbf{x}_{\neg g,i}$ in Eq (4.9). This will yield the objective of $\mathcal{U}_{diff}(\bar{\mathbf{X}})$ (4.3). The first term in Eq (4.9) corresponds to the gradient of first term of Eq (4.8). Hence, it will yield the gradient of the objective $\mathcal{U}_{div}(\bar{\mathbf{X}})$ (4.4).

4.6 Experiments and Results

Here, we first introduce the new evaluation strategies we developed for comparative summarization. We evaluate the summary prototypes using both automatic and crowd-sourced evaluations. We then elaborate on the experiments settings and present results for comparatively summarizing controversial news topics over time.

4.6.1 Evaluations

We can evaluate the *intrinsic* quality of a summary in terms of information coverage, and non-redundancy. Intrinsic evaluation of summaries often requires human written ground truth summaries, which are costly to curate. A commonly used intrinsic evaluation in summarization is ROUGE [Lin, 2004], which evaluates the information coverage by comparing the n-grams match between the system and reference summary (§2.4.2). An alternative is to evaluate the usefulness of summaries on extrinsic tasks such as classification performance. Such evaluation is called an *extrinsic* in the NLP literatures [Jones and Galliers, 1995]. Recall that we provided a brief overview of different summarization evaluations in §2.4.

We *extrinsically* evaluate different approaches to comparative summarization using both automatic and crowd-sourced human classification tasks. This choice of extrinsic evaluations stems from our classification perspective (see §4.3.2), and has been used in the prototype selection literature [Bien and Tibshirani, 2011; Kim et al., 2016]. Furthermore, several evaluation protocols were used to determine the relevance of a topic to a summary, or to an entire document in information retrieval scenario [Mani et al., 1999; Mani and Bloedorn, 1997; Tombros

and Sanderson, 1998; Brandow et al., 1995]. In the retrieval experiment by Mani et al. [1999] for query focused summarization, human subjects were presented with a pair of a topic description, and an entire document or a summary, and asked to determine the relevance of the document or the summary to the topic. In their second experiment, they present the human subjects with a generic summary or an entire, and ask them to categorize into one of five categories. They argue that the summary would be useful if human judges can accurately identify the relevance, or the category in these tasks. They found that the accuracies are not significantly different in both tasks, and summaries would significantly speed up decision making in the first task. Similarly, in comparative summarization, we hypothesize that a good set of prototype articles should uniquely identify a new article's group without much impact in classification performance, when evaluated with automatic or human classification. A collection of prototype documents should give an indicative idea of the group it intends to represent to a classifier or to a human subject.

The advantage of this evaluation is that it is applicable to unsupervised settings as we no longer need the dataset with human written ground truth summaries. Furthermore, automatic evaluation can be applied to large scale datasets. This addresses the RQ3 of our thesis, which is about evaluating comparative summaries in unsupervised settings. We elaborate on out automatic evaluation settings in §4.6.3, and crowd-sourced human evaluation settings in §4.6.5.

4.6.2 Experiment settings and baselines

Before going into details of the experiment, we now provide a brief overview of how the summarization method and evaluation fits together. We illustrate this in Figure 4.3. We first split the datasets into training and test sets, and then summarize by selecting prototypes from training set. Then we train a classifier on prototypes only, which we use to evaluate the performance on test set for automatic evaluation. Similarly, for human evaluation, we present the two groups of summaries to human judges and ask them to classify the documents from test set to either of two groups.

Datasets and features. We empirically validate the classification and prototype selection methods on the standard USPS dataset [Bien and Tibshirani, 2011; Kim et al., 2016]. USPS contains 16×16 grayscale handwritten digits in 10 classes (i.e., digits 0 through 9). In using the USPS dataset, our aims are twofold. First, it shows the versatility of the method: the domain need not be text, collections need not be separated by time, and the number of classes need not be less than three. Indeed, by thinking of each digit as a *group*, our method can identify representative and diverse examples of digits. Second, our method can be seen as a special kind of prototype selection for which the USPS dataset has been used as a standard benchmark [Bien and Tibshirani, 2011; Kim et al., 2016].

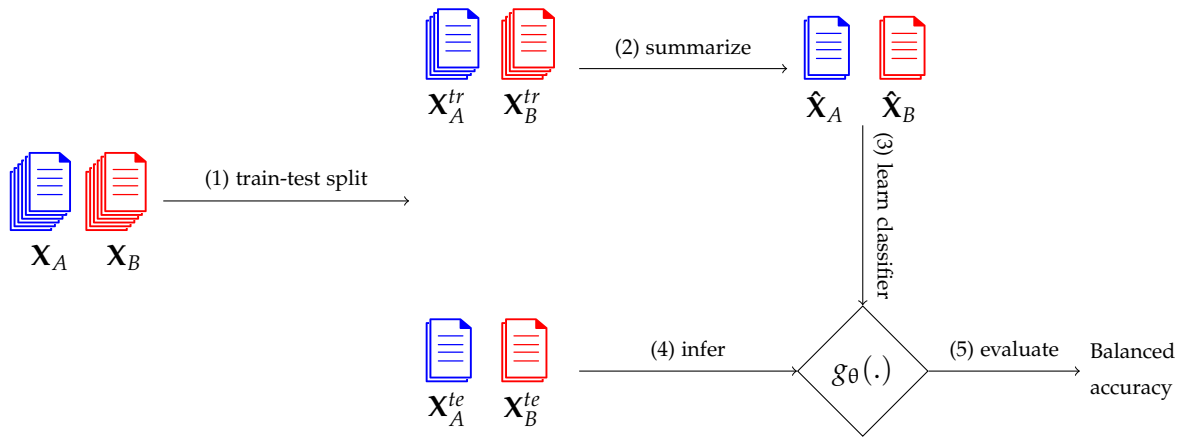


Figure 4.3: Combined pipeline illustrating comparative summarization and evaluation.

The USPS dataset provides 7291 training and 2007 test images. We generate another 9 random splits with exactly the same number of training and test images for the purposes of estimating confidence intervals. To reduce the dimensionality and thereby reduce the computation time, we use PCA, projecting the 256 dimensional image vectors into 39 features that explain 85% of the variance.

We further use the ControvNews2017 dataset described in §3.3 to evaluate comparative summarization. We adopt the pre-trained GloVe-300 [Pennington et al., 2014] vector representation for each word, and then represent the article as an average of the word vectors from its the title and first 3 sentences – the most important text due to the inverted pyramid structure in news style [Pöttker, 2003]. This feature performs competitively in retrieval tasks despite its simplicity [Joulin et al., 2016]. For each news topic, we generate 10 random splits with 80% training articles and 20% test articles for automatic evaluation. One of these splits is used for human evaluation.

Approaches and baselines. We compare:

- *nn-comp-greedy*, which represents the nearest neighbor objective $\mathcal{U}_{nn}$, optimized using greedy algorithm.
- *mmd-diff*, which represents the difference of MMD objective $\mathcal{U}_{diff}$. *mmd-diff-grad* uses gradient based optimization while *mmd-diff-greedy* is optimized greedily.
- *mmd-div-grad* and *mmd-div-greedy*, which are the gradient-based and greedy variants of the diverse MMD objective $\mathcal{U}_{div}$.

with three baseline approaches:

- *kmeans* clusters with kmeans++ initialization [Arthur and Vassilvitskii, 2007] found sep-

arately for each document group. The M cluster centers for each group are snapped to the nearest data point using Equation 4.7.

- *kmedoids* [Kaufman and Rousseeuw, 1987] clustering algorithm with kmeans++ initialization, computed separately for each document group. The medoids become the prototypes.
- *mmd-critic* [Kim et al., 2016] which selects prototypes using greedy optimization of MMD^2 and criticisms by choosing points that deviate from the prototypes. The summary is selected from the unlabeled training set and consists of prototypes and criticisms in a one-to-one ratio.

4.6.3 Automatic evaluation settings

The ControvNews2017 dataset topics are divided into two groups of equal duration based on article timestamp. Note that typically the number of documents in each time range is imbalanced. The USPS handwritten digits dataset is divided into 10 groups corresponding to the 10 different digits. As discussed in 4.6.1, we use classification performance to *extrinsically* evaluate quality of the prototypes. On each training split we select the prototypes for each group and then train support vector machine (SVM) and 1-nearest neighbor (1NN) on the set of prototypes. We use the Radial Basis Function (RBF) kernel when applicable. The hyperparameter γ is chosen along with the trade-off factor λ, and SVM soft margin C using grid search 3-fold cross-validation on the training set. Note that 1NN has no tunable parameters. The *grad* optimization approach uses the L-BFGS algorithm [Byrd et al., 1995], with initial prototype guesses chosen by the *greedy* algorithm for the ControvNews2017 dataset and K-means for the USPS dataset.

We measure the classifier performance on the test set using balanced accuracy, defined as the average of the per-class accuracies[Brodersen et al., 2010]. For binary classification, this is $\frac{1}{2}(\frac{\text{TP}}{\text{P}} + \frac{\text{TN}}{\text{N}})$, defined in terms of total positives P, total negatives N, true negatives TN, and true positives TP. Balanced accuracy accounts for class imbalance, and is applicable to both binary and multi-class classification tasks. For all approaches, we report the mean and 95% confidence interval over the 10 random splits.

We report results on 2, 4, 8, or 16 prototypes per group because a small number of prototypes is necessary for the summaries to be meaningful to humans. This is in contrast to the hundreds of prototypes used by Bien and Tibshirani [2011]; Kim et al. [2016], in automatic evaluations of the predictive quality of prototypes.

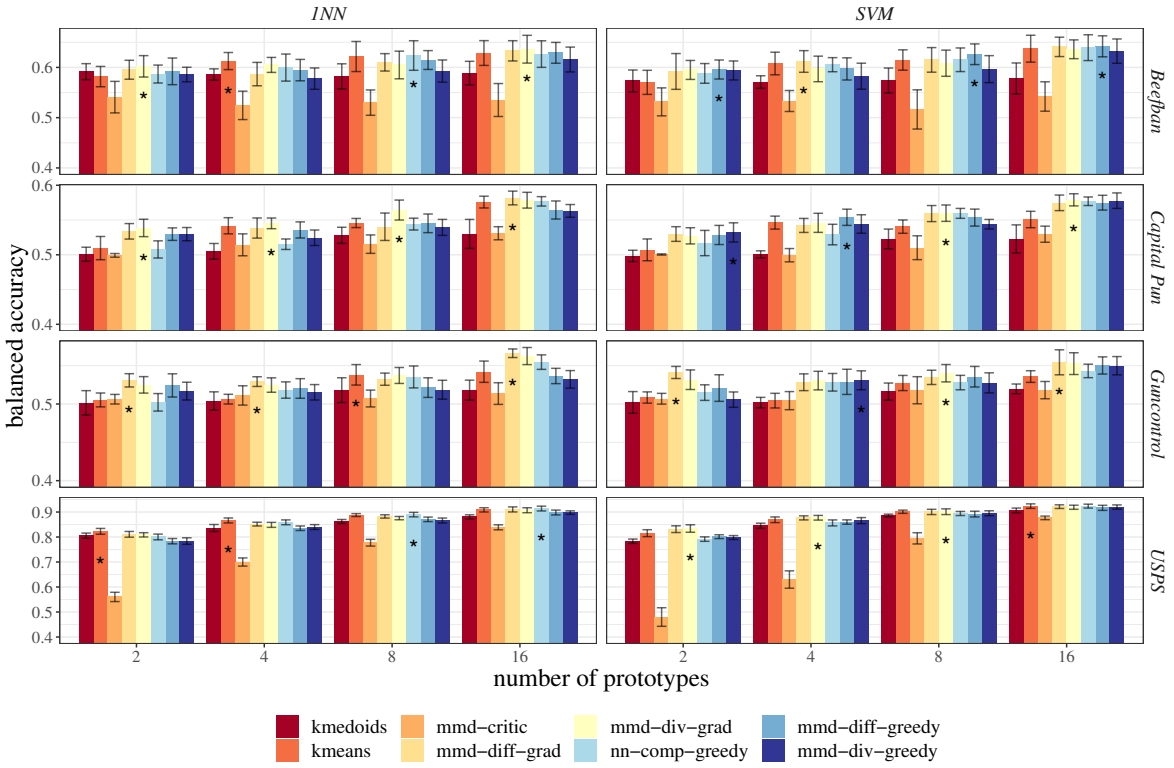


Figure 4.4: Comparative summarization methods evaluated using the balanced accuracy of 1-NN (left) and SVM (right) classifiers. Each row represents a dataset. Error bars show 95% confidence intervals.

4.6.4 Automatic evaluation results

Figure 4.4 reports the balanced accuracy of SVM and 1NN trained on the prototypes generated by all methods. We annotate the bar with '*' if it is best performing in an evaluation. On the USPS dataset, most methods perform well, and the differences are small. *Mmd-critic* performs poorly on USPS; this is because it does not guarantee a fixed number of prototypes per group, and sometimes misses a group altogether. Note that this is not a problem with on news dataset as it is very unlikely to occur with only 2 groups in the ControvNews2017 dataset.

On the ControvNews2017 datasets, comparative summaries based on *mmd* objectives are the best-performing approach in 21 out of 24 evaluations (2 classifiers x 4 prototype sizes x 3 news topics. In three out of four remaining cases, they are the second-best with overlapping confidence intervals against the best. Despite the lack of optimization guarantees, *grad* optimization produces prototypes of better quality in 15 out of 24 settings. Our *mmd* based objectives perform best in 9 out of 12 cases using 1NN evaluation, and 12 out of 12 evaluations using SVM evaluations. The detailed table for figure 4.4 is provided in Appendix A.2.

Generally, all methods produce better classification accuracy as the number of prototypes

increases. This indicates that the chosen prototypes do introduce new information that helps with the classification. In the case where all documents are selected as prototypes – a setting that is clearly unreasonable when summarization is the goal – the performance is determined by the classifier alone. In this setting, SVM achieves 0.763 on *Capital Punishment* and *Beef Ban*, 0.707 on *Gun Control*, while 1-NN achieves 0.762 on *Capital Punishment*, 0.763 on *Beef Ban* and 0.702 on *Gun Control*. As seen in Figure 4.4, no prototype selection method approaches this accuracy. This highlights the difficulty of selecting only a few prototypes to represent complex distributions of news articles over time.

4.6.5 Crowd-sourced evaluation settings

We conduct a user study on the crowd-sourcing platform figure-eight[2] with two questions in mind: (1) using article classification accuracy as a proxy, do human perform similarly to automatic evaluation?, and (2) how useful do humans find the comparative summaries? This is an acid test on providing value to users who need to comprehend large document corpora. Human evaluations in this work are designed to grade our method in a real world task: accurately identifying a news article's group (e.g. the month it is published) given only a few (4) articles from each month. The automatic evaluations in §4.6.3 are instructive proxies for efficacy, but inherently incomplete without human evaluation.

Generating summaries for the crowd. We present summaries from four methods – *kmeans*, *nn-comp-greedy*, *mmd-diff-greedy*, and *mmd-diff-grad* – chosen because they perform well in automatic evaluation and together form a cross-section of different method types. We opt to vary the groups of news articles being summarized by choosing many pairs of time ranges, since summaries on the same pair of groups (by definition) tend to be very similar or identical, which incurs user fatigue. We use the *Beef Ban* topic because it has the longest time range: June 2017 to July 2018 inclusive. The articles are grouped into each of the 14 months, and then 91 (i.e., $\binom{14}{2}$) pairs are formed. We take pairs as judged by the automatic evaluation scores, using each of the four approaches, the union of these lead to 21 pairs. We pick the top-performing pairs because preliminary human experiments showed that humans are unable to classify an article when automatic results do poorly (e.g. $<$0.65 in balanced accuracy). Articles from each of the 21 pairs of months are randomly split into training and testing sets. We ask participants to classify six randomly sampled test articles. To reduce evaluation variance, all methods share the same test articles, different methods are randomized and are blind to workers. We record three independent judgments for each (test article, month-pair) tuple – totaling 1,512 judgments from 126 test questions over four methods. We also restrict the crowd workers to be from India, where *Beef Ban* is locally relevant, and workers will be familiar with the people, places and organizations mentioned in the news articles.

[2]https://www.figure-eight.com

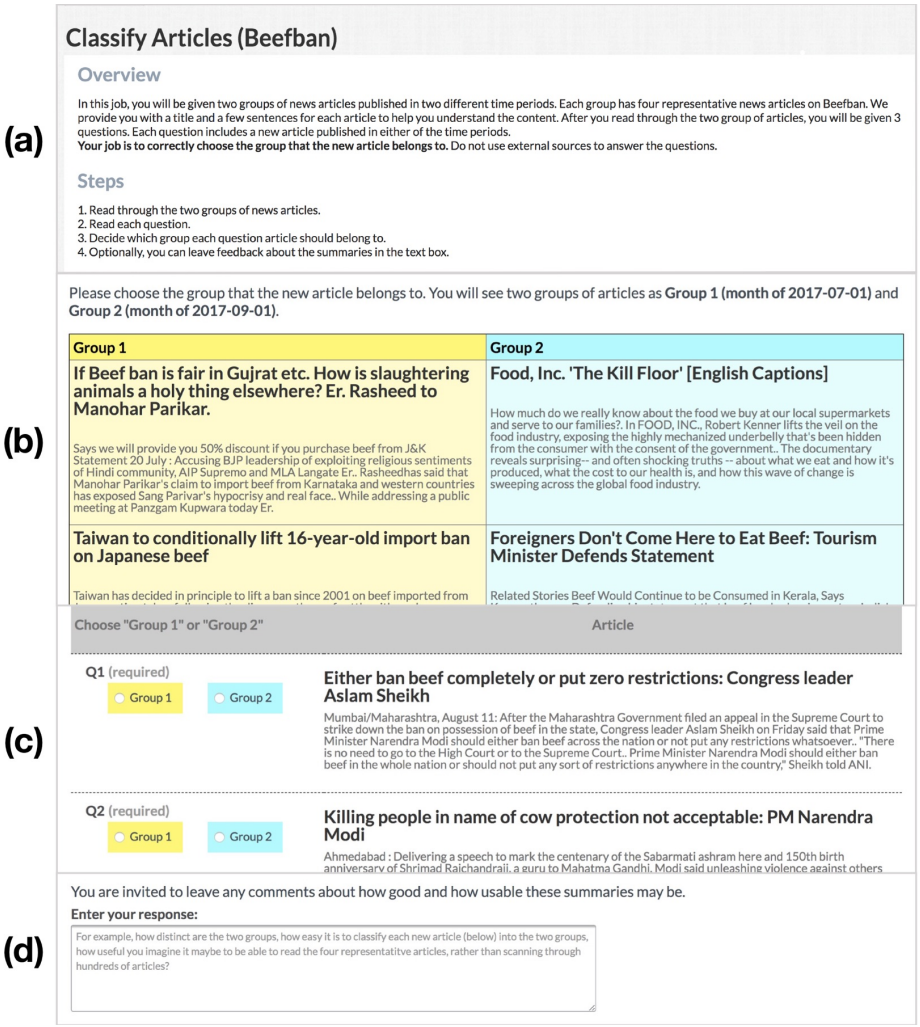


Figure 4.5: An example questionnaire used for crowd-sourced evaluation. It consists of: (a) instructions, (b) two groups of summaries, (c) question articles, and (d) a comment box for feedback. See §4.6.5.

Questionnaire design. Figure 4.5 shows the questionnaires we designed for human evaluation. Each questionnaire has 4 parts: (a) instructions, (b) two groups of prototypes, (c) test articles that must be classified into a group, and (c) a comment box for free-form feedback.

In the instruction (a), we explain that the two groups of representative articles (the prototypes for each time range) are articles from different time ranges and lay out the steps to complete the questionnaire. We ask participants not to use external sources to help classify test articles.

The two groups of prototype articles (b) are chosen by one of the method being evaluated (e.g., *mmd-diff-grad* or *kmeans*) from articles in two different time ranges. Each group has four representative articles and each article has a title and a couple of sentences to help participants

Method	#unique workers	correct by majority	correct judgements
kmeans	31	81	240
mmd-diff-grad	25	94	270
nn-comp-greedy	28	80	243
mmd-diff-greedy	29	83	235
Total	40	126	378

Table 4.1: Results of a human pilot study on Classification Task. Unique workers participating in classifying test articles for each method is in the first column. Correct by majority denotes the number of test articles (out of 126) classified correctly by majority (at least two people). Correct Judgments indicates the number of individual judgments that are correct (out of 378)

understand the content. We assign a different background color to each group of summaries to give participants a visual guide.

Below the groups of summary articles are three questions (c), though for brevity only two are shown in Figure 4.5. Each question asks participants to decide which of the two time ranges a test article belongs to.

We add a comment box (d) to gather free-form feedback from participants. This helps to quickly uncover problems with the task, provides valuable insight into how participants use the summaries to make their choices, and gives an indication of how difficult users find the task. As a quality-control measure, we include questions with known ground truth amongst the test questions. These ground truth questions are manually curated, and we reviewed any questions on which several workers failed. Each unit of work includes 4 questionnaires (of 3 questions each), one of which is a group of ground truth questions randomly positioned. Note that ground truth questions are only used to filter out participants and are not included in the evaluation results.

4.6.6 Crowd-sourced evaluation results

We now describe our crowd-sourced evaluation results.

Worker profile. The number of unique participants answering test questions ranged from 25 to 31 for each method as seen in Table 4.1, indicating that the results were not dominated by a small number of participants. On average, participants spent 51 seconds on each test question and 2 minutes 33 seconds on each summary.

Quantitative results: Figure 4.6 shows that on average crowd workers with *mmd-diff-grad* summaries classify an article more accurately than summaries from other approaches by at least 7%. The results are statistically significant with $p < 0.05$ under a one-sided *sign test*, which applies because the 126 test questions where answered by three random people, and we cannot assume normality. It also has the highest number of consensus correct judgments as seen in Figure 4.7 (left), suggesting the usefulness of the summaries.

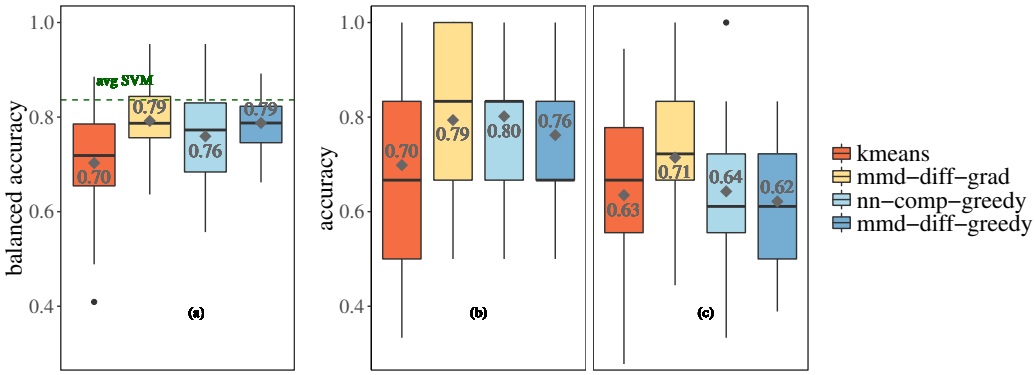


Figure 4.6: Classification accuracies for 21 pairs of summaries. (a) Automatic classification using prototypes (by SVM) on the entire test set. The green *avg SVM* line is the mean accuracy of SVMs trained on the entire training set. (b) Automatic classification evaluated on 6 test articles per pair. (c) Human classification accuracy on 6 test articles per pair.

The good performance of gradient-based optimization is surprising given greedy approaches are usually preferred in subset selection tasks, due to approximation guarantees for submodular objectives. One plausible explanation is that early prototypes selected by *greedy* tend to cluster around the first prototype, whereas the simultaneous optimization in *grad* tend to spread prototypes in feature space. With only four prototypes being shown to users, diversity is an important factor for human classification. Previous studies of *greedy* methods for prototype selection have used hundreds of prototypes [Bien and Tibshirani, 2011] – a setting in which the diversity of the early prototypes matters less – or used criticisms [Kim et al., 2016] to improve diversity in tandem.

Comparing Figure 4.6 (a) – (c), automatic classifiers trained on both the entire training set and prototypes have higher classification accuracy than human workers across all methods. This indicates that using summaries to classify articles is difficult for humans. It could also indicate that humans use different features for article grouping, and word vectors alone may not capture those features. But, the drop in performance in the best performing method (*mmd-diff-grad*) is less compared to other methods in human evaluation. When we measure accuracy by majority correct instead of total correct, the human accuracy is 74.6% instead of 71.4%, which is even closer to the classifier trained on entire training set. Furthermore, in a large scale measurement study in Chapter 6, we observe that the accuracy of a classifier trained on prototypes is proportional to the one trained on entire training dataset across different datasets, and different facets of comparisons. We observe that the human judgement is moderately correlated with the automatic evaluation for kmeans and *mmd-diff-grad* as seen in Figure 4.7 (right). The reason for moderate correlation (spearman $\rho = 0.4$, $p < 0.01$) is due to small number of comparison pairs (21). Nevertheless, we observe a pattern that human judgements increases as automatic evaluation performance increases, which is often

used to justify the evaluations proposed in natural language processing [Papineni et al., 2002; Lin, 2004; Anderson et al., 2016]. Overall, we conclude our evaluation is useful in evaluating comparative summaries.

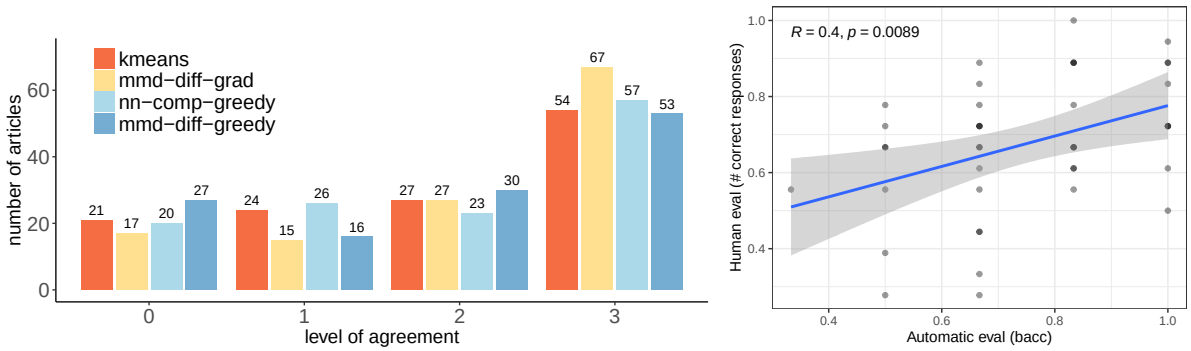


Figure 4.7: (Left) Shows the number of test articles that humans correctly classified, at different agreement levels. A level of 3 means all participants correctly classified the test article, while 0 means all participants incorrectly classified the test article. (Right) shows the human accuracy vs automatic accuracy for *mmd-diff-grad* and kmeans methods for 21 comparison pairs.

Inter annotator agreements: Figure 4.7 (left) shows the level of agreement across participants for each method. First we note that participants were frequently able to complete the task of classifying new articles correctly into one of two groups: this is shown by the large fraction of articles for which the correct group was unanimously chosen. Compared to other comparative prototype selection methods, *mmd-diff-grad* has the largest number of articles correctly classified by all three participants, beating the next best *nn-comp-greedy* by 10 articles. Consequently, *mmd-diff-grad* also has fewer articles which were unanimously assigned to the incorrect group by participants.

We also compute the Fleiss Kappa statistic to measure inter-annotator agreement. The statistics are: 0.418 for *kmeans*, 0.456 for *mmd-diff-grad*, 0.435 for *nn-comp-greedy*, and 0.483 for *mmd-diff-greedy* and a combined statistic of 0.451. All statistics fall into the range of moderate agreement [Landis and Koch, 1977], which means the results we obtain in crowdsourced evaluations are reliable.

Example summary. We show an example of comparative summary prototypes (title only) from the *Beefban* topic, using two methods kmeans and $\mathcal{U}_{diff}$ for comparing two months. In the summary produced by kmeans, we observe that the third article of first month and the first article of second month are similar. This is due to a lack of discriminativeness in the kmeans method.

Qualitative observations. Results from the optional free-form comments show that the participants found the classification difficulty to vary wildly. While some sets of articles were apparently easy to classify (e.g., "Group articles are distinct in their manner, among which

August 2017	May 2018
Privacy verdict to have 'some bearing' in beef ban matters: SC	No ban on cow slaughter, BJP says in poll-bound Meghalaya
Maha: Maharashtra beef ban law to be tested in light of privacy verdict: SC	Three years after beef ban, slaughter of buffaloes at all-time high in Maharashtra - Nagpur Today : Nagpur News
BJP's 'no beef ban in Northeast' exposes its hypocrisy	Japan to Lift Part of Import Ban on Argentine Beef
Trump's top trade nominees lobbied for hormone-meat exports	85 kg plastic waste removed from stomach of an abandoned bull
Privacy ruling to affect beef ban	Three years after beef ban, slaughter of buffaloes in Maharashtra at an all-time high
Privacy verdict to have 'some bearing' in beef ban matters: Supreme Court	Australia Finally Lifts Ban on Japanese Beef Imports
Beef ban: Goa's Catholics not apprehensive, says CM Parrikar	McDonald's is being sucked into the movement to ban plastic straws
Beef ban: Supreme Court to hear Maharashtra Government's appeal challenging Bombay High Court order	No ban on cow slaughter, BJP says in poll-bound Meghalaya

Table 4.2: An example summary prototypes (titles only) from *Beefban* topic using kmeans(top half) and $\mathcal{U}_{diff}$ (bottom half) from two months.

all are articles are easy to determine."), other articles were difficult to classify (e.g., "Although two groups are clearly distinct, this one (news article) was pretty difficult to ascertain in which group it belongs to"). In some cases poor summaries seem to have made the task exceedingly difficult; e.g., "Q1, Q2, Q3 all are not belongs to group 1 and group 2 any topic I think." (quoted verbatim).

We found that the *Beef Ban* topic interested many of our participants, with some expressing their views on the summarized articles, for example "Firstly we should define what is beef ..is it a cow or any animal?" and "It is a broad matter, what we should eat or not, it cannot be decided by government." (edited for clarity). Participant comments also give some insight into what features were used to make classification. In particular, word and entity matching were frequently mentioned, a representative user comment is "None of the questions match the given article, but I had to go by words used." All crowd-sourced evaluation results and comments are available in the dataset repository.

4.7 Conclusion

In this chapter, we formulated comparative document summarization in terms of competing binary classifiers. This inspired new MMD based objectives amenable to both gradient and greedy optimization, thus addressing the first research question (RQ1) of this thesis. Moreover, the setting enabled us to design efficient automatic and human evaluations, addressing the third research question (RQ3) of this thesis. We use the evaluation to compare different

objectives and optimization methods on a ControvNews2017 dataset of controversial news topics. We found that our new MMD approaches, optimized by gradient methods, frequently outperformed all alternatives, including the greedy optimizations currently favored by the literature.

Chapter 5

Supervised Model for Comparative Summarization

“ Don’t hate it for being simple. Levers are simple too, but they can move the world. ”

Cassie Kozyrkov, *2018*

In this chapter, we present SupMMD, a novel supervised learning technique for generic and comparative multi-document summarization, addressing the research question 2 (RQ2) of the thesis. SupMMD is based on the *maximum mean discrepancy* from kernel two-sample testing [Gretton et al., 2012a], that we introduced in §2.6.3. SupMMD combines both supervised learning for sentence salience scoring and unsupervised learning for maximizing the information coverage and diversity. Further, we adapt multiple kernel learning by Cortes et al. [2010] to make use of similarity across multiple information sources (e.g., text features and knowledge based concepts). We show the efficacy of SupMMD in both generic and update summarization (hybrid of generic and comparative summarization as introduced in §1.1) tasks by meeting or exceeding the current state-of-the-art on the DUC-2004 and TAC-2009 datasets. We make the software for this chapter’s experiments publicly available[1]. This work is published in Bista et al. [2020].

5.1 Introduction

Recall from Chapter 2 that multi-document summarization is the problem of producing condensed digests of salient information from multiple sources, such as articles. Multi-document extractive summarization works by extracting few key sentences (or documents) and combining them to form the summary. Most existing work on this topic has focused on the

[1] https://github.com/computationalmedia/supmmd

generic summarization task. However, update summarization is of equal practical interest. As introduced in §1.1, update summarization is a hybrid case of generic and comparative summarization. Intuitively, the comparative aspect of this setting aims to inform a user of new information on a topic they are already familiar with.

Multi-document extractive summarization methods can be unsupervised or supervised. *Unsupervised* methods typically define salience (or coverage) using a global model of sentence-sentence similarity. Methods based on retrieval [Goldstein et al., 1999], centroids [Radev et al., 2004], graph centrality [Erkan and Radev, 2004], or utility maximization [Lin and Bilmes, 2010, 2011; Gillick and Favre, 2009] have been well explored. However, sentence salience also depends on *surface features* (e.g., position, length, presence of cue words); effectively capturing these require supervised models specific to the dataset and task. A body of work has incorporated such information through *supervised learning*, for example based on point processes [Kulesza and Taskar, 2012], learning important words [Hong and Nenkova, 2014], graph neural networks [Yasunaga et al., 2017], and support vector regression [Varma et al., 2009]. These supervised methods have either a separate model for learning and inference, leading to a disconnect between learning sentence salience and sentence selection [Varma et al., 2009; Yasunaga et al., 2017; Hong and Nenkova, 2014], or are designed specifically for generic summarization [Kulesza and Taskar, 2012].

In this chapter, we propose *SupMMD*, which has a single model of learning sentence salience and summaries inference. We use *SupMMD* to generic and update multidocument summarization tasks, where we summarize a group of documents by selecting a few prototype sentences, unlike in Chapter 4, where we picked prototype documents. We make the following contributions:

1. We present *SupMMD*, a novel technique for both generic and update summarization that combines supervised learning for salience and unsupervised learning for coverage and diversity. *SupMMD* has a single model for learning and inference.

2. We adapt multiple kernel learning [Cortes et al., 2010] into our model, which allows similarity across multiple information sources (e.g., text features and knowledge based concepts) to be used.

3. We show that *SupMMD* meets or exceeds the state-of-the-art in generic and update summarization on the DUC-2004 and TAC-2009 datasets.

5.2 Literature review

As introduced in Chapter 2, multi-document summarization can be *extractive*, where salient pieces of the original text such as sentences are selected to form the summary; or *abstractive*,

where a new text is generated by paraphrasing important information. The former is popular as it often creates semantically and grammatically correct summaries [Nallapati et al., 2017]. In this work, we focus on *generic* and *update multi-document* summarization in the *extractive* setting.

Most extractive summarizers have two components: sentence scoring and selection. A variety of unsupervised and supervised methods have been developed for the former. *Unsupervised* sentence scorers are based on centroids [Radev et al., 2004], graph centrality such as in Lexrank [Erkan and Radev, 2004], retrieval relevance [Goldstein et al., 1999], word statistics [Nenkova and Vanderwende, 2005], topic models [Haghighi and Vanderwende, 2009], or concept coverage [Gillick and Favre, 2009; Lin and Bilmes, 2011]. *Supervised* techniques include: using a graph-based neural network [Yasunaga et al., 2017], learning sentence quality from point processes [Kulesza and Taskar, 2012], combining word importances [Hong and Nenkova, 2014], combining sentence and phrase importances [Cao et al., 2015], employing a mixture of submodular functions [Lin and Bilmes, 2012], or sequence labeling with RNNs [Nallapati et al., 2017].

Sentence selection methods can be broadly categorized as *greedy* methods [Goldstein et al., 1999; Radev et al., 2004; Erkan and Radev, 2004; Nenkova and Vanderwende, 2005; Cao et al., 2015; Haghighi and Vanderwende, 2009; Hong and Nenkova, 2014; Kulesza and Taskar, 2012; Cao et al., 2015; Varma et al., 2009], which produce approximate solutions by iteratively selecting the sentences with the maximal score, or *exact* integer linear programming (ILP) based methods [Gillick and Favre, 2009; Cao et al., 2015]. Some greedy methods use an objective which belongs to a special class of set functions called *submodular functions* [Lin and Bilmes, 2010, 2012, 2011; Kulesza and Taskar, 2012], which have good approximation guarantees under greedy optimization [Nemhauser et al., 1978].

There has been limited research into update and comparative summarization. Notable prior work includes maximizing concept coverage using ILP [Gillick et al., 2009], learning sentence scores using a support vector regressor [Varma et al., 2009], and temporal content filtering [Zhang et al., 2009]. In §4.3, we cast the comparative summarization problem as classification, and introduced MMD [Gretton et al., 2012a] based unsupervised extractive summarization objectives (§4.4). In this work, we extend the method to learn *sentence importances* driven by surface features.

5.3 The SupMMD method

Now, we describe the methodology in detail. We start by revisiting the notations, and developing a technique for incorporating sentence importance into MMD for the purpose of generic multi-document extractive summarization. We then extend this method to compara-

tive summarization, and incorporate multiple different kernels to use a diverse set of features. SupMMD is a supervised multi-document summarization technique, thus it addresses the second research question (RQ2) of the thesis (RQ2).

5.3.1 Supervised extractive multi-document summarization problem

In Chapter 4, we did not make distinction of topics as we were working in unsupervised summarization problem setting. Now, since, we focus on a supervised learning settings, we would like to learn a common function applicable to all topics. We will discuss this common function in detail in next subsection. Formally, the multi-document extractive summarization problem of topic t is a prototype selection problem $\hat{\mathbf{X}}^t \subset \mathbf{X}^t$.

The next difference with Chapter 4 is in the constraint. We now have a budget (number of words) constraint $\sum_{x \in \bar{\mathbf{x}}} \text{len}(x) \leq L$, instead of cardinality constraint. This is because, we evaluate our models in standard DUC-2004 and TAC-2009 datasets, which has 100 words limit on the summaries.

In update summarization, we have two groups of documents in each topic, i.e., $\mathbf{X}^t = \mathbf{X}^t_A \cup \mathbf{X}^t_B$. Recall, update summarization is generic summarization of set A, and comparative summarization of group B against the group A. Our summarization goal is to select subsets $\hat{\mathbf{X}}^t_A \subset \mathbf{X}^t_A$ and $\hat{\mathbf{X}}^t_B \subset \mathbf{X}^t_B$ as summaries with budget constraints for each summary.

5.3.2 From MMD to weighted MMD

Unsupervised MMD we introduced in Chapter 4 selects representative sentences that cover relevant concepts while retaining diversity. The notion of representativeness is based on a global model of sentence-sentence similarity; however, this notion of representativeness is not necessarily well-matched to the selection of salient information. Salience of a sentence may be determined by *surface features* such as position in the article, or number of words. For example, news articles are often written such that sentences at the start of a article have the characteristics of a summary [Kedzie et al., 2018]. Learning a notion of salience that is specific to the summarization task and dataset requires supervised training. Thus, we extend the MMD model by incorporating supervised *sentence importance weighting*.

Let $\mathbf{x}, \bar{\mathbf{x}} \in \mathcal{X}$ be independent samples drawn from the distributions of article sentences p and summary sentences (either ground truth or inferred from our models) q on the space of all sentences $\mathcal{X}$. We define non-negative *importance functions* f^p_θ, f^q_θ parameterized by learnable parameters θ. We restrict these functions so that $\mathbb{E}_p f^p_\theta(\mathbf{x}) = 1$ and $\mathbb{E}_q f^q_\theta(\bar{\mathbf{x}}) = 1$. Equipped with f_θ, we may modify MMD such that the importance of sentences which are good summary candidates is increased.

Definition 5.1. The *weighted MMD*, $\text{wMMD}_{\mathcal{F}}(p, q, \theta)$ between p, q is

$$\sup_{h \in \mathcal{F}} \left\{ \mathop{\mathbb{E}}_{p} \left[f_\theta^p(x) \cdot h(\mathbf{x}) \right] - \mathop{\mathbb{E}}_{q} \left[f_\theta^q(\bar{\mathbf{x}}) \cdot h(\bar{\mathbf{x}}) \right] \right\} \tag{5.1}$$

Note that classic MMD (2.5) is a special case of (5.1) where $f_\theta \equiv 1$.

In practice, the supremum over all h is impossible to compute directly. We thus derive an alternative form for Equation 5.1.

Lemma 5.1. For $\|h\|_{\mathcal{H}} \leq 1$, and let $\phi : \mathcal{X} \mapsto \mathcal{F}$ is a canonical feature mapping of sentences and summaries from $\mathcal{X}$ to RKHS. Then Equation (5.1) is equivalently

$$\left\| \mathbb{E}_p[f_\theta^p(\mathbf{x}) \cdot \phi(\mathbf{x})] - \mathbb{E}_q[f_\theta^q(\bar{\mathbf{x}}) \cdot \phi(\bar{\mathbf{x}})] \right\|_{\mathcal{H}}. \tag{5.2}$$

Proof of Lemma 5.1 Recall f_θ is a non-negative importance weighting function. Then, according to Patil and Rao [1978], the weighted probability density $\bar{p}_\theta$ of p is:

$$\bar{p}_\theta(\mathbf{x}) = \frac{f_\theta^p(\mathbf{x}) \cdot p(\mathbf{x})}{\mathbb{E}_p[f_\theta^p(\mathbf{x})]}$$

and similarly $\bar{q}_\theta$ for q. Since we restrict $\mathbb{E}_p[f_\theta^p(\mathbf{x})] = 1$, and $\mathbb{E}_q[f_\theta^q(\bar{\mathbf{x}})] = 1$, we have $\bar{p}_\theta(\mathbf{x}) = f_\theta^p(\mathbf{x}) \cdot p(\mathbf{x})$ and $\bar{q}_\theta(\bar{\mathbf{x}}) = f_\theta^q(\bar{\mathbf{x}}) \cdot q(\bar{\mathbf{x}})$. Thus, the weighted MMD is

$$\begin{aligned} &\sup_{h \in \mathcal{F}} \left(\mathop{\mathbb{E}}_{\mathbf{x} \sim \bar{p}_\theta} [h(\mathbf{x})] - \mathop{\mathbb{E}}_{\bar{\mathbf{x}} \sim \bar{q}_\theta} [h(\bar{\mathbf{x}})] \right) \\ &= \sup_{||h||_{\mathcal{H}} \leq 1} \left(\mathop{\mathbb{E}}_{\mathbf{x} \sim \bar{p}_\theta} [h(\mathbf{x})] - \mathop{\mathbb{E}}_{\bar{\mathbf{x}} \sim \bar{q}_\theta} [h(\bar{\mathbf{x}})] \right) \end{aligned}$$

Since in an RKHS, $\mathbb{E}_p[h(x)] = \left\langle h,\ \mathbb{E}_p[\phi(x)] \right\rangle_{\mathcal{H}}$, this simplifies to:

$$\begin{aligned} &\sup_{||h||_{\mathcal{H}} \leq 1} \left\langle h, \mathop{\mathbb{E}}_{\mathbf{x} \sim \bar{p}_\theta} [\phi(\mathbf{x})] - \mathop{\mathbb{E}}_{\bar{\mathbf{x}} \sim \bar{q}_\theta} [\phi(\bar{\mathbf{x}})] \right\rangle_{\mathcal{H}} \\ &= \left\| \mathop{\mathbb{E}}_{\mathbf{x} \sim \bar{p}_\theta} [\phi(\mathbf{x})] - \mathop{\mathbb{E}}_{\bar{\mathbf{x}} \sim \bar{q}_\theta} [\phi(\bar{\mathbf{x}})] \right\|_{\mathcal{H}} \\ &= \left\| \mathop{\mathbb{E}}_{\mathbf{x} \sim p} \left[f_\theta^p(\mathbf{x}) \cdot \phi(\mathbf{x}) \right] - \mathop{\mathbb{E}}_{\bar{\mathbf{x}} \sim q} \left[f_\theta^q(\bar{\mathbf{x}}) \cdot \phi(\bar{\mathbf{x}}) \right] \right\|_{\mathcal{H}} \end{aligned} \quad \blacksquare$$

where the penultimate step follows from the definition *Dual norm* [Daners, 2008, p. 85]. The proof is similar to MMD in [Gretton et al., 2012a].

5.3.3 Importance function

We use log-linear models as importance functions, as they are a common choice of sentence importance [Kulesza and Taskar, 2012] and easy to fit when training data is scarce. Formally, the log-linear importance function is: $f_\theta(\mathbf{x}) = \exp(\langle \theta, \omega(\mathbf{x}) \rangle)$, where $\omega(\mathbf{x})$ is the surface features of sentence v. We can define the empirical estimates $f_\theta^{n_t}(\mathbf{x})$, $f_\theta^{m_t}(\bar{\mathbf{x}})$ of the importance functions $f_\theta^p(\mathbf{x})$ and $f_\theta^q(\bar{\mathbf{x}})$ as:

$$\begin{aligned} f_\theta^{n_t}(\mathbf{x}) &= \frac{f_\theta(\mathbf{x})}{\sum_{\mathbf{x}' \in V^t} f_\theta(\mathbf{x}')} \cdot n_t \\ f_\theta^{m_t}(\bar{\mathbf{x}}) &= \frac{f_\theta(\bar{\mathbf{x}})}{\sum_{\bar{\mathbf{x}}' \in S^t} f_\theta(\bar{\mathbf{x}}')} \cdot m_t \end{aligned} \tag{5.3}$$

where $n_t = |\mathbf{X}^t|$ is the number of sentences and $m_t = |\mathbf{X}^t|$ is the number of summary sentences in topic t.

5.3.4 Training: generic summarization

The parameters θ of the log-linear importance function must be learned from data, so we define a loss function based on weighted MMD. Let $\{(\mathbf{X}^t, \hat{\mathbf{X}}^t)\}_{t=1}^T$ be the T training tuples. Then, the loss of topic t is the square of importance weighted empirical MMD between sentences and summary sentences from within the topic:

$$\mathcal{L}^t = \mathcal{L}(\mathbf{X}^t, \hat{\mathbf{X}}^t, \theta) = \mathrm{wMMD}_{\mathcal{F}}^2(\mathbf{X}^t, \hat{\mathbf{X}}^t, \theta) \tag{5.4}$$

where $\mathrm{wMMD}_{\mathcal{F}}^2(\mathbf{X}^t, \hat{\mathbf{X}}^t, \theta)$ is an empirical estimate of the weighted $\mathrm{wMMD}_{\mathcal{F}}^2(p, q, \theta)$.

Empirical estimate of $\mathrm{wMMD}_{\mathcal{F}}^2(p, q, \theta)$: First, $\mathrm{wMMD}_{\mathcal{F}}^2(p, q, \theta)$ can be expanded as:

$$\begin{aligned} &\mathop{\mathbb{E}}_{\mathbf{x}, \mathbf{x}' \sim p} \left[f_\theta^p(\mathbf{x}) \cdot f_\theta^p(\mathbf{x}') \cdot \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle_{\mathcal{H}} \right] - 2 \cdot \mathop{\mathbb{E}}_{\mathbf{x} \sim p, \bar{\mathbf{x}} \sim q} \left[f_\theta^p(\mathbf{x}) \cdot f_\theta^q(\bar{\mathbf{x}}) \cdot \langle \phi(\mathbf{x}), \phi(\bar{\mathbf{x}}) \rangle_{\mathcal{H}} \right] \\ &+ \mathop{\mathbb{E}}_{\bar{\mathbf{x}}, \bar{\mathbf{x}}' \sim q} \left[f_\theta^q(\bar{\mathbf{x}}) \cdot f_\theta^q(\bar{\mathbf{x}}') \cdot \langle \phi(\bar{\mathbf{x}}), \phi(\bar{\mathbf{x}}') \rangle_{\mathcal{H}} \right] \end{aligned}$$

Applying the kernel trick (Theorem 2.1),

$$\begin{aligned} &\mathop{\mathbb{E}}_{\mathbf{x}, \mathbf{x}' \sim p} \left[f_\theta^p(\mathbf{x}) \cdot f_\theta^p(\mathbf{x}') \cdot k(\mathbf{x}, \mathbf{x}') \right] - 2 \cdot \mathop{\mathbb{E}}_{\mathbf{x} \sim p, \bar{\mathbf{x}} \sim q} \left[f_\theta^p(\mathbf{x}) \cdot f_\theta^q(\bar{\mathbf{x}}) \cdot k(\mathbf{x}, \bar{\mathbf{x}}') \right] \\ &+ \mathop{\mathbb{E}}_{\bar{\mathbf{x}}, \bar{\mathbf{x}}' \sim q} \left[f_\theta^q(\bar{\mathbf{x}}) \cdot f_\theta^q(\bar{\mathbf{x}}') \cdot k(\bar{\mathbf{x}}, \bar{\mathbf{x}}') \right] \end{aligned}$$

Our loss for generic summarization $\mathcal{L}(\mathbf{X}^t, \hat{\mathbf{X}}^t, \theta)$ is $\text{wMMD}^2_{\mathcal{F}}(\mathbf{X}^t, \hat{\mathbf{X}}^t, \theta)$. Recalling $n_t = |\mathbf{X}^t|$ and $m_t = |\hat{\mathbf{X}}^t|$:

$$\begin{aligned}\mathcal{L}^t =& \frac{1}{n_t^2} \sum_{\mathbf{x},\mathbf{x}'} f_\theta^{n_t}(\mathbf{x}) \cdot f_\theta^{n_t}(\mathbf{x}') \cdot k(\mathbf{x}, \mathbf{x}') - \frac{2}{n_t \cdot m_t} \sum_{v,s} f_\theta^{n_t}(\mathbf{x}) \cdot f_\theta^{m_t}(\bar{\mathbf{x}}) \cdot k(\mathbf{x}, \bar{\mathbf{x}}') \\ &+ \frac{1}{m_t^2} \sum_{\bar{\mathbf{x}},\bar{\mathbf{x}}'} f_\theta^{m_t}(\bar{\mathbf{x}}) \cdot f_\theta^{m_t}(\bar{\mathbf{x}}') \cdot k(\bar{\mathbf{x}}, \bar{\mathbf{x}}') \end{aligned} \tag{5.5}$$

Equation 5.5 is the loss for a single topic but during training we will instead minimize the average loss over all topics in the training set, i.e., $\min_\theta \frac{1}{T} \sum_{t=1}^{T} \mathcal{L}(\mathbf{X}^t, \hat{\mathbf{X}}^t, \theta)$. Intuitively, we learn the parameters θ by minimizing an importance weighted distance between sentences and ground truth summary sentences over all topics.

5.3.5 Training: comparative summarization

We now extend the learning task to comparative summarization using the competing binary classifiers idea of Chapter 4 (§4.3.2). Specifically, we replace the accuracy terms in Equation 4.1 with the square of weighted MMD. Given the T comparative training tuples $\{(\mathbf{X}_B^t, \mathbf{X}_A^t, \hat{\mathbf{X}}^t)\}_{t=1}^T$, then the objective is to minimize:

$$\min_{\theta_B, \theta_A} \frac{1}{T} \sum_t \left(\mathcal{L}(\mathbf{X}_B^t, \hat{\mathbf{X}}^t, \theta_B) - \lambda \cdot \mathcal{L}(\mathbf{X}_A^t, \hat{\mathbf{X}}^t, \theta_A) \right) \tag{5.6}$$

Note there are two sets of importance parameters θ_B, θ_A one for each of the two document sets.

5.3.6 Multiple kernel learning (MKL)

We employ Multiple Kernel Learning (MKL) to make use of data from multiple sources such as text, and knowledge base features in our MMD summarization framework. We adapt two stage kernel learning [Cortes et al., 2010], where different kernels are linearly combined to maximize the alignment with the *target kernel* of the classification problem. Since MMD can be interpreted as classifiability [Sriperumbudur et al., 2009a] MKL fits neatly into our MMD based summarization objective. Intuitively, MKL should identify a good combination of kernels for building a classifier that separates summary and non-summary sentences.

Let $\{k_i\}_{i=1}^p$ be p kernel functions, each acting on different input features such as text, knowledge base features, or sentence embeddings. For topic t, let $\mathbf{K}_i^t$ be the kernel matrix according to kernel function k_i, and $\overline{\mathbf{K}}_i^t = \mathbf{U}_{n_t} \mathbf{K}_i^t \mathbf{U}_{n_t}$ be the centered kernel matrix, with $\mathbf{U}_{n_t} = \mathbf{I} - \mathbf{1}\mathbf{1}^\top / n_t$. Let $\mathbf{y}^t = \{\pm 1\}^{n_t}$ be the ground truth summary labels with $y_i^t = +1$ if and only if sentence i belongs to the summary. The *target kernel* $\mathbf{y}^t (\mathbf{y}^t)^\top$ represents the ideal notion

of similarity between sentences. Then, the non-negative kernel weights $\mathbf{w} = [w_1, w_2, \dots w_p]$ which lead to the optimal alignment with the target kernel are given by [Cortes et al., 2010]

$$\min_{\mathbf{w} \geq 0} \mathbf{w}^\top (\mathbf{M}^t)^\top \mathbf{w} - 2\mathbf{w}^\top \mathbf{a}^\mathrm{t}, \tag{5.7}$$

where $\mathbf{M}^\mathrm{t} \in \mathbb{R}^{p \times p}$ has $\mathbf{M}^\mathrm{t}_{rs} = \langle \overline{\mathbf{K}}_r, \overline{\mathbf{K}}_s \rangle_\mathrm{F}$ and $\mathbf{a}^\mathrm{t} \in \mathbb{R}^p$ has $a_i = \langle \overline{\mathbf{K}}^\mathrm{t}_i, \mathbf{y}^\mathrm{t} (\mathbf{y}^t)^\top \rangle_\mathrm{F}$.

The kernel function must be characteristic for MMD to be a valid metric [Muandet et al., 2017]. Most popular kernels used for bag of words like text features (including TF-IDF), the linear kernel ($k(\mathbf{x}, \mathbf{y}) = \langle \mathbf{x}, \mathbf{y} \rangle$) and the cosine kernel ($k(\mathbf{x}, \mathbf{y}) = \frac{\langle \mathbf{x}, \mathbf{y} \rangle}{\|\mathbf{x}\| \|\mathbf{y}\|}$), are not characteristic [Sriperumbudur et al., 2010]. But, the exponential kernel,

$$k(\mathbf{x}, \mathbf{y}) = \exp(\gamma k'(\mathbf{x}, \mathbf{y})), \ \gamma > 0 \tag{5.8}$$

is characteristic for any kernel k' [Steinwart, 2001]. Hence, we use the *normalized exponential kernel* combined with the cosine kernel, $k(\mathbf{x}, \mathbf{y}) = \exp(-\gamma) \exp(\gamma \sum_{i=1}^{p} w_i \cdot \cos(\mathbf{x}^{(i)}, \mathbf{y}^{(i)}))$.

5.3.7 Inference

Given a learned importance function f_θ, we may find the best set of summary sentences $\bar{\mathbf{X}}^\mathrm{t}$ for generic summarization via:

$$\bar{\mathbf{X}}^\mathrm{t} = \operatorname*{Argmax}_{\bar{\mathbf{X}} \in \mathcal{X}^\mathrm{t}} - \mathcal{L}(\mathbf{X}^\mathrm{t}, \bar{\mathbf{X}}, \theta) \tag{5.9}$$

where $\mathcal{X}^\mathrm{t}$ is the set of all valid summaries satisfying the number of words constraints. Similarly, for the comparative task, with learned importance functions, we seek $\bar{\mathbf{X}}^\mathrm{t}$ as:

$$\bar{\mathbf{X}}^\mathrm{t}_B = \operatorname*{Argmax}_{\bar{\mathbf{X}} \in \mathcal{X}^\mathrm{t}_B} (-\mathcal{L}(\mathbf{X}^\mathrm{t}_B, \bar{\mathbf{X}}, \theta_B) + \lambda \cdot \mathcal{L}(\mathbf{X}^\mathrm{t}_A, \bar{\mathbf{X}}, \theta_A)) \tag{5.10}$$

Both these inference problems are budgeted maximization problems, which are often solved by greedy algorithms [Lin and Bilmes, 2010]. The generic unsupervised summarization task is submodular and monotone under certain conditions [Kim et al., 2016], so greedy algorithms have good theoretical guarantees [Nemhauser et al., 1978]. While our supervised variants do not have these guarantees, we find that greedy optimization nonetheless leads to good solutions.

5.4 Experimental setup

We now describe our experimental setup on applying *SupMMD*, and baselines and evaluation setup. We also describe our method for extracting oracles from human written summaries.

			Oracle (ours)			Oracle [Liu and Lapata, 2019b]		
Dataset	**# topics**	**# sents**	**avg summ sents**	**R1**	**R2**	**avg summ sents**	**R1**	**R2**
DUC2003	30	6989	3.73	43.1	17.0	3.40	42.2	16.2
DUC2004	50	12148	4.02	42.0	14.9	3.46	40.6	14.2
TAC2008-A	48	9914	3.90	45.5	19.4	3.42	44.0	18.6
TAC2008-B	48	9147	3.83	44.9	19.5	3.50	43.6	18.7
TAC2009-A	44	9509	4.07	46.9	20.5	3.32	44.5	19.1
TAC2009-B	44	8543	3.61	44.8	19.2	3.27	43.1	18.1

Table 5.1: Dataset statistics and oracle performance. We report the number of topics in each dataset, along with the number of sentences after preprocessing. We show the ROUGE scores of our oracle method and the one by Liu and Lapata [2019b] with average number of sentence in summary from each method.

5.4.1 Datasets

We use four standard multi-document summarization benchmark datasets: DUC-2003, DUC-2004, TAC-2008 and TAC-2009[2]; dataset statistics are provided in Table 5.1. Each of these datasets has multiple topics, where each topic in turn has multiple news articles and four human written summaries. In one setting we use DUC-2003 as the training set and DUC-2004 as test set, and in another setting we use TAC-2008 as the training set and TAC-2009 as the test set – both settings are common in the literature. The DUC datasets can be used for generic summarization while TAC, being an update summarization task, can be used for both generic (set A) and comparative summarization (set B).

5.4.2 Data preprocessing and preparation

The DUC and TAC datasets are provided as collections of XML documents, so it is necessary to extract relevant text and then perform sentence and word tokenization. For DUC we clean the text using various regular expressions the details of which are provided in our code release. We train `PunktSentenceTokenizer` to detect sentence boundaries, and use the standard NLTK [Bird, 2006] word tokenizer. For the TAC dataset, we use the preprocessing pipeline employed by Gillick et al. [2009][3]. This enables a cleaner comparison with the state-of-the-art ICSI [Gillick et al., 2009] method on the TAC dataset. For all datasets, we keep the sentences between 8 and 55 words per Yasunaga et al. [2017].

[2] https://duc.nist.gov/data.html
[3] https://github.com/benob/icsisumm

5.4.3 Feature representations

Our method requires two different sets of sentence features: *text features*, which are used to compute the sentence-sentence similarity as part of the kernel; and *surface features*, which are used in learning the sentence importance model.

5.4.3.1 Text features

Each sentence has three different feature representations: unigrams, bigrams and entities. The unigrams are stemmed words, with stop words from the NLTK english list removed. The bigrams are a combination of stemmed unigrams and bigrams. The entities are DBPedia concepts extracted using DBPedia Spotlight [Mendes et al., 2011]. We use a Term Frequency Inverse Sentence Frequency (TF-ISF) [Neto et al., 2000] representation for all text features. TF-ISF has been used extensively in multi-document summarization [Dias et al., 2007; Alguliev et al., 2011; Wan et al., 2007].

5.4.3.2 Surface features

We use 10 surface features for the DUC dataset, and 12 for the TAC dataset:

position: There are five position features. Four indicators denote the 1st, 2nd, 3rd or a later position of the sentence in the article. The final feature gives the position relative to the length of the article.

counts: There are two count features: the number of words and number of nouns. We use the `spaCy`[4] part of speech tagging to find nouns.

tfisf: This is the sum of the TS-ISF scores for unigrams composing the sentence. For sentence s, this is $\sum_{w \in s} \text{isf}(w) \cdot \text{tf}(w, s)$, where $\text{isf}(w)$ is the inverse sentence frequency of unigram w, and $\text{tf}(w, s)$ is the term frequency of w in s.

btfisf: The boosted sum of TS-ISF scores for unigrams composing the sentence. Specifically, we compute $\sum_{w \in s} \text{isf}(w) \cdot b(w) \cdot \text{tf}(w, s)$, where we boost the score of unigrams w that appear in the first sentence of the article as $b(w)$. In the generic summarization $b(w) = 2$, for comparative summarization $b(w) = 3$, as used by Gillick et al. [2009]. Unigrams that do not appear in the first sentence of the article have $b(w) = 1$.

lexrank The LexRank score [Erkan and Radev, 2004] computed on the bigrams' cosine similarity.

For the TAC datasets, we additionally use:

par_start: An indicator whether the sentence begins a paragraph. This is provided by the preprocessing pipeline from ICSI [Gillick et al., 2009].

[4] https://spacy.io

qsim: The fraction of topic description unigrams present in each sentence; these topic descriptions are only available for TAC.

5.4.4 Oracle extraction

Both DUC and TAC provide four human written summaries for each topic. Since our goal is extractive summarization with supervised training, we need to know which sentences in the articles could be used to construct the summaries in the training set. The article sentences that best match the abstractive summaries are called the *oracles* ($\hat{S}^{\mathrm{t}}$).

Algorithm 2 Oracle extraction

function ExtractOracle($\alpha, V^{\mathrm{t}}, H^{\mathrm{t}}, r, L$)
  $\hat{S}^{\mathrm{t}} \leftarrow \emptyset$
  while $\sum_{s \in \hat{S}^{\mathrm{t}}} \mathrm{len}(s) \leq L$ **do**
    $s^* \leftarrow \underset{s \in V^{\mathrm{t}} \setminus \hat{S}^{\mathrm{t}}}{\mathrm{Argmax}} \frac{\alpha(\hat{S}^{\mathrm{t}} \cup \{s\}, H^{\mathrm{t}}) - \alpha(\hat{S}^{\mathrm{t}}, H^{\mathrm{t}})}{\mathrm{len}(s)^r}$
    $\hat{S}^{\mathrm{t}} \leftarrow \hat{S}^{\mathrm{t}} \cup \{s^*\}$
  return $\hat{S}^{\mathrm{t}}$

Our extraction algorithm (Algorithm 2), is inspired by Liu and Lapata [2019b]. We greedily select sentences (s) which provide the maximum gain in extraction score $\alpha(\hat{S}^{\mathrm{t}}, H^{\mathrm{t}})$ against the human summaries (H^{t}) until a word budget (L) is reached. We only include sentences between 8 and 55 words as suggested by Yasunaga et al. [2017], and set a budget of 104 words to ensure our oracle summaries are within 100 ± 4 words, consistent with the evaluation (§5.4.6).

In contrast to Liu and Lapata [2019b] which uses only ROUGE-2 recall score [Lin, 2004], our method balances both ROUGE-1 and ROUGE-2 recall scores using the harmonic mean and explicitly accounts for sentence length. Grid search on the validation sets shows that the optimal value for r is 0.4 across different datasets and summarization tasks. As reported in Table 5.1, on average our method produces oracles consisting of more sentences and with higher ROUGE-1 and ROUGE-2 scores compared to oracles from Liu and Lapata [2019b]. This is consistent across all datasets.

5.4.5 Implementation details

Supervised variants use an ℓ_2 regularized log-linear model of importance (§5.3.3) trained using the oracles (§5.4.4) as ground truth. We selected the number of training epochs using 5-fold cross validation. We then tune the other hyperparameters on the training set with respect to ROUGE-2 Recall scores. The hyperparameters of the generic summarization task are: γ, a parameter of the kernel; β, the ℓ_2 regularization weight for the log-linear importance function; and r, which defines the length dependent scaling factor in greedy selection [Lin and Bilmes,

2010]. The comparative objective (5.6) has an additional hyperparameter λ, which controls the comparativeness.

We train generic summarization model with full batch LBFGS [Liu and Nocedal, 1989] with learning rate 0.005. We train comparative summarization model using Yogi optimizer [Zaheer et al., 2018], with a mini batch size of 8 topics, learning rate 0.002, and decreasing the learning rate by half every 20 epochs. We set the patience to 20 epochs for early stopping with LBFGS optimizer and 50 epochs with Yogi optimizer. We tune the other hyperparameters on the training set, and the optimal hyperparameters of best model (SupMMD + MKL) and searched space are shown in Table 5.2. The kernel combination weights **w** (§5.3.6) are also shown in Table 5.2. The kernel combination weights (**w**) are written in order: unigrams, bigrams and entities.

hyp.	DUC-2003	TAC-2008-A	TAC-2009-B
γ	2.5[1-4]	4.5[2-6]	2.2[1-3]
β	0.04[.02-.16]	0.08[.02-.16]	0.02[.01-.16]
λ	-	-	0.5[.25-.625]
r	0.001[0-.01]	0.01[0-0.01]	0.01[0-0.01]
epoch	64	53	94
w	[.0, .968, .032]	[.01, .97, .02]	[.014, .98, .006]

Table 5.2: Optimal hyperparameters, their search space and MKL combination weights on each dataset.

5.4.6 Evaluation settings

To evaluate our methods we use the ROUGE [Lin, 2004] metric, the *de facto* choice for evaluating both generic summarization [Hong and Nenkova, 2014; Cho et al., 2019; Yasunaga et al., 2017; Kulesza and Taskar, 2012], and update summarization [Varma et al., 2009; Gillick and Favre, 2009; Zhang et al., 2009; Li et al., 2009b]. ROUGE metrics have been shown to correlate with human judgments [Lin, 2004] in generic summarization task. In previous chapter (§4), we show that human judgments are consistent with the automatic metrics for evaluating comparative summaries.

Both DUC and TAC evaluations use the first 100 words of the generated summary. Our DUC-2004 evaluation setup mirrors Hong et al. [2014]. This allows us to compare performance with the state-of-the-art methods they reported, and other works also evaluated using this setup[5]. As is standard for the DUC-2004 datasets, we report ROUGE-1 and ROUGE-2 recall scores.

For TAC-2009 datasets (both set A and B), we adopt the evaluation settings from the TAC-2009 competition[6], so we can compare against the three best performing systems in the

[5]ROUGE 1.5.5 with args -n 4 -m -a -l 100 -x -c 95 -r 1000 -f A -p 0.5 -t 0

[6]tac.nist.gov/2009/Summarization

competition[7]. As is standard for the TAC-2009 dataset, we report ROUGE-2 and ROUGE-SU4 recall scores.

5.4.7 Baselines

DUC-2004: We select the top performing methods from a recent benchmark paper [Hong et al., 2014] to serve as baselines and report ROUGE scores from the benchmark paper. They are:

ICSI: an integer linear programming method that maximizes coverage [Gillick et al., 2009],

DPP: a determinantal point process method that learns sentence quality and maximizes diversity [Kulesza and Taskar, 2012],

Submodular: a method based on a learned mixture of submodular functions [Lin and Bilmes, 2012],

OCCAMS_V: a method base on topic modeling [Conroy et al., 2013],

Regsum: a method that focuses on learning word importance [Hong and Nenkova, 2014],

Lexrank: a popular graph based sentence scoring method [Erkan and Radev, 2004].

We also include recent deep learning methods evaluated using the same setup as Hong et al. [2014] and report ROUGE scores from the individual papers:

DPPSim: an extension to the DPP model which learns the sentence-sentence similarity using a capsule network [Cho et al., 2019],

HiMAP: a recurrent neural model that employs a modified pointer-generator component [Fabbri et al., 2019], and

GRU+GCN: a model that uses a graph convolution network combined with a recurrent neural network to learn sentence saliency [Yasunaga et al., 2017].

TAC-2009: As baselines for the TAC-2009 dataset we use the top three systems in the TAC-2009 competition for each task, resulting in four systems altogether. To the best of our knowledge these systems are the current state-of-the-art. We report the ROUGE scores from the competition. The systems are:

ICSI: with two variants: *Sys.34* uses integer linear programming to maximize coverage of concepts [Gillick et al., 2009], and *Sys.40*, which additionally uses sentence compression to generate new candidate sentences,

IIT: uses a support vector regressor to predict sentence ROUGE scores [Varma et al., 2009],

ICTCAS: a temporal content filtering method [Zhang et al., 2009], and

ICL: a manifold ranking based method [Li et al., 2009b].

[7] args -n 4 -w 1.2 -m -2 4 -u -c 95 -r 1000 -f A -p 0.5 -t 0 -a -l 100

5.5 Experimental results

We compare our methods with the baselines on the DUC-2004, TAC-2009-A and TAC-2009-B datasets. We present several variants of our method to analyze the effects of different components and modeling choices. We report the performance of unsupervised MMD (*UnsupMMD*) which does not explicitly consider sentence importance. For our supervised method *SupMMD*, we report the performance with a bigram kernel (*SupMMD*) and combined kernels (*SupMMD + MKL*). We also evaluated the impact of our oracle extraction method by replacing it with the extraction method suggested by Liu and Lapata [2019b] in *SupMMD + alt oracles*. Meanwhile, *SupMMD + MKL + compress* presents the result of applying sentence compression [Gillick et al., 2009] to our model.

5.5.1 Generic Summarization

DUC-2004	**R1**	**R2**
ICSI [Gillick et al., 2009]	38.41	9.78
DPP [Kulesza and Taskar, 2012]	**39.79**	9.62
Submodular [Lin and Bilmes, 2012]	39.18	9.35
OCCAMS_V [Conroy et al., 2013]	38.50	9.76
Regsum [Hong and Nenkova, 2014]	38.57	9.75
Lexrank Erkan and Radev [2004]	35.95	7.47
DPP-Sim [Cho et al., 2019]	39.35	10.14
HiMAP [Fabbri et al., 2019]	35.78	8.90
GRU+GCN [Yasunaga et al., 2017]	38.23	9.48
UnsupMMD	35.73	7.76
SupMMD (alt oracle)	39.02	10.22
SupMMD	39.36	10.31
SupMMD + MKL + compress	*39.63*	*10.50*
SupMMD + MKL	39.27	**10.54**

Table 5.3: Results on DUC-2004 generic multi-document summarization task.

The performance of our methods on the DUC-2004 generic summarization task are shown in Table 5.3. On the DUC-2004 dataset all *SupMMD* variants exceed the state-of-the-art, when evaluated with ROUGE-2, and perform similarly to the best existing methods when evaluated with ROUGE-1. Our best system *SupMMD + MKL* outperforms the previous best system (*ICSI*) on ROUGE-2 score by $+\mathbf{3.9}\%$. While the *DPP* baseline achieves the highest ROUGE-1 score on DUC-2004, it has a relatively low ROUGE-2 score which suggests it is optimized for unigram performance at the cost of bigram performance. *SupMMD + MKL* strikes a better balance, scoring the best in ROUGE-2 and second best in ROUGE-1. On the TAC-2009 generic summarization task in Table 5.4 our *SupMMD + MKL* model outperforms the state-of-the-art ICSI model on both ROUGE-2 and ROUGE-SU4. Specifically, *SupMMD + MKL* scores 12.33 in

TAC-2009-A	**R2**	**RSU4**
ICSI(Sys.34) [Gillick et al., 2009]	12.10	*15.09*
ICSI(Sys.40) [Gillick et al., 2009]	*12.16*	15.03
IIIT(Sys.35) [Varma et al., 2009]	10.89	14.49
ICTCAS(Sys.45) [Zhang et al., 2009]	10.64	13.99
UnsupMMD	8.35	11.75
SupMMD (alt oracle)	11.13	14.22
SupMMD	11.76	14.67
SupMMD + MKL + compress	12.02	15.02
SupMMD + MKL	**12.33**	**15.19**

Table 5.4: Results on TAC-2009 generic multi-document summarization task (TAC-2009 set A).

ROUGE-2 while the best ICSI variant scores 12.16 in ROUGE-2.

Supervised modeling: Models using supervised training to identify important sentences substantially outperform the unsupervised method *UnsupMMD*. In fact, *UnsupMMD* is the lowest scoring method across all metrics and datasets. This strongly indicates that a degree of supervision is essential to perform well in this task, and that the importance function is a suitable way to adapt the *UnsupMMD* model to supervised training. Moreover, as show in Table 5.6 we observe a strong correlation between the relative position of a sentence and the score given by *SupMMD*. This observation is consistent with previous works [Kedzie et al., 2018], and demonstrates that *SupMMD* has learned to use the surface features to capture salience.

Oracle extraction: Our oracle extraction technique for transforming abstractive training data to extractive training data helps *SupMMD* methods achieve higher ROUGE performance. An alternative technique developed by Liu and Lapata [2019b] and implemented in *SupMMD (alt oracle)* gives lower performance than our technique. For example, on DUC-2004 *SupMMD (alt oracle)* has a ROUGE-1 of 39.02 and ROUGE-2 of 10.22, while *SupMMD* has a ROUGE-1 of 39.36 and a ROUGE-2 of 10.31. Thus, the advantages of our proposed oracle extraction method are substantial and consistent across multiple datasets and evaluation metrics.

Multiple Kernel Learning: We observe that combining multiple kernels helps the performance of *SupMMD* models on the generic summarization task. *SupMMD + MKL* which combines both bigram and entity kernels has a ROUGE-2 of 10.54 on DUC-2004, while *SupMMD* only uses the bigrams kernel and scores 10.31 in ROUGE-2. Multiple kernels show even clearer gains in the TAC-2009-A dataset.

Sentence compression incorporated into the post-processing steps of *SupMMD + MKL + compress* does not clearly improve the results over *SupMMD + MKL*. On TAC-2009-A, compression clearly reduces performance, and on DUC-2004 *SupMMD + MKL + compress* has a higher ROUGE-1 score but a lower ROUGE-2 score than *SupMMD + MKL*. Incorporating compression into the summarization pipeline is an appealing direction for future work.

5.5.2 Comparative summarization

The results for the comparative summarization task on the TAC-2009-B dataset are shown in Table 5.5. Our supervised MMD variants *SupMMD* and *SupMMD + MKL* both outperform the state-of-the-art baseline ICSI in ROUGE-SU4 but fall short in ROUGE-2. It would be hard to claim that either method is superior in this instance; however, it does show that *SupMMD* – which uses a substantially different approach to that of *ICSI* – provides an alternative state-of-the-art. Thus, *SupMMD* further maps out the set of techniques that are useful for comparative summarization. As per the generic summarization task, both our supervised training method and oracle extraction method are essential for achieving good performance in ROUGE-2 and ROUGE-SU4. We also identify sentence position and btfisf as important features for sentence salience (5.5.4).

Multiple kernels as in *SupMMD + MKL* has relatively little effect, reducing the ROUGE-2 score to 10.24 from the slightly higher 10.28 achieved by *SupMMD*. A similar small decrease is seen for ROUGE-SU4. Manual inspection shows that the summaries from *SupMMD* and *SupMMD + MKL* methods are largely identical with differences primarily on topic D0908, which covers political movements in Nepal. The key entities in this topic are not resolved accurately by DBPedia Spotlight, contributing additional noise and affecting the MKL approach.

Model variants: We have tested an additional variant of our model for comparative summarization, *SupMMD*^{2}, which defines two different importance functions: one for each of the two document sets - A and B (See §5.3.5 for details). In contrast, *SupMMD* has a single importance function shared between document sets, i.e., in Equation (5.6), $\theta_A = \theta_B$. *SupMMD*^{2} performed substantially worse than *SupMMD* in both metrics, for example, *SupMMD* has a ROUGE-2 of 10.28 while *SupMMD*^{2} has a ROUGE-2 of 9.94. We conjecture that a single importance function performs better when training data is relatively scarce because it reduces the number of parameters and simplifies the learning problem. Techniques for tying together the parameters for both importance functions, such as with a hierarchical Bayesian model, are left as future work.

5.5.3 Correlation with ROUGE score

We analyze the correlation between normalized ROUGE recall scores of the sentences and sentence scores from *SupMMD* and *Lexrank*. The normalized rouge score of each sentence is defined as $\overline{\text{ROUGE}}(s) = \frac{\text{ROUGE}(s)}{\#\text{words}(s)}$. As shown in Table 5.7, we find that *SupMMD* has a slightly high correlation with sentence rouge scores. This suggests that *SupMMD* is better in capturing sentence importance for summarization.

TAC-2009-B	**R2**	**RSU4**
ICSI (Sys.34) [Gillick et al., 2009]	**10.39**	13.85
ICSI (Sys.40) [Gillick et al., 2009]	*10.37*	13.97
IIIT (Sys.35) [Varma et al., 2009]	10.10	13.84
ICL (Sys.24) [Li et al., 2009b]	9.62	13.52
UnsupMMD	7.20	11.29
SupMMD (alt oracle)	10.06	13.86
SupMMD^2	9.94	13.76
SupMMD	*10.28*	**14.09**
SupMMD + MKL + compress	10.25	13.91
SupMMD + MKL	10.24	*14.05*

Table 5.5: Results on TAC-2009 comparative multi-document summarization task (TAC-2009 set B).

	DUC2004		TAC2009-A		TAC2009-B	
feature	SupMMD	LexRank	SupMMD	LexRank	SupMMD	LexRank
position	0.34	0.16	0.32	0.18	0.44	0.22
tfisf	0.07	0.38	0.22	0.37	0.01	0.36
btfisf	0.30	0.52	0.48	0.53	0.46	0.57
#words	0.0	0.35	0.08	0.33	-0.15	0.31
#nouns	0.15	0.43	0.27	0.41	0.08	0.40

Table 5.6: Correlation of some features with sentence scores from SupMMD and Lexrank eigenvector centrality.

5.5.4 Feature correlations

We analyze the correlation between various surface features and sentence importance scores from *SupMMD* and *Lexrank* [Erkan and Radev, 2004]. As shown in table 5.6, *SupMMD* has higher correlation with relative position, signifying the importance of position of sentence in summary sentences. *Lexrank* has higher correlations with the number of words, number of nouns and TFISF scores of the sentences, which is expected as *Lexrank* is an eigenvector centrality of sentence-sentence similarity matrix. This suggests *SupMMD* is able to learn that first few sentences are important in news summarization. Similar result is reported by Kedzie et al. [2018], where they show that the first few sentences are important in creating summary of news articles.

	ROUGE-2		ROUGE-1	
dataset	SupMMD	LexRank	SupMMD	LexRank
TAC2009A	0.590	0.555	0.571	0.543
DUC2004	0.595	0.577	0.567	0.545

Table 5.7: Correlation of sentence importance scores with normalized sentence ROUGE scores.

5.5.5 Example summary

We present the update summaries (Set A and B) of topic D0906, which contains articles about *"Rains and mudslides in Southern California"* in Table 5.8. We highlight few phrases in bold which could help us to identify the difference between set A and B. Summaries from *ICSI* and *SupMMD* methods suggest that set A contains articles describing events from earlier days of the disaster and set B contains articles from later stage of the disaster. We can clearly see this from the summaries from both methods – ICSI and SupMMD. The ICSI summary of set A mentions that the disaster is in fourth day, has the highest rain in 40 years and has caused three deaths. Whereas for set B, the ICSI summary mentions about even more rain, the record since 1883-1884, and nine deaths. Similarly, from SupMMD summary of set A, we can see that there has been 15 inch rain, and traffic disruptions. For set B, SupMMD summary includes estimates of damage, and mentions that it has rained for two weeks, and stormed for 6 days. From this, we can see the usefulness of update summarisation methods.

5.6 Conclusion

In this work, we present SupMMD, a novel technique for generic and update summarization (hybrid of generic and comparative summarization as introduced in §1.1) based on the *maximum mean discrepancy*. SupMMD combines supervised learning for salience, and unsupervised learning for coverage and diversity. Further, we adapt multiple kernel learning to exploit multiple sources of similarity (e.g., text features and knowledge based concepts). We show the efficacy of SupMMD in both generic and update summarization tasks on two standard datasets, when compared to the existing approaches. We also show that the importance model we introduce on top of our existing unsupervised MMD (Chapter 4) improves the summarization performance substantially on both generic and comparative summarization tasks. This chapter addresses the second research question (RQ2) of the thesis, which is about developing the supervised methods for comparative summarization.

method	set A	set B
ICSI	**A fourth day** of thrashing thunderstorms began to take a heavier toll on southern California with at least **three deaths** blamed on the rain, as flooding and mudslides forced road closures and emergency crews carried out harrowing rescue operations. Downtown Los Angeles has had more than 15 inches of rain since Jan. 1, more than its average rainfall for an entire year, including 2.6 inches, a record. Meteorologists say Southern California has not been hit by **this much rain in nearly 40 years**. The disaster was the latest caused by **rain and snow** that has battered California since Dec. 25.	Californians braced for **even more rain** as they struggled to recover from storms that have left at least **nine people dead**, triggered mudslides and tornadoes, and washed away roads and runways. The **record**, 38.18 inches (96.98 centimeters), was set in **1883-1884**. **Mudslides** forced Amtrak to suspend train service between Los Angeles and Santa Barbara through at least Thursday. A **winter storm** pummeled Southern California for the third straight day, claiming the lives of three people and raising fears of mudslides, even as homes around the region were evacuated. Staff Writers Rick Orlov and Lisa Mascaro contributed to this story.
SupMMD	Downtown Los Angeles has had more than **15 inches of rain** since Jan. 1, more than its average rainfall for an entire year, including 2.6 inches, a record. A **fourth day** of thrashing thunderstorms began to take a heavier toll on southern California with at least three deaths blamed on the rain, as **flooding and mudslides** forced road closures and emergency crews carried out harrowing rescue operations. The roads in Los Angeles County were equally frustrating. Part of a rain-saturated hillside gave way, sending a Mississippi-like **torrent of earth and trees** onto four blocks of this oceanfront town and killing two men.	Storms have caused **$52.5 million** (euro39.8 million) in damage to Los Angeles County roads and facilities since the beginning of the year. Multi-million-dollar homes collapsed and mudslides trapped residents in their homes as a heavy rains that have claimed three lives pelted Los Angeles for the fifth straight day. In scenes reminiscent of the aftermath of the Northridge Earthquake 11 years ago this month, Los Angeles area residents faced gridlocked freeways and roads Wednesday while cleanup crews cleared mud, rubble and debris left from a **two-week siege of rain**. A **record-shattering storm** slammed Southern California for a **sixth straight day** Tuesday, triggering **mudslides and tornadoes** and forcing more road closures, but forecasters predicted it would wane Wednesday before a new storm moves in Sunday night.

Table 5.8: Example summaries of topic D0906, containing articles about "Rains and mudslides in Southern California".

Chapter 6

Applicability of Comparative Summarization

> *The quality of democracy and the quality of journalism are deeply entwined.*
>
> Bill Moyers, *Nat'l Conf. for Media Reform*, 2005

In this chapter, we apply the unsupervised comparative summarization methods developed in Chapter 4 to a range of tasks in two datasets with a goal of determining to what extent such techniques are effective, hence answering Research Question 4 (RQ4) of the thesis. We start by defining two metrics – *distinguishability* and *summarizability*, that help us to quantify the applicability of comparative summarization to different datasets. Given two groups of documents, distinguishability measures the amount by which these two groups are comparable, whereas summarizability quantifies the usefulness of comparative summaries for these groups. We measure these metrics in a newly curated News2019+Bias dataset (§3.4.1) in comparing articles *over time*, and *across ideological leanings* (or ideological biases) of media outlets. First, we observe that the summarizability is proportional to the distinguishability, and identify the groups of articles that are less or more distinguishable. Second, better distinguishability and summarizability is amenable to the choice of document representations according to the comparisons we make, either *over time*, or *across ideological leanings* of media outlets. Finally, we apply the comparative summarization method to the task of comparing stances in the social media domain. We make the software and dataset publicly available[1].

6.1 Introduction

Comparative summarization problem is a way of comparing different groups of documents in a collection. It does so by summarizing different groups of document collections, with a goal

[1] https://github.com/bistaumanga/compsumm-applicability

of producing summaries that contain the salient information of each document group and discriminate against the other document groups. In the previous chapters, we formally introduced the problem of comparative summarization, developed supervised and unsupervised algorithms, and conducted extrinsic evaluations in unsupervised settings (§4.6.1).

Comparative summarization has been used to compare news collections over time [Duan and Jatowt, 2019; Huang et al., 2011; Bista et al., 2019, 2020], and compare patents [Zhang et al., 2015]. The underlying assumption in this literature is that the groups of documents are comparable. No previous work in the comparative summarization literature has explored in quantifying the degree of comparativeness between document groups. However, doing so is important because it allows us to determine the extent to which the comparative summarization algorithms are applicable to the groups of documents we are comparing. In this work, we define two measures – *distinguishability* and *summarizability*, which takes values in between 0 and 1, inclusive. *Distinguishability* quantifies the amount by which the different document groups in a collection are comparable. *Summarizability* quantifies the amount by which the distinguishability between document groups is retained by a small number of prototypes. We can view summarizability as a measure of usefulness of the comparative summaries. The research question we seek to address here is: **Q1. Which groups of documents are distinguishable and summarizable?** In particular, which group pairs of news articles are comparable?

Second, different feature representations such as bag of words and document embeddings techniques can be used in measuring distinguishability and summarizability. A common perception in the NLP literature is that embedding techniques such as BERT does better than traditional bag of words like features [Devlin et al., 2019] in many NLP tasks including document classification. The research question we seek to address here is: **Q2. How do the feature representations affect the measures according to the ways we compare?**

We measure distinguishability and summarizability in three tasks in two datasets, with a goal of quantifying the applicability of comparative summarization. We next introduce and briefly discuss the main results obtained in each of the two datasets, and the role comparative summarization plays in these datasets.

6.1.1 Comparing news articles over time and across ideological leanings

News2019+Bias dataset that we introduced in §3.4 has two sets of groups – *publication month* and *ideological leanings* of media outlets. **Ideology** is defined as "a set of beliefs or principles, especially one on which a political system, party, or organization is based" according to Cambridge dictionary (2021b). Ideological biases exist in news-media in several forms [Mullainathan and Shleifer, 2002; Baron, 2006; D'Alessio and Allen, 2000], and shape the political polarization [Bernhardt et al., 2008] and public opinions on important topics [Boykoff and Boykoff, 2004]. Hence, studying media-bias is an important problem in political and social

science and computers can aid in the study by answering questions with machine learning.

We measure the distinguishability and summarizability in News2019+Bias dataset in the tasks of comparing the news articles over months and across ideological leanings and make the following two main observations:

1. Summarizability is proportional to the Distinguishability. Summarizability is behind distinguishability by a small fraction, suggesting that the differences between different document groups (over time, or across ideological leanings) can be evident by just looking at few representative prototypes.

2. By analyzing the distinguishability of different comparison tasks, we can categorize the comparisons as the least-distinguishable, distinguishable and the most-distinguishable. This help us to quantify the amount of coverage change over time, differences in ideologies on important topics.

3. Sentence embeddings features are useful in comparing articles across ideologies, whereas bag-of-words features are useful in comparing over time.

6.1.2 Comparing stances in social media

Stance is defined as "a way of thinking about something, especially expressed in a publicly stated opinion" by Cambridge Dictionary (2021c). With the advent of social media platforms like Twitter, politicians and active individuals have been using it to share their beliefs and opinions, and react to other people's opinion towards various topics. This trend has prominently emerged since the 2008 US presidential election [Johnson and Goldwasser, 2016]. Members of the public and politicians often express opposing and polarizing viewpoints on different controversial social, political, and economic issues in social media. Stance detection is an important task because it helps analyze the public opinion towards controversial issues. A benchmark stance detection task of social media posts (tweets) with dataset is provided by Mohammad et al. [2016]. They provide over 3000 labeled tweets on five different topics/issues for supervised stance detection.

The role of comparative summarization in social media is to help quantify and understand the differences among different stances on important socio-economic topics. We apply comparative summarization to this stance detection dataset, from which we obtain a few representative prototype tweets. We observe that while the tweets in a topic are distinguishable between opposing stances, but their summarizability is poor. This is because, the automatic measure summarizability suffers due to short texts on Twitter, while summaries still making sense to human judges.

6.2 Literature review

We now briefly review the different tasks and methods on each of the two datasets.

6.2.1 Ideological bias in news media

Media bias in news media : While media outlets should be in principle independent, objective and free from any ideological biases, this is not the case in reality [Mullainathan and Shleifer, 2002; Baron, 2006]. Media bias can take several forms such as gatekeeping bias – stories are selected or dis-selected based on ideology, coverage bias – overreporting or underreporting certain stories, statement bias – presenting the stories favoring certain party [D'Alessio and Allen, 2000]. Media bias has been identified to be shaping political polarization [Bernhardt et al., 2008], and divergence of public discourse from scientific discourse in regard to climate change [Boykoff and Boykoff, 2004]. Examples of media bias in recent COVID-19 pandemic include the contrastive coverage in US and Chinese media [AlAfnan, 2020] or, criticizing the lockdowns in Victoria, Australia by conservative mainstream media [Graham et al., 2020].

Detecting media bias : The literature studying media bias goes back the decades into the last century [White, 1950; Williams, 1975], including television coverage in the 1972 US presidential election [Hofstetter, 1976]. In the social sciences, researchers study media bias by analyzing contents (news coverage) and frames (perspectives portrayed in the news articles), or performing meta-analysis of several other studies [Hamborg et al., 2019]. A more detailed overview of interdisciplinary study of news media bias is provided by Hamborg et al. [2019].

Machine learning based approaches to detect bias of the media outlets [Patricia Aires et al., 2019; Stefanov et al., 2020; Baly et al., 2019] and the articles [Baly et al., 2020; Chen et al., 2020b,a] have gained attention recently. Patricia Aires et al. [2019] used community detection on a hyperlink graph to identify outlets with the same political leanings. Alternatively, media bias can be detected from social media data [Stefanov et al., 2020], or by joint modeling of factuality and bias using information from multiple sources such as sample articles, Wikipedia and Twitter metadata and web based features [Baly et al., 2019]. Our work matches more with article level media-bias detection. Baly et al. [2020] used triplet loss with adversarial adaptation to detect biases of news articles when an article level annotated corpus is available. Chen et al. [2020b,a] have developed methods to detect and analyze biases at different levels of granularity within an article.

While a number of works exists in classifying/detecting the ideological bias of news articles, there is no work in summarizing different ideological biases. We focus on measuring the classifiability of articles across different ideological leanings of the outlets and over pub-

lication months with the goal of identifying the distinguishable comparisons, and the feature representations that facilitate better classifiability in each scenario. Measuring the classifiability aids in quantifying the degree of applicability of comparative summarization methods we developed in Chapter 4.

6.2.2 Stance detection in social media

Stance detection is the task of automatically classifying the stance of a piece of text or a user in social media with respect to an issue or a target [Mohammad et al., 2016]. A *target* can be a *public policy*, e.g. policy on climate change, death penalty, etc., *a movement*, e.g. Black Lives Matter, *a product*, e.g. an electric vehicle, or *a person*, e.g. Hillary Clinton, Donald Trump, etc. An example of stance classes are 'FAVOR', 'AGAINST' and 'NONE' towards a target. Political ideology is a combination of stances on different issues. Stance detection is different from sentiment analysis, a related task in social media, which is about determining the emotional polarity of a piece of text [AlDayel and Magdy, 2020].

Earlier works in stance detection focused on political discussions in online forums [AlDayel and Magdy, 2020]. Stance detection has been gaining interest in social media research in last few years. A variety of works have tackled the problem of detecting stances of social media users [Aldayel and Magdy, 2019; Johnson and Goldwasser, 2016; Darwish et al., 2020; Stefanov et al., 2020; Dong et al., 2017; Lahoti et al., 2018], or from social media posts [Zarrella and Marsh, 2016; Wei et al., 2016; Ebrahimi et al., 2016; Augenstein et al., 2016; Sun et al., 2018; Dey et al., 2018]. A more detailed overview of other ways to classify stance detection tasks is reviewed by AlDayel and Magdy [2020].

Mohammad et al. [2016] provided first benchmark datasets in the SemEval 2016 stance detection competition. This has spawned a variety of work in detecting stances from social media posts. Examples include sequence modeling with recurrent neural networks [Zarrella and Marsh, 2016], convolutional neural network [Wei et al., 2016], bidirectional conditional encoding [Augenstein et al., 2016], long term short term (LSTM) networks with attention [Dey et al., 2018], joint sentiment-target-stance modeling [Ebrahimi et al., 2016], and hierarchical attention to combine text and linguistic features [Sun et al., 2018]. Labeling a large scale dataset for such tasks is costly, hence researchers often leverage unsupervised pretraining [Zarrella and Marsh, 2016; Augenstein et al., 2016], and, weak supervision [Zarrella and Marsh, 2016; Augenstein et al., 2016] to improve the performance of models.

An alternative is to detect stances of users in social media. This task can make use of multimodal information such as content and network features [Darwish et al., 2020; Dong et al., 2017; Lahoti et al., 2018; Aldayel and Magdy, 2019], unlike stance detection from a social media post which just has a piece of text available. Content features can be textual such as tweet text and hashtags, whereas network features can be social interactions such

as retweets, mentions and replies, and connections such as friends and followers networks. Network features are found to be more useful than content features [Aldayel and Magdy, 2019]. Network features are often based on homophily, e.g. users with similar stance forms a well partitioned retweet graph [Rajadesingan and Liu, 2014], or heterophily, e.g. users responds to opposing views [Trabelsi and Zaiane, 2018]. A variety of models have leveraged such content and/or network features in supervised settings with linear classifiers [Aldayel and Magdy, 2019], in semi-supervised settings with label propagation [Rajadesingan and Liu, 2014], and in unsupervised settings using non-negative matrix factorization [Lahoti et al., 2018], clustering [Darwish et al., 2020; Stefanov et al., 2020]. Some authors additionally make use of weak supervision to leverage large scale unlabeled data [Johnson and Goldwasser, 2016; Dong et al., 2017].

We focus on comparative summarization of stances in social media, i.e. selecting a few prototype tweets favoring and opposing an issue. To our best knowledge, this problem has not been explored before. We use the SemEval 2016 stance detection dataset [Mohammad et al., 2016], and qualitatively show that summaries are useful.

6.3 Distinguishability and Summarizability

In order to study the applicability of comparative summarization, we define the two automatic metrics *distinguishability* (ρ_D) and *summarizability* (ρ_S). Given a document collection with two, or more groups ($\mathbf{X} = \mathbf{X}_A \cup \mathbf{X}_B$), and the labels $\mathbf{y}$ indicating the groups of documents. We first split the datasets into train and test set ($\mathbf{X}_{tr}, \mathbf{y}_{tr}, \mathbf{X}_{te}, \mathbf{y}_{te}$). We then compute the metrics as a classification performance (measured with balanced accuracy) in test dataset. The metrics take a value between 0 and 1, and higher value means better *distinguishability* or *summarizability*. These metrics help us to quantify the applicability of comparative summarization methods to different datasets and tasks. We compute these metrics in three comparison tasks in two datasets. In the News2019+Bias dataset, we compare articles over publication months (e.g. articles from Jan vs Feb on Climate change topic) and across the ideological leanings of the outlets that produce them (e.g. articles from Left vs Right leaning in Gun control topic) (§6.4). In the stance detection dataset, we compare different stances to the target entity (e.g. pro vs against Abortion) (§6.6). We now define each of the two metrics as:

Distinguishability (ρ_D): According to the Cambridge dictionary, *distinguishable* means "different or separate from other things or people in a way that is easy to notice or understand" [Dictionary, 2021a]. We define the *distinguishability* between the groups of documents to compare as the classification performance (e.g. balanced accuracy) on test set by the classifier trained on entire training set. Distinguishability measures how easy it is for a binary

classifier to discriminate between two document groups. If two groups of documents are distinguishable, a classifier should be able to effectively learn to separate it from training set, and vice-versa. The classification performance on test set reflects the degree of easiness or difficulty is it for the classifier to discriminate two groups. Hence, *distinguishability* quantifies the amount by which different document groups are distinguishable in a comparison.

Summarizability (ρ_S): We define *summarizability* between two groups of documents to compare as the classification performance (e.g. balanced accuracy) on the test set by the classifier trained only on the prototypes. The number of prototypes can vary, e.g. 4, 8, 16, 32, etc., and are obtained from training set. If the groups can be effectively summarized by a few number of prototypes, the classification performance on test set by the classifier trained only on prototypes would reflect this effectiveness. The classification performance on test set indicates how easy it is to discriminate between document groups looking at the summaries only. Hence, *Summarizability* is a measure of usefulness of the comparative summaries. It is based on the automatic extrinsic evaluation of the prototypes (§4.6.1).

Algorithm 3 Pseudocode for computing *distinguishability* and *summarizability*

function DISTINGUISHABILITY($\mathbf{X}, \mathbf{y}$)
  $a \leftarrow 0$
  for $i \leftarrow 1$ to N **do**
    $\mathbf{X}_{tr}, \mathbf{y}_{tr}, \mathbf{X}_{te}, \mathbf{y}_{te} \leftarrow$ train_test_split($\mathbf{X}, \mathbf{y}$, $stratify = \mathbf{y}$, $test\_size = 0.25$, $seed = i$)
    $g^D_\theta \leftarrow$ train_classifier($\mathbf{X}_{tr}, \mathbf{y}_{tr}$)
    $a \leftarrow a +$ balanced_acc($g^D_\theta(\mathbf{X}_{te}), \mathbf{y}_{te}$)
  $\rho_D \leftarrow \frac{a}{N}$
return ρ_D

function SUMMARIZABILITY($\mathbf{X}, \mathbf{y}, m$)
  $a \leftarrow 0$
  for $i \leftarrow 1$ to N **do**
    $\mathbf{X}_{tr}, \mathbf{y}_{tr}, \mathbf{X}_{te}, \mathbf{y}_{te} \leftarrow$ train_test_split($\mathbf{X}, \mathbf{y}$, stratify $= \mathbf{y}$, test_size $= 0.25$, $seed = i$)
    $\hat{\mathbf{X}}, \hat{\mathbf{y}} \leftarrow$ select_prototypes($\mathbf{X}_{tr}, \mathbf{y}_{tr}, m$)
    $g^S_\theta \leftarrow$ train_classifier($\hat{\mathbf{X}}, \hat{\mathbf{y}}$)
    $\rho_S \leftarrow$ balanced_acc($g^S_\theta(\mathbf{X}_{te}), \mathbf{y}_{te}$)
  $\rho_S \leftarrow \frac{a}{N}$
return ρ_S

We illustrate the pseudocode for computing these two metrics in Algorithm 3. To compute both metrics, we first split the dataset into 75% train and 25% test sets ($\mathbf{X}_{tr}, \mathbf{y}_{tr}, \mathbf{X}_{te}, \mathbf{y}_{te}$) using stratified sampling on the group labels. For *distinguishability*, we then train a classifier g^D_θ on training set ($\mathbf{X}_{tr}, \mathbf{y}_{tr}$) and compute the balanced accuracy on test set ($\mathbf{X}_{te}, \mathbf{y}_{te}$). We take

the average balanced accuracy across N random splits as the *distinguishability* metric. Similarly, for *summarizability*, we train a classifier g_θ^S on prototypes obtained from training set $(\hat{\mathbf{X}}, \hat{\mathbf{y}})$ and compute the balanced accuracy on test set $(\mathbf{X}_{te}, \mathbf{y}_{te})$. We take the average balanced accuracy across N random splits as the *distinguishability* metric. We provide further details of experimental setup in §6.4.3.

We define distinguishability and summarizability in terms of a classifier performance (e.g. balanced accuracy) of the best possible classifier that separates the document groups we are comparing. Another possible candidate could be MMD, which is related to the classifiability. MMD is a distance metric between two sets of datapoints (e.g. two groups of documents) for certain choices of kernel functions [Gretton et al., 2012a]. Sriperumbudur et al. [2009a] showed that MMD is inversely proportional to the margin of hard-margin SVM, thus establishing the link between the distance between sets of datapoints and classifiability. While MMD is related to the classifiability, the classifier performance is easier to deal with when measuring across different tasks. It always exists in range $[0, 1]$, while MMD measure can vary with different choice of kernel and kernel parameters. Hence, we choose classifier performance to quantify the distinguishability and summarizability.

It is to be noted that the measures can be interpreted comparatively, not linearly, i.e., a comparison A-B with distinguishability of 0.8 is better than a comparison C-D distinguishability of 0.4, but it does not mean A-B is doubly distinguishable than C-D comparison. Also, choice of feature or classifier should not matter, as long as they are powerful enough to reveal comparative differences.

6.4 Experiments on News2019+Bias dataset: Setup

In this section we describe the experiments we did to measure the *distinguishability* and *summarizability* on the News2019+Bias dataset. As introduced in §3.4.2, we have a document collection of 5 different topics in the dataset. Each topic has documents with two type of labels – 9 *publication months*, and 6 *ideological leanings* of the media outlet. Using this dataset, we measure *distinguishability* and *summarizability* for comparisons *over publication month* and *across ideological leaning*. As an example, article statistics for Climate change topic is shown in Table 6.1.

6.4.1 Comparisons over different groups

We now describe the two comparison settings we choose – comparing over time and across ideological leanings. We choose month as a unit of time period over days, weeks or quarters because choosing month provide us a good number of comparisons with enough data volume, that are just beyond what a person is willing to read, but not too diverse that summarization

	ER	R	RC	C	LC	L	Total
2019-Jan	124	359	514	698	2010	623	4328
2019-Feb	147	389	550	821	2165	659	4731
2019-Mar	135	490	612	1017	2707	757	5718
2019-Apr	84	480	558	825	2405	648	5000
2019-May	100	442	583	877	2466	713	5181
2019-Jun	113	335	499	834	2313	696	4790
2019-Jul	98	338	436	822	2346	617	4657
2019-Aug	164	426	519	946	2613	750	5418
2019-Sep	319	686	671	1078	2906	887	6547
Total	1284	3945	4942	7918	21931	6350	46370

Table 6.1: Count *over publication month* and *across ideological leanings* for *Climate change* topic within the News2019+Bias dataset. The two highlighted groups shows an example of comparing over publication months (within the same leaning) and comparing over ideological leanings (with the same time period).

with a few prototypes stops making sense anymore.

Publication months : We compute the *distinguishability* and *summarizability* for comparisons over *publication month* for each of the 5 topics in each of the 6 ideological leanings. We have altogether $6 \times 5 \times \binom{9}{2} = 1080$ comparison pairs due to 9 months of data. An example of such a comparison in *Climate change* topic is highlighted in Table 6.1, where we compare the articles produced by Right center leaning outlets for March vs June.

Ideological leanings : We compute the *distinguishability* and *summarizability* for comparisons over *ideological leanings* for each of the 5 topics in each of the 9 publication months. We have altogether $9 \times 5 \times \binom{6}{2} = 675$ comparison pairs due to 6 ideological leanings' data. An example of such comparison in *Climate change* topic is highlighted in Table 6.1, where we compare the articles produced in January for Right vs Left center leaning outlets.

6.4.2 Hops in comparisons

We treat *publication month*, and *ideological leaning* as ordinal variables. We assume the ordering in *ideological leanings* to be ER, R, RC, C, LC, L, and EL. This is because we assume the political spectrum to be linear for simplicity, even though non-linear political spectrums such as horseshoe theory has been proposed in the political science literature[2]. The orderings of *publication month* are just the natural order of the months according to the Gregorian calendar. Now we define *hops in comparisons*, which measures how far the groups we are comparing are, within the spectrum they lie.

[2]https://en.wikipedia.org/wiki/Horseshoe_theory

Definition 6.1 (Hops in comparison)**.** We define hops in comparison as the absolute difference between two groups, when groups are ordinal variables.

For example, in comparing across ideologies, $\text{hop}(LC - R) = 3$ in Table 6.1, whereas in comparison over publication months, $\text{hop}(Jun - Mar) = 3$ in Table 6.1. Defining the hops in comparisons allows us to analyze the relation between the measurement variables: *distinguishability* and *summarizability*, and hops itself. We expect the measurement variables to be correlated with the hops, as hops quantify the amount by which the groups of documents are apart according to ideological spectrum or publication time.

6.4.3 Experimental settings

We repeated each of 1080 comparisons over publication months, and 675 comparisons across ideological leanings over 10 different random train-test splits. We use two types of text representations – Bag-of-words (BoW) and Sentence-BERT (SBERT) embedding [Reimers and Gurevych, 2019], extracted from the title and first three sentences. News articles are often written such that the first three sentences contains the key information, a phenomenon known as the inverted pyramid in journalism [Pöttker, 2003]. For Bag-of-words features, we set the minimum document frequency to 2, and maximum document frequency to 0.2 times the dataset size, and maximum features to 2000 on each classification task's data subset. We then normalize the bag-of-words vectors to have a unit ℓ_2 norm.

In case of Sentence-BERT embeddings, we sum the embeddings of the title and the three sentences, and normalize them to have a unit ℓ_2 norm. Reimers and Gurevych [2019] provide a variety of pretrained sentence embeddings (SBERT) models. They take the base models such as GLOVE word embeddings [Pennington et al., 2014], or contextual word embeddings such as RoBERTa- [Liu et al., 2019], BERT [Devlin et al., 2019], DistilBERT [Sanh et al., 2019], etc., and represent the sentence using pooling. Then the pooled embeddings are fine-tuned using a Siamese network with triplet loss [Schroff et al., 2015] on Natural Language Inference (NLI), Semantic text Similarity (STS), or paraphrasing tasks. Among the various models provided by Reimers and Gurevych [2019], we use paraphrase-distilroberta-base-v1 as it gave us the best performance. This is a distilled version [Sanh et al., 2019] of RoBERTa base model [Liu et al., 2019][3], fine-tuned on paraphrase data. In our corpus containing news articles from multiple media sources often reporting same story/event, the embedding model fine-tuned on paraphrase data (paraphrase-distilroberta-base-v1) helps the downstream classifier to better identify the articles covering the same story/event. We also considered stsb-roberta-large (base model RoBERTa fine-tuned on STS and NLI dataset), stsb-berta-large (base model BERT fine-tuned on STS and NLI dataset), average_word_embeddings_glove.6B.300d (averaging GLOVE word embeddings), and distilbert-base-nli-mean-tokens (base model DistilBERT fine-tuned on

[3] https://huggingface.co/distilroberta-base

NLI dataset). All models we considered use mean pooling to get sentence embeddings from word embeddings.

We select prototypes using the unsupervised method because, we do not have ground truth summaries to apply our supervised methods. Among the unsupervised methods introduced in Chapter 4, we use $\mathcal{U}_{diff}(\cdot)$ (Equation 4.3) as it outperforms $\mathcal{U}_{div}(\cdot)$ (Equation 4.4) in our preliminary experiments consistently. We measure the *distinguishability* and *summarizability* using Balanced Accuracy [Brodersen et al., 2010], as the dataset is often imbalanced like the one for Climate change as shown in Table 6.1. We use SVM as the classifier [Cortes and Vapnik, 1995] in both distinguishability and summarizability. For summarizability, we choose four different number of prototypes in measuring the summarizability – 4, 8, 16, and 32, and use exponential kernel (Equation 5.8) with SVM. But, the choice of classifier should not matter as we observed in §4.6.4, where using both 1NN and SVM classifier provided similar results. For distinguishability, we use linear kernel SVM as it can be optimized quickly for a large dataset. The hyperparameters are SVM soft margin C, and kernel-parameter γ, and are chosen by grid-search using three-fold cross-validation. Next, we analyze and discuss the key results we obtain from our experiments on News2019+Bias datasets.

6.5 Experiments on News2019+Bias datasets: Results and discussions

Here, we describe our key observations from the measurement of distinguishability and summarizability across ideological leanings and over publication months. This help us in identifying the comparisons that are worth conducting, and also see which features are useful in these two comparison scenarios. Finally, we see some example summaries that help us in explaining the differences between the groups. But, before going into analysis of the results obtained, one could argue the usefulness of MMD summarization method we use over the random prototype selection baseline for each group. In Figure 6.1 we plot the distribution of difference between summarizability scores from our MMD method and random prototype selection baseline using violin plot and line-range showing mean and standard deviation. Across different number of prototypes, feature choices, and comparisons (across time and over ideological leanings), we see that our MMD method is consistently better in classifying the unseen test documents. The average difference is between 4.3% to 6.6%, and most of the time the difference is greater than 0. This show us that prototypes from our method are better than random extracts in representing the groups we are comparing. Now we present the detailed results.

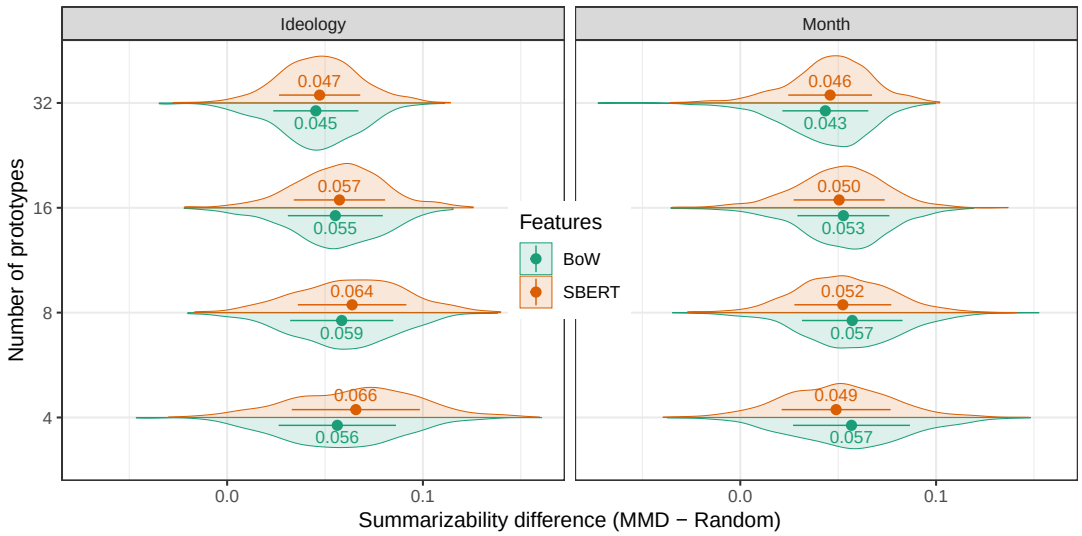


Figure 6.1: Summarizability difference between the prototype selected by our MMD based method and random prototype selection baseline using both BoW and SBERT features for each of the four different number of prototypes.

6.5.1 Summarizability and Distinguishability

Figure 6.2 shows the *summarizability* and *distinguishability*, when measured in comparing over time and across ideological leanings of the media outlets using SBERT and BoW features. The *summarizability* is measured with 8 prototypes. Figure A.1 shows the same plot in comparing over publication months. We observe that *the summarizability is proportional to the distinguishability*. As seen in Fig 6.2, the summarizability is a few points below the distinguishability, suggesting that the differences among the groups can be identified from a few prototypes. This is consistent with the evaluations of Mani et al. [1999], as we discussed in §4.6.1. Hence, we conclude that the prototypes are useful in identifying the differences, and may facilitate in understanding the differences among the document groups.

The second, but obvious observation from Figure 6.3 is that summarizability *improves with increasing number of prototypes* in comparing across ideologies. This is evident because the difference between distinguishability and summarizability decreases, and the correlation between these two measures improves as the number of prototypes increases. This is expected as the performance of the classifier trained on only prototypes improves as the number of prototypes increases. We observe a similar phenomenon in comparing over publication months as well (see Figure A.1). Next, we see which of these comparisons are less or more distinguishable and the relation between hops and distinguishability.

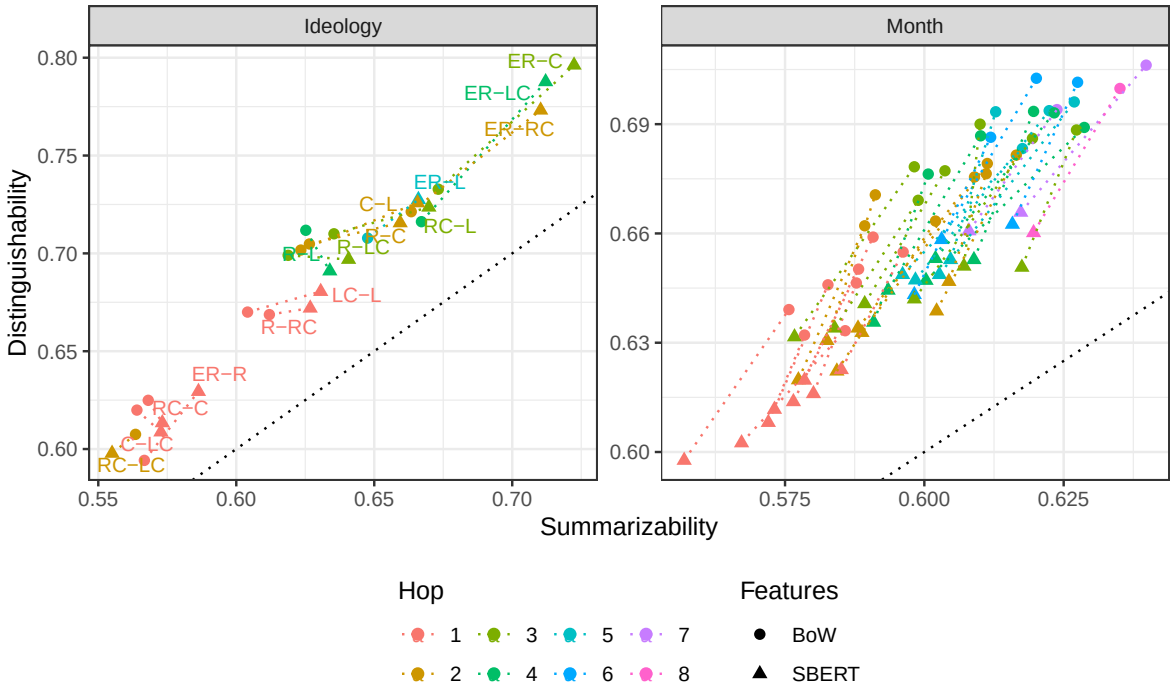


Figure 6.2: Summarizability (number of prototypes = 8) vs Distinguishability. Each point on month comparison is averaged over $6 \times 5 \times 10 = 300$ comparisons, and on ideology comparison is averaged over $9 \times 5 \times 10 = 450$ comparisons. Two different feature representations - BoW and SBERT are denoted by different points shape, and connected by dotted line for same comparison. We denote the hops by color, and black dotted line is the line with slope 1.

6.5.2 Degree of distinguishability

We can categorize the comparisons into three clusters of *least distinguishable, distinguishable and most distinguishable*. In ideology comparisons with SBERT features, as seen in Fig 6.2 and Fig 6.3, comparing the articles from Extreme right (the most biased) outlet with the articles from the least biased sources, i.e., Left center, Center and Right center outlets are the *most distinguishable*. Comparing the articles from Extreme right outlets with articles from Right outlets, and in-between the least biased outlets (Left center, Center, and Right center) is *least distinguishable*. Every remaining comparison is in between these, and can be categorized as *distinguishable*. In comparing over time, we observe that the *extreme right content is the least distinguishable over time* compared to other ideologies as seen in Figure 6.4. We hypothesize that this is because extreme right news content changes the least over time. This categorization help us to quantify the amount of coverage change over time, differences in ideologies on important topics.

Recall that hops in comparing over time is the absolute difference in month number. In comparing articles over time, as seen in Figure 6.4, distinguishability increases with hops for

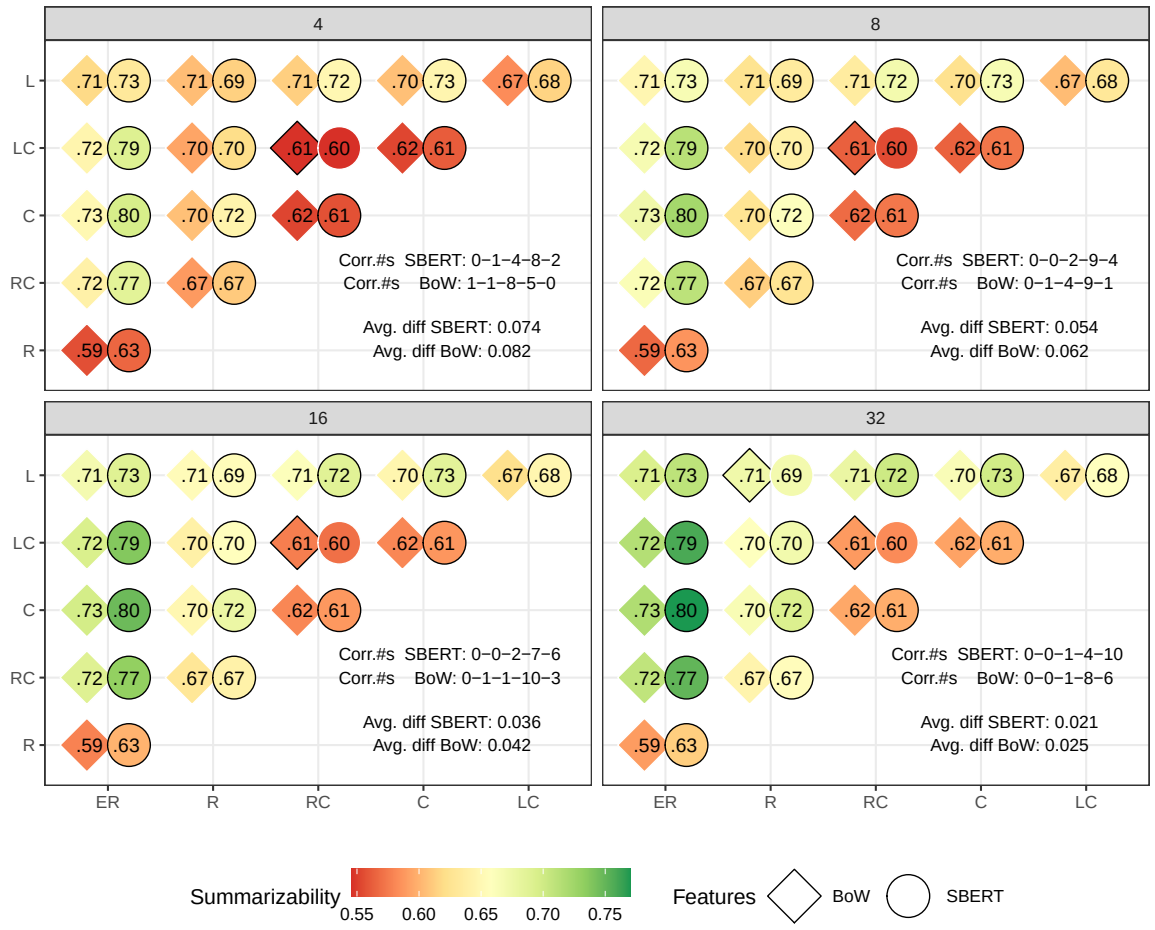


Figure 6.3: Ideology comparison results when viewed on 2D matrix over ideology spectrum for each of 4 different number of prototypes (4, 8, 16, 32). Numbers inside the bubbles are *distinguishability*, whereas *summarizability* is denoted by the intensity of color. The shape represents the features representations, and a border is added to the bubble corresponding to the feature that provides better summarizability. The average difference, and the correlation (binned counts) between the distinguishability and the summarizability for each number of prototype settings is annotated. The correlation is binned as: negligible $[-1, .2)$, low $[.2, .4)$, moderate $[.4, .6)$, strong $[.6, .8)$, and very-strong $[.6, .1]$.

all ideologies. It is natural that the distinguishability between articles increases as the months they are produced are further apart. In ideology comparison, we defined hops as the steps in linear ideology spectrum (§6.4.2), and expect the distinguishability to increase with hops. This is clearly not what we observe in Figure 6.3 (the numbers inside the bubbles). This suggests that our assumption of linear ideology spectrum is incorrect. In political science, non-linear ideological spectrums such as horse-shoe theory has been proposed, which might be amenable here. This requires further investigation from multidisciplinary research, with larger and diverse datasets, including articles from extreme-left media outlets. Next, we see

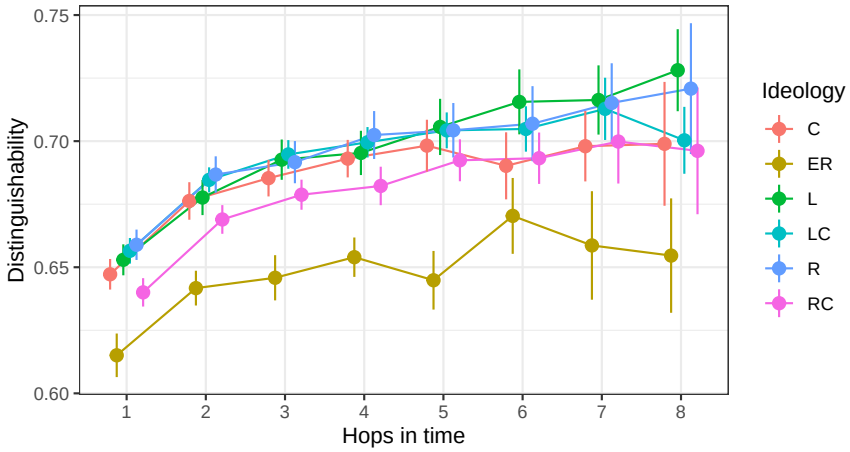


Figure 6.4: Distinguishability of each ideology in comparing over publication month for each hop using Bag-of-words features. We observe that the articles from Extreme right outlets are the least comparable over publication months.

how different feature representations effects the distinguishability and summarizability.

6.5.3 Effects of feature representations

We observe *better distinguishability and summarizability with SBERT features in comparing across ideologies, whereas better distinguishability and summarizability with BoW features in comparing over months*. As seen in Figure 6.2, and Figure 6.3, sentence BERT generally achieves better distinguishability (10 out of 15 cases) in comparing across ideologies of the media outlets. Furthermore, on closer inspection, better distinguishability in ideology comparisons is dominated when comparing the articles from Extreme right (most biased) outlet with the articles from Left center, Center and Right center (least biased) outlets. We hypothesize that the language structure is different across ideologies within a month, hence the SBERT based models are able to discriminate more accurately.

The average difference between distinguishability and summarizability is lower using SBERT features in ideology comparisons. For four prototypes the average difference in balanced accuracy is 0.074 for SBERT and 0.082 for BoW. This means SBERT achieves even better summarizability compared to distinguishability (see Figure 6.3). In fact, in 55 out of 60 cases, summarizability from SBERT outperforms that due to BoW features. We also notice better correlation between these two measures while using SBERT features, as the correlation numbers are binned in better correlation ranges as we see in Figure 6.3.

The bag-of-words achieves better distinguishability and summarizability in all comparisons over publication months (see Figure A.1). We hypothesize that the content changes due to change in events over months, enabling BoW to achieve better measures as the vocabulary

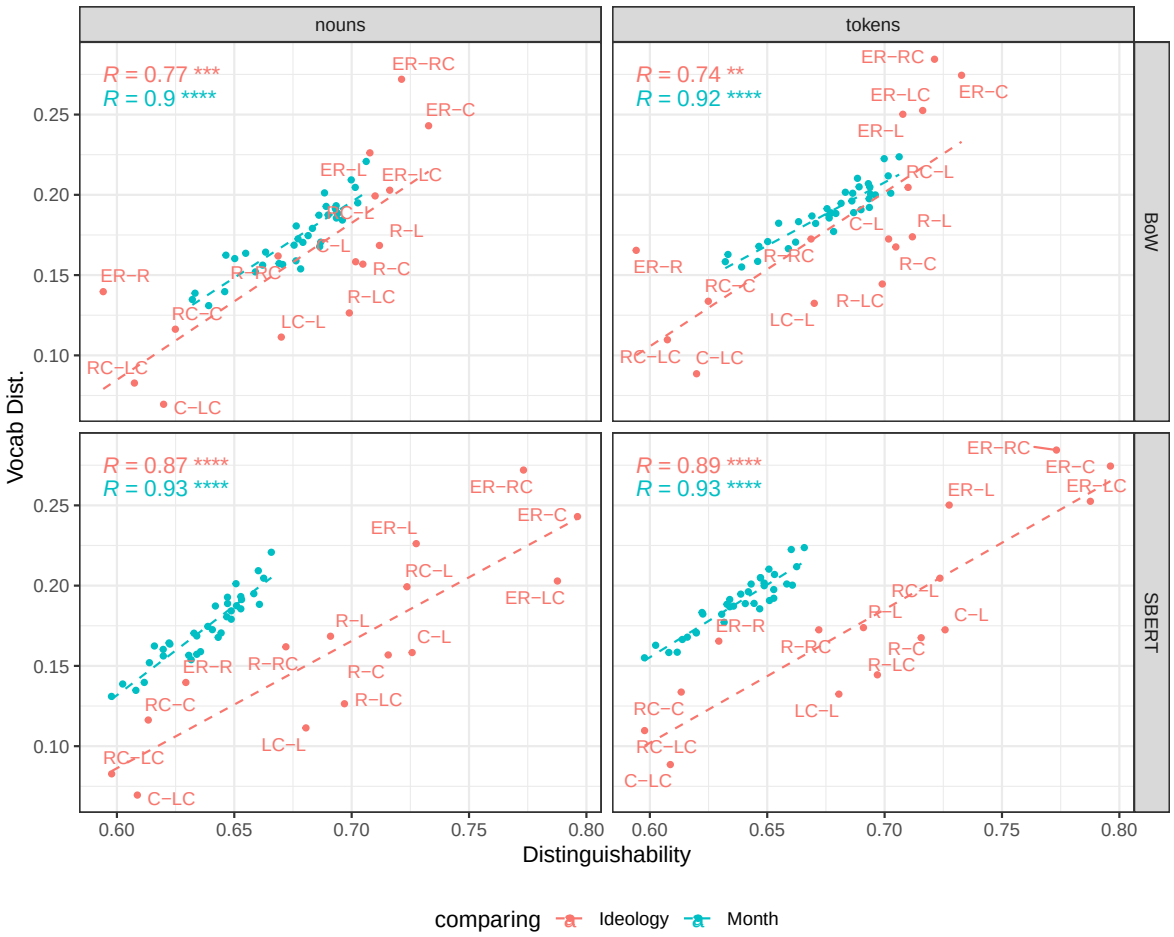


Figure 6.5: Vocabulary distance (cosine) vs Distinguishability between two comparison groups. Vocabulary distance is calculated using nouns and unigram-tokens. Each point on month comparison is averaged over $6 \times 5 \times 10 = 300$ comparisons, and on ideology comparison is averaged over $9 \times 5 \times 10 = 450$ comparisons. The line of best fit, and its correlation and p-value is annotated.

is likely to change. To test this hypothesis, we measure the vocabulary distances in terms of cosine distance of unigram-tokens distributions and nouns distributions between the two groups we are comparing. From Figure 6.5, we observe that the distinguishability due to BoW features is proportional to the vocabulary distances (both nouns and unigram-tokens) in both ideology and month comparisons (see Figure A.2 for same plot but disaggregated over topics). This suggests that a classifier with BoW features is consistent in comparing over month or across ideologies. However, a classifier with SBERT features is able to learn the differences across ideologies, even with low vocabulary distance, but not so in comparing over months. It will be interesting to investigate the nuances in language structure differences across ideologies as picked by the classifier with SBERT features. Next, we see a few example comparative summaries that aids in explaining the differences between the groups.

Left leaning	Right leaning
Gun Violence Against Women In 2019 Has Barely Made The News Cycle"	DOJ Confirms: Guns Committed In Crimes Come From Black Market
The NRA may have illegally coordinated with GOP Senate campaigns	U.S. Supreme Court (Finally) Takes Another Second Amendment Challenge to a Gun Control Law
David Hogg Says The 'National Emergency' Donald Trump Should Care About is Gun Violence, Not Border Wall	Republicans Torpedo Virginia Democrats' Anti-Gun Package, But This Is Only Round One
A suspect killed five people inside a SunTrust bank branch in Florida. Just one employee escaped.	DA calls Pittsburgh Gun-Control Plan Illegal, City Council Moves Forward Anyway

Table 6.2: An example summary from GunControl topic (title only) on comparing Left vs Right leaning media outlets (Feb 2019).

6.5.4 Explaining the comparisons with comparative summaries

In §6.5.1 and in Chapter 4, we concluded that the summaries (a few prototypes) are useful in understanding the differences among the document groups quantitatively. We now see if it holds qualitatively. Here, we present a couple of example summaries in Table 6.2 and Table 6.3. In Table 6.2, we comparatively summarize articles from left and right leaning outlets produced in GunControl topic in February 2019. This comparison task has a *summarizability* of 0.727 and a *distinguishability* of 0.744, making it to a distinguishable category in our categorization. From the summary, we can discern the two viewpoints on this topic.

Left-center leaning	Right-center leaning
Climate change will be a decisive issue in 2020	Damon Gameau: 2040 filmmaker on climate change negativity
Europe's youth want climate action. Elected leaders should give it to them	'Alarm bells:' It's spring, but B.C. already sounding drought warnings
Yorkshire village faces petrochemical giant in anti-fracking fight	The Pentagon emits a staggering level of greenhouse gases, study finds
According to NYT, 'Relentless Flooding' in Midwest Just Happens	Combating climate change needs to be a consolidated effort: Amy Khor

Table 6.3: An example summary from Climatechange topic (title only) on comparing Right-center vs Left-center leaning media outlets (Jun 2019).

In Table 6.3, we comparatively summarize articles from left-center and right-center leaning outlets produced in Climate change topic in June 2019. This comparison task has a *summarizability* of 0.605 and a *distinguishability* of 0.619, making it to the least-distinguishable category in our categorization. From the summary, we cannot clearly see if the two viewpoints are in favor or against the climate change actions. From these two example, our qualitative observation

is consistent with the quantitative measures, and summaries indeed helps in understanding the differences among the document groups if the differences exist. We note that a large scale human evaluation similar to §4.1 would be useful in shedding the further lights in qualitatively understanding the usefulness of summaries, but we leave that as future works. We next apply the comparative summarization to a different domain (tweets in social media), and see the limitations of the measurements in the domain of very short text.

6.6 Stance summarization in social media

We now briefly describe the experiments and results from applying the unsupervised comparative summarization methods to a task of stance summarization in social media (twitter).

6.6.1 SemEval 2016.6 stance classification dataset

A key challenge of automatic stance detection is the lack of available large scale labeled data. SemEval launched the stance detection competition in 2016, with 2914 labeled training and 1249 labeled test examples for supervised stance detection task and, 707 labeled test examples for unsupervised stance detection task [Mohammad et al., 2016]. The targets in the supervised stance classification task are "Hillary Clinton", "Abortion", "Feminist movement", "Climate change is real concern", and "Atheism", whereas for unsupervised task, the target is "Donald Trump". The task authors obtained the labels with human annotation, and each tweet takes either of three stance labels: "Favor", "Against", and "None" towards the target [Mohammad et al., 2016]. We use this SemEval 2016.6 stance detection dataset in our experiments to measure the distinguishability and summarizability.

6.6.2 Summarizability and Distinguishability

We use the official metric as the measure of distinguishability and summarizability instead of balanced accuracy. This makes the models we train for distinguishability and summarizability comparable with the existing works. The official evaluation metric in SemEval 2016.6 dataset is the mean of F_{favor} and $F_{against}$ [Mohammad et al., 2016], i.e.,

$$F_{avg} = \frac{F_{favor} + F_{against}}{2} \tag{6.1}$$

where F_{favor} and $F_{against}$ are calculated as:

$$F_{favor} = \frac{2 \cdot P_{favor} \cdot R_{favor}}{P_{favor} + R_{favor}} \tag{6.2}$$

$$F_{against} = \frac{2 \cdot P_{against} \cdot R_{against}}{P_{against} + R_{against}} \tag{6.3}$$

where P_{favor}, R_{favor} are the precision and recall scores of favor stance, $P_{against}$, $R_{against}$ are the precision and recall scores of against stance, are measured by micro-averaging for all targets.

6.6.3 Experimental setup

To train the models for distinguishability and summarizability, we first extracted the pre-trained BERTweet features[4] [Nguyen et al., 2020] for each labeled tweet in the SemEval 2016.6 supervised stance detection dataset. BERTweet is a large scale tweet language model. It is a BERT [Devlin et al., 2019] model trained with RoBERTa pretraining procedures [Liu et al., 2019] language model, trained on 850M English tweets, and achieves the state-of-the-art performance on three downstream NLP tasks – Part of speech (POS) tagging, Named Entity Recognition (NER), and text classification [Nguyen et al., 2020]. We then train a softmax classifier on the extracted BERTweet features, with ℓ_2 regularization, and early-stopping strategies, and trained using Adam optimizer [Kingma and Ba, 2014].

For *distinguishability*, we train the model on the entire training dataset and measure the performance on test set using the official metric. For *summarizability*, we first select prototypes using unsupervised $\mathcal{U}_{diff}(\cdot)$ (Equation 4.3), and then extrinsically evaluate the quality of prototypes using the official metric. In this experiment, there are three classes in contrast to the media-bias experiments where we had two classes. We could just train the models on a subset of labels (pro and against), but we choose to compare our results to the existing works.

6.6.4 Results and discussions

We report the classifier performance from the baselines, our distinguishability and summarizability models in Table 6.4. We first report the performance from two existing works – SVM+ngrams was a baseline model provided by the authors of the dataset [Mohammad et al., 2016], and MITRE was the best performing model in the competition [Zarrella and Marsh, 2016]. Our *distinguishability* model Lin+BERTweet performs on par with these models. Surprisingly for this task, the best model in the competition performs poorer than the baseline model [Mohammad et al., 2016].

First, we note that summarizability model trained on BERTweet features performs better

[4]https://github.com/VinAIResearch/BERTweet

	#prototypes	F_{favor}	$F_{against}$	F_{avg}
SVM+ngrams		**62.98**	74.98	**68.98**
MITRE		59.32	**76.33**	67.33
Lin+BERTweet		60.65	75.45	68.05
MMD + BoW	5	32.19	59.53	45.86
	10	34.50	60.32	47.41
MMD + BERTweet	5	39.69	61.09	50.39
	10	43.47	60.75	52.11

Table 6.4: Results using the official metric on SemEval 2016.6 Stance classification task. The first three rows are due to models trained using all trained dataset (distinguishability), whereas last 4 rows are summarizability using 5 and 10 prototypes for BoW and BERTweet features.

FAVOR	AGAINST
So prolifers are against health rights?	Abortion does not prevent rape. Kittington
You know what's best for you You know what's best for your happiness You know what's best for your well being	#ProLifeYouth know that human life = human life, inside the womb or out.
@SuePalmers @LSDsr Nothing to do with me. It's not my choice, nor is it yours, to dictate what another woman chooses. #feminism	You can't kill someone cause you claim you made a mistake. If true nobody would go to jail for killing their spouse.
@LogicOfLife7 Premise is wrong. Nothing on earth has a r̈ightẗo use someone's body w/out consent. #rapeculture	So, sorry Bernie Sanders. There are a lot of people that won't get a chance to be people. I had that chance and would like to share
God forbid you'd ever have to walk a mile in her shoes. Then you really might know what it's like to have to choose.	If being in a mother's womb isn't safe I guess neither are churches; specifically black ones #WhoIsBurningBlackChurches

Table 6.5: An example summary on Stance summarization task for the target "Abortion".

than that trained on BoW features. However, overall the summarizability models performs poorly as compared to media-bias ideology comparison tasks. For example, the difference between *distinguishability* and *summarizability* is about 0.186 in terms of in official metric (#prototypes = 5), which is significantly higher than that in media-bias ideology comparisons, which is 0.074 in terms of balanced accuracy (#prototypes = 4). We hypothesize that, this is because of short text in tweets. Tweets are limited by number of characters allowed to be posted, and hence we do not have enough data for a classifier to reliably learn to classify. However, on manually inspecting the prototypes, we find that the prototypes actually make sense, and allow us to decide if the set of prototypes are in favor or against the target. An example of stance summary on "Abortion" is shown in Table 6.5, where we can see that the prototype tweets in favor and against actually make sense for the target "Abortion"'. This suggests that

we might need an alternative evaluation for short text, as the extrinsic evaluation we proposed in this thesis might not be ideal to quantify the quality of prototypes in such domains.

6.7 Conclusion

In this chapter, we study the applicability of comparative summarization in three tasks over two datasets. We defined two automatic metrics – *distinguishability*, and *summarizability* that help us in quantifying the applicability of comparative summarization. Our first key observation is that the summarizability is proportional to the distinguishability, and is behind distinguishability by a small fraction (e.g. difference is 0.072 in ideology comparisons with 4 prototypes). This means that the prototypes are useful in understand the differences among the document groups. However, in social media dataset such as Tweets, summarizability suffers most probably due to the short text, but still the summaries make sense to the human judges. Second, we identify the groups that are least distinguishable, distinguishable or the most distinguishable in comparing articles over time or across ideological leanings. This help us to quantify the amount of coverage change over time, differences in ideologies on important topics. Finally, embeddings features are useful in comparing articles across ideologies, whereas bag of words features are useful in comparing over time. This is because in comparing over time, the vocabulary changes due to changing events, however, across the ideologies the discrepancy might be because of subtle changes in language structures which is picked up by BERT like models. While we posit these automatic metrics are useful in accessing the applicability based on our results in §4.5, a large scale human evaluations in the datasets used in this chapter would further reinforce the validity of metrics. We leave that as future works.

Chapter 7

Conclusions

> *In literature and in life we ultimately pursue, not conclusions, but beginnings.*
>
> Sam Tanenhaus, *Literature Unbound*

In this chapter, we first summarize the contributions we made in addressing the problem of comparative summarization. We then present some potential research directions for extending this work in the future.

7.1 Summary

Given a collection with multiple groups of documents, comparative summarization is the problem of summarizing each group such that the summaries represent the information salient of each group, while simultaneously highlighting the differences with other document groups. It helps us to tackle the problem of information overload and also understand the differences between the groups of documents. It has several interesting applications in news and social media, such as comparatively summarizing the news coverage in *beefban* in two different time periods, comparing bias in the coverage of *guncontrol* by news media, or comparing the stances in *abortion* on twitter. However, comparative summarization has not got much attention as other summarization types. In Chapter 1, we identified several research questions in comparative summarization.

Comparative summarization is a challenging problem as summaries need to be representative and contrastive. Even more challenging is in unsupervised setting, where we seek to develop a method without ground truth summaries. Another challenge is in evaluating the summaries in the absence of ground truth summaries, as most popular existing methods in summarization evaluation works by comparing against ground truth summaries [Lin, 2004]. Unsupervised method and evaluations would allow us to apply comparative summarization

in any appropriate dataset. While unsupervised methods are helpful in applying the comparative summarization to different datasets, supervision could provide further signals to build even more effective summarization systems in domains where ground truth summaries are available. Finally, identifying the different facets of comparisons within a dataset could shed further light into the applicability of the methods, and understanding the groups being compared. In this thesis, we addressed these in detail by making several contributions, which we briefly summarize now.

First, availability of relevant datasets is antecedent in addressing the research goals in machine learning. While there are many datasets in summarization literature as discussed in §2.3, most of them are not applicable to comparative summarization. To address this, we first curated two datasets of controversial news topics such as climatechange, beefban, etc., as controversial topics evolve over time and attracts contrasting coverage in news media. First, our ControvNews2017 dataset (§3.3) allows us to study comparative summarization over time in Chapter 4. Our second News2019+Bias dataset (§3.4) allows us to study the applicability of comparative summarization in diverse topics when the groups are defined in multiple ways, such as publication month or ideological leanings of the media outlets in Chapter 6.

We formulated the problem of comparative summarization in terms of competing binary classifiers. This novel formulation is intuitive as it naturally captures the desired properties of comparative summaries – information coverage, diversity and discriminativeness (§4.3). Using this formulation, we first developed unsupervised methods incorporating the representativeness and discriminativeness of summaries based on maximum mean discrepancy (§4.4.2). Unsupervised methods allow us to compare document collections in datasets without ground truth summaries. The methods are submodular-monotone which admits an efficient greedy algorithm for discrete optimization, and can also be optimized using gradient-based methods which admits lower memory footprints (§4.5).

Next, we developed new extrinsic evaluation protocols for comparative summarization, when ground truth summaries are not available (§4.6.1). The evaluation is based on the performance of the classifier trained on summaries, and can be done automatically, or crowdsourced to human judges. As we discuss with experiments in Chapter 4, the evaluation is useful in identifying the usefulness of the summaries (§4.6.4, §4.6.6). And we show the efficacy of our methods against the baselines in comparing news topics over time in ControvNews2017 datasets. Furthermore, this automatic evaluation is scalable to large datasets, and applicable to new datasets. This allows us to perform large scale measurement studies on news and social media domain using News2019+Bias and stance detection datasets in Chapter 6, which we discuss later.

While our competing binary classifier based methods work well in unsupervised setting as we show in Chapter 4, extending it to supervised setting would make it widely applicable,

and make it more effective in the presence of ground truth summaries. The notion of summary formation in our unsupervised methods comes from inter-document (or inter-sentence) similarity measures only. In the presence of ground truth summaries, they can provide extra signals about the documents (or sentences) that form the summary. Our new supervised method for generic and comparative summarization in Chapter 5, *SupMMD* (§5.3), can learn to identify the sentences important in forming the summaries from extra features such as the position of sentence within a document, presence of nouns, graph centrality scores, etc. Furthermore, we adapt our method to combine multiple features using multiple kernel learning [Cortes et al., 2010] (§5.3.6). The method meets or exceeds the state-of-the-art performance on the benchmark DUC-2004 and TAC-2009 datasets (§5.5).

The underlying assumption in a limited literature of comparative summarization is that groups are comparable, without regarding the degree of comparativeness between the groups, and how much summaries help in understanding the comparisons. We defined two metrics that helped us to quantify the degree of comparativeness between document groups, and usefulness of the summaries – distinguishability and summarizability (§6.3). We perform the large scale measurement studies based on these metrics on different news and social media datasets, and across different facets of comparisons such as comparing over time or across ideological leanings. In News2019+Bias datasets, we observe that the summarizability is proportional and close to the distinguishability, meaning summaries are useful in comparing document collections. We also observe that the distinguishability is amenable to feature representations based on the type of comparisons we are doing – over months or across the ideological leanings. However, in the task of stance summarization in social media, summarizability suffers due to sparsity of data, even though the summaries make sense in qualitative analysis.

Overall, our methods and empirical study demonstrate the potential applications of supervised and unsupervised extractive methods to the problem of comparative summarization in news and social media domains. As the techniques evolve, we are likely to see a shift of focus towards abstractive comparative summarization. In terms of applications, we are likely to see comparisons using other facets of documents, such as publication source, authors or geography, and in other data domains such as joint multimodal comparisons of text and images. We next summarize the ways in which this work can be extended in the future.

7.2 Future work

Abstractive comparative summarization. We exclusively focused on extractive comparative summarization in this work, so abstractive comparative summarization using recent deep-neural network based methods would be an interesting investigation. While extractive sum-

maries are useful, they often incoherent due to disconnect between the prototypes. This is not the case in abstractive summaries, as abstractive summarization model can be taught to be coherent [Christensen et al., 2013]. An example of abstractive comparative summarization would be question-answering type system that would provide a short coherent and factual comparative summary with salient information. In last few years neural abstractive summarization methods has gained significant interest for generic summarization using recurrent neural networks [Nallapati et al., 2016a; Chen and Zhuge, 2018], autoencoders [Chu and Liu, 2019], convolutional neural networks [Narayan et al., 2018a], transformers based architectures [Pilault et al., 2020; Liu and Lapata, 2019b](see §2.1.2). We envisage the enhancements in this direction for comparative summarization in the coming future.

Multi-modal comparative summarization. Recently, multi-modal summarization, such as summarizing images and text jointly has gained some interests [Zhu et al., 2018, 2020; Chen and Zhuge, 2018]. Multimodal summaries such as text and images aids users to get a more visualized understanding of events [Zhu et al., 2018]. So, a multi-modal comparative summarization would be an interesting research problem, where images provide a visual representation of the differences and the text summary adds more details to the differences between document groups. Moreover, there has been recent works on multimodal generation for recommendation systems [Truong and Lauw, 2019], medical reports [Liu et al., 2021]. Hence, we foresee multimodal abstractive summarization catching the research interests in coming years.

Other facets of comparisons. We studied comparisons over time in Chapter 4, Chapter 5, and Chapter 6, comparisons across ideological leanings and stances in Chapter 6. There are other ways by which we can define groups, such as publication sources, authors, or geography. An example would be the coverage of COVID-19 in western and Asian media as identified by [AlAfnan, 2020]. Comparative summarization on these groups might reveal interesting differences between them. Second, our work focused only on differences between groups, while summarizing the similarity between the document groups can be of interests as well. For example, would the western and Asian media agree on COVID-19 origins?

Improving supervised method and training. Our *SupMMD* method (§5.3) can be improved in several ways. First, we currently employ a two stage method to learn the kernel combination weights, and sentence saliency separately. A single stage learning would help us build end-end-end system. A single stage optimization to learn a linear combination of kernels in weighted MMD (§5.3.2) did not work because since MMD is a linear form of kernel (Lemma §2.1), minimizing weighted MMD with a linear combination of kernels would always place all weight on a single kernel. There has been some works on combining kernels for MMD [Gretton et al., 2012b], so learning kernel combination and sentence importance

in a single optimization step would be an interesting research direction. Second, instead of log-linear models and surface features, employing a deep neural-network based methods to learn sentence importances [Yasunaga et al., 2017] within MMD framework using additional set of features. Finally, we could employ data augmentation to avoid overfitting [Shorten and Khoshgoftaar, 2019] in addition L_2 regularization that we currently use for small scale DUC-2003 and TAC-2008 datasets. For example, in the multi-document summarization datasets we use, there are typically 10 documents in each group/topic to be summarized, along with ground truth summaries. One potential candidate data augmentation strategy would be to randomly remove 1-2 documents from the topic/group and use the pair of 8-9 documents, and ground truth summary as a new training example.

Understanding linguistic differences across ideologies. We made several interesting observations in the News2019+Bias datasets (§6.5). We hypothesize that the sentence BERT feature doing better in ideology comparisons is due to the linguistic differences across ideologies, which is better picked up by the transformer based SBERT representations. However, in comparing over publication months, the events changes, hence giving different vocabularies and enabling bag-of-words to perform better. We tested the second part of the hypothesis in §6.5.3. But, investigating if subtle linguistic differences across ideologies exists in news media, and if transformer based representation models can accurately identify such differences would be an interesting topic for future study. Large scale human evaluations could shed further lights on this.

Appendices

A.1 Comments on MMD-critic (Chapter 4)

The criticisms proposed in *MMD-critic* [Kim et al., 2016] implies a curious intuition: it is trying to find criticisms (additional prototypes) in the region where prototypes have under- and over-represented the dataset, because of $abs(\cdot)$ in the first term of Equation (A.1). The $abs(\dot{)}$ is needed to make the objective monotone. Instead, we only need to search for additional prototypes in under-represented regions only and this can be achieved by removing the absolute sign. Let V, S, C be the indices of data, prototypes and criticisms, then the criticisms objective that only searches for criticisms in underrepresented regions is given by Equation (A.1).

$$\mathcal{U}_{critic}(C; V, S) = \sum_{c \in C} \left(\frac{1}{n} \sum_{i \in V} \mathrm{k}(\mathbf{x}_i, \mathbf{x}_c) - \frac{1}{m} \sum_{j \in S} \mathrm{k}(\mathbf{x}_j, \mathbf{x}_c) \right) + \log\det \mathbf{K}_{C,C} \tag{A.1}$$

The second part $\log\det \mathbf{K}_{C,C}$ is a regularizer which encourages the diversity of criticisms by maximizing the volume spanned by sub-matrix $\mathbf{K}_{C,C}$, and is submodular [Kim et al., 2016].

A.2 Additional results of Chapter 4

Tables A.2, A.1, A.3, A.4 provide results table of Figure 4.4.

A.3 Media bias dataset statistics of each topic

We present the counts over different months and across ideological leanings for Illegal immigration topic in Table A.5, refugees topic in Table A.6 and un control topic in Table A.7. Topics Climate change (Table 6.1) and topic LGBT (Table 3.4) are presented in main text.

method	2	4	8	16	2	4	8	16
kmedoids	0.501 ± .010	0.505 ± .011	0.528 ± .012	0.530 ± .021	0.498 ± .008	0.501 ± .005	0.523 ± .014	0.523 ± .02
kmeans	0.510 ± .017	0.542 ± .012	*0.546 ± .007*	0.576 ± .008	0.507 ± .016	*0.546 ± .009*	0.541 ± .01	0.551 ± .012
mmd-critic	0.499 ± .003	0.514 ± .016	0.515 ± .013	0.531 ± .009	0.5 ± .001	0.499 ± .01	0.51 ± .017	0.53 ± .012
mmd-diff-grad	*0.534 ± .011*	*0.538 ± .014*	0.540 ± .020	**0.582 ± .010**	*0.53 ± .011*	0.542 ± .01	0.559 ± .012	0.575 ± .011
mmd-div-grad	**0.539 ± .013**	**0.545 ± .008**	**0.564 ± .014**	*0.579 ± .011*	0.527 ± .012	0.546 ± .014	**0.56 ± .012**	**0.579 ± .009**
nn-comp-greedy	0.509 ± .011	0.515 ± .008	0.544 ± .009	0.577 ± .007	0.517 ± .018	0.529 ± .015	*0.56 ± .007*	0.577 ± .006
mmd-diff-greedy	0.530 ± .009	0.536 ± .012	0.545 ± .013	0.564 ± .013	0.529 ± .014	**0.554 ± .012**	0.553 ± .012	0.575 ± .011
mmd-div-greedy	0.530 ± .010	0.525 ± .011	0.539 ± .012	0.563 ± .009	**0.532 ± .014**	0.544 ± .013	0.544 ± .007	*0.578 ± .011*

Table A.1: Classification performance on *Capital Punishment* News dataset. (left) 1-NN, (right) SVM.

method	2	4	8	16	2	4	8	16
kmedoids	0.592 ± .016	0.586 ± .011	0.582 ± .025	0.589 ± .023	0.573 ± .022	0.571 ± .013	0.574 ± .025	0.578 ± .031
kmeans	0.582 ± .020	**0.613 ± .017**	*0.622 ± .030*	0.629 ± .025	0.57 ± .024	*0.608 ± .022*	*0.615 ± .02*	0.637 ± .027
mmd-critic	0.541 ± .031	0.524 ± .028	0.530 ± .025	0.535 ± .033	0.531 ± .028	0.533 ± .021	0.516 ± .039	0.542 ± .029
mmd-diff-grad	*0.595 ± .019*	0.587 ± .023	0.610 ± .017	*0.633 ± .020*	0.592 ± .036	**0.612 ± .022**	0.615 ± .024	*0.641 ± .019*
mmd-div-grad	**0.602 ± .021**	*0.605 ± .015*	0.605 ± .028	**0.636 ± .028**	*0.595 ± .019*	0.597 ± .025	0.608 ± .026	0.636 ± .019
nn-comp-greedy	0.587 ± .018	0.600 ± .027	**0.624 ± .029**	0.627 ± .026	0.588 ± .02	0.605 ± .014	0.615 ± .023	0.639 ± .026
mmd-diff-greedy	0.592 ± .027	0.595 ± .021	0.615 ± .019	0.629 ± .021	**0.596 ± .019**	0.597 ± .022	**0.626 ± .021**	**0.642 ± .021**
mmd-div-greedy	0.586 ± .014	0.578 ± .021	0.593 ± .022	0.616 ± .025	0.594 ± .019	0.583 ± .026	0.597 ± .027	0.632 ± .024

Table A.2: Classification performance on *Beefban* News dataset. (left) 1-NN, (right) SVM.

method	2	4	8	16	2	4	8	16
kmedoids	0.501 ± .016	0.504 ± .012	0.518 ± .016	0.518 ± .013	0.502 ± .014	0.502 ± .007	0.516 ± .011	0.52 ± .006
kmeans	0.505 ± .009	0.506 ± .006	**0.538 ± .013**	0.542 ± .014	0.508 ± .007	0.504 ± .01	0.527 ± .01	0.536 ± .008
mmd-critic	0.506 ± .006	0.511 ± .013	0.507 ± .011	0.514 ± .014	0.507 ± .007	0.504 ± .012	0.518 ± .018	0.518 ± .011
mmd-diff-grad	**0.531 ± .009**	**0.529 ± .006**	0.532 ± .008	**0.566 ± .006**	0.541 ± .008	*0.528 ± .011*	0.535 ± .009	**0.554 ± .016**
mmd-div-grad	0.525 ± .011	*0.525 ± .009*	*0.537 ± .010*	*0.563 ± .011*	**0.532 ± .013**	0.53 ± .012	**0.54 ± .011**	*0.553 ± .014*
nn-comp-greedy	0.502 ± .010	0.518 ± .011	0.535 ± .014	0.555 ± .009	0.515 ± .01	0.528 ± .011	0.528 ± .009	0.543 ± .009
mmd-diff-greedy	*0.524 ± .015*	*0.520 ± .013*	0.521 ± .013	0.537 ± .010	*0.521 ± .017*	**0.529 ± .016**	*0.536 ± .013*	0.55 ± .011
mmd-div-greedy	0.517 ± .012	0.515 ± .010	0.519 ± .012	0.532 ± .011	0.506 ± .01	0.531 ± .012	0.528 ± .013	0.55 ± .012

Table A.3: Classification performance on *Gun Control* News dataset. (left) 1-NN, (right) SVM.

method	2	4	8	16	2	4	8	16
kmedoids	0.805 ± .010	0.836 ± .014	0.862 ± .008	0.881 ± .008	0.783 ± .009	0.845 ± .01	0.886 ± .005	0.907 ± .009
kmeans	**0.823 ± .012**	**0.866 ± .010**	*0.888 ± .006*	0.909 ± .009	0.815 ± .014	0.869 ± .011	**0.901 ± .006**	*0.924 ± .009*
mmd-critic	0.560 ± .019	0.700 ± .016	0.777 ± .013	0.839 ± .010	0.48 ± .037	0.63 ± .035	0.795 ± .022	0.877 ± .007
mmd-diff-grad	*0.811 ± .011*	0.852 ± .008	0.882 ± .007	*0.910 ± .010*	*0.831 ± .013*	**0.877 ± .008**	0.9 ± .011	0.922 ± .007
mmd-div-grad	0.806 ± .010	0.849 ± .010	0.876 ± .007	0.907 ± .010	**0.834 ± .015**	*0.877 ± .009*	*0.901 ± .011*	0.919 ± .008
nn-comp-greedy	0.800 ± .011	*0.859 ± .010*	**0.890 ± .009**	**0.914 ± .010**	0.792 ± .009	0.856 ± .012	0.894 ± .008	**0.924 ± .008**
mmd-diff-greedy	0.783 ± .011	0.835 ± .009	0.871 ± .009	0.898 ± .010	0.801 ± .008	0.86 ± .009	0.892 ± .011	0.917 ± .01
mmd-div-greedy	0.784 ± .013	0.840 ± .010	0.866 ± .010	0.898 ± .007	0.798 ± .008	0.866 ± .012	0.895 ± .01	0.92 ± .009

Table A.4: Classification performance on *USPS* dataset. (left) 1-NN, (right) SVM.

	ER	**R**	**RC**	**C**	**LC**	**L**	**Total**
2019-01	501	870	436	656	1513	346	4322
2019-02	288	467	221	285	828	197	2286
2019-03	357	577	240	306	743	165	2388
2019-04	359	708	267	455	960	349	3098
2019-05	284	504	208	298	673	161	2128
2019-06	272	553	177	348	740	234	2324
2019-07	356	684	272	451	1040	343	3146
2019-08	277	407	141	280	660	215	1980
2019-09	171	277	149	225	510	122	1454
Total	2865	5047	2111	3304	7667	2132	23126

Table A.5: Counts over months and across ideologies for Illegal Immigration topic

	ER	R	RC	C	LC	L	Total
2019-01	144	319	343	466	1132	247	2651
2019-02	113	268	276	398	994	166	2215
2019-03	202	313	280	417	1061	198	2471
2019-04	145	254	201	303	829	188	1920
2019-05	115	199	178	292	782	164	1730
2019-06	106	259	238	389	992	240	2224
2019-07	178	233	199	348	881	299	2138
2019-08	121	184	190	317	719	161	1692
2019-09	92	151	175	240	675	190	1523
Total	1216	2180	2080	3170	8065	1853	18564

Table A.6: Counts over months and across ideologies for Refugees topic

	ER	R	RC	C	LC	L	Total
2019-01	89	410	156	234	474	149	1512
2019-02	128	526	242	337	628	232	2093
2019-03	128	561	212	328	751	233	2213
2019-04	71	368	157	239	497	148	1480
2019-05	96	421	192	219	512	154	1594
2019-06	69	326	127	203	436	161	1322
2019-07	69	238	115	212	399	113	1146
2019-08	364	984	422	818	1915	843	5346
2019-09	269	645	187	304	774	330	2509
Total	1283	4479	1810	2894	6386	2363	19215

Table A.7: Counts over months and across ideologies for Gun Control topic

A.4 Additional results to Chapter 6

We show the 2d matrix plot for comparison over time in Figure A.1 (corresponding plot for comparison over ideologies is provided in main text 6.3). We also provide vocabulary distance vs distinguishibility plot for each topic in Figure A.2, which is de-aggregated plot of Figure 6.5.

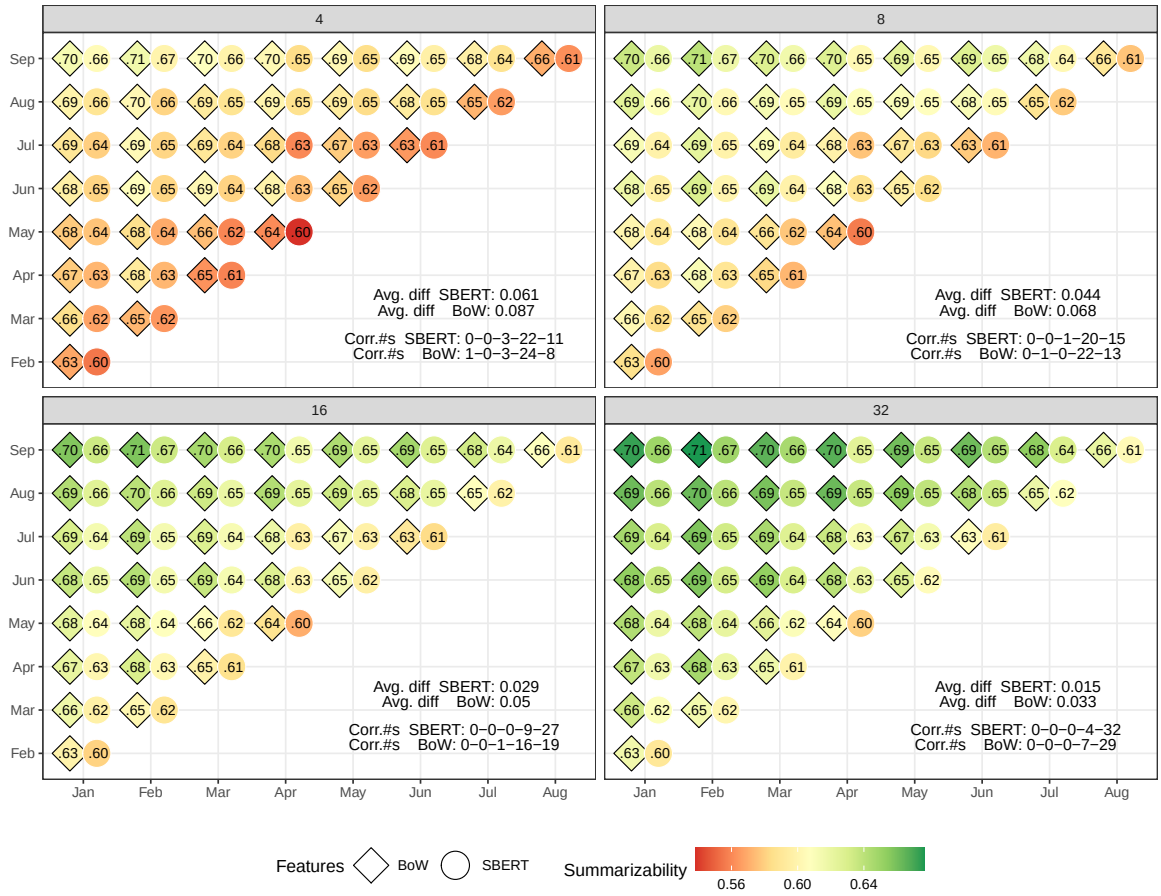


Figure A.1: Comparison over time results when viewed on 2D matrix over ideology spectrum for each of 4 different number of prototypes (4, 8, 16, 32). Numbers inside the bubbles are *distinguishability*, whereas *summarizability* is denoted by the intensity of color. The shape represents the features representations, and a border is added to the bubble corresponding to the feature that provides better summarizability. The average difference, and the correlation (binned counts) between the distinguishability and the summarizability for each number of prototype settings is annotated. The correlation is binned as: negligible $[-1, .2)$, low $[.2, .4)$, moderate $[.4, .6)$, strong $[.6, .8)$, and very-strong $[.6, .1]$.

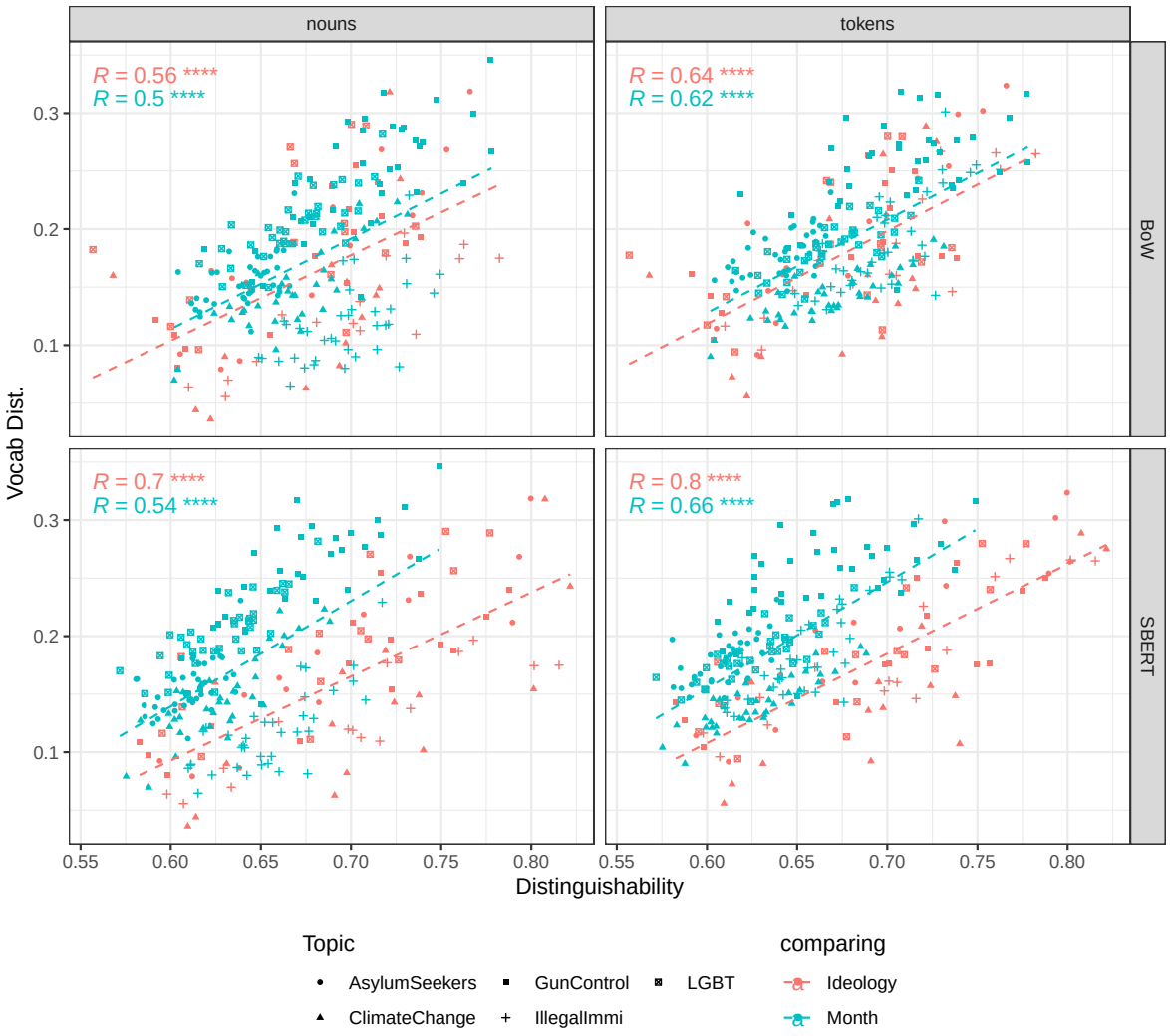


Figure A.2: Vocabulary distance (cosine) vs Distinguishability in terms of nouns and unigram-tokens between two groups we are comparing. Each point in ideology comparisons is averaged over 9 months, and each point in month comparison is averaged over 6 ideologies. The line of best fit, correlation and p-value are annotated.

Bibliography

AlAfnan, M. A., 2020. Covid 19-the foreign virus: Media bias, ideology and dominance in chinese and american newspaper articles. *International Journal of Applied Linguistics and English Literature*, 9, 1 (2020), 56–60. (cited on pages 44, 98, and 119)

Aldayel, A. and Magdy, W., 2019. Your stance is exposed! analysing possible factors for stance detection on social media. *Proc. ACM Hum.-Comput. Interact.*, 3, CSCW (Nov. 2019). doi: 10.1145/3359307. URL https://doi.org/10.1145/3359307. (cited on pages 99 and 100)

AlDayel, A. and Magdy, W., 2020. Stance detection on social media: State of the art and trends. *arXiv preprint arXiv:2006.03644*, (2020). (cited on page 99)

Alguliev, R. M.; Aliguliyev, R. M.; Hajirahimova, M. S.; and Mehdiyev, C. A., 2011. Mcmr: Maximum coverage and minimum redundant text summarization model. *Expert Systems with Applications*, 38, 12 (2011), 14514 – 14522. doi: https://doi.org/10.1016/j.eswa.2011.05.033. URL http://www.sciencedirect.com/science/article/pii/S0957417411008177. (cited on page 85)

Anderson, P.; Fernando, B.; Johnson, M.; and Gould, S., 2016. Spice: Semantic propositional image caption evaluation. In *European conference on computer vision*, 382–398. Springer. (cited on page 73)

Aronszajn, N., 1950. Theory of reproducing kernels. *Transactions of the American mathematical society*, 68, 3 (1950), 337–404. (cited on pages 30 and 31)

Arthur, D. and Vassilvitskii, S., 2007. k-means++: The advantages of careful seeding. In *ACM-SIAM symposium on Discrete algorithms*. (cited on page 66)

Augenstein, I.; Rocktäschel, T.; Vlachos, A.; and Bontcheva, K., 2016. Stance detection with bidirectional conditional encoding. *arXiv preprint arXiv:1606.05464*, (2016). (cited on page 99)

autotl;dr, 2021. autotl;dr. URL https://http://autotldr.io. (cited on page 1)

Baly, R.; Karadzhov, G.; Saleh, A.; Glass, J.; and Nakov, P., 2019. Multi-task ordinal regression for jointly predicting the trustworthiness and the leading political ideology of news media. *arXiv preprint arXiv:1904.00542*, (2019). (cited on page 98)

Baly, R.; Martino, G. D. S.; Glass, J.; and Nakov, P., 2020. We can detect your bias: Predicting the political ideology of news articles. *arXiv preprint arXiv:2010.05338*, (2020). (cited on page 98)

Baron, D. P., 2006. Persistent media bias. *Journal of Public Economics*, 90, 1-2 (2006), 1–36. (cited on pages 96 and 98)

Bernhardt, D.; Krasa, S.; and Polborn, M., 2008. Political polarization and the electoral effects of media bias. *Journal of Public Economics*, 92, 5-6 (2008), 1092–1104. (cited on pages 96 and 98)

Bien, J. and Tibshirani, R., 2011. Prototype selection for interpretable classification. *The Annals of Applied Statistics*, (2011). (cited on pages 4, 23, 24, 26, 56, 59, 64, 65, 67, and 72)

Bird, S., 2006. NLTK: The Natural Language Toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, 69–72. Association for Computational Linguistics, Sydney, Australia. doi: 10.3115/1225403.1225421. URL https://www.aclweb.org/anthology/P06-4018. (cited on page 84)

Bista, U., 2019. Comparative summarisation of rich media collections. WSDM '19 Doc. Sym., 812–813. doi: 10.1145/3289600.3291603. URL http://doi.acm.org/10.1145/3289600.3291603. (cited on page 53)

Bista, U.; Mathews, A.; Menon, A.; and Xie, L., 2020. SupMMD: A sentence importance model for extractive summarization using maximum mean discrepancy. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, 4108–4122. Association for Computational Linguistics. doi: 10.18653/v1/2020.findings-emnlp.367. URL https://www.aclweb.org/anthology/2020.findings-emnlp.367. (cited on pages 76 and 96)

Bista, U.; Mathews, A. P.; Shin, M.; Menon, A. K.; and Xie, L., 2019. Comparative document summarisation via classification. AAAI 2019. AAAI Press. doi: 10.1609/aaai.v33i01.330120. URL https://doi.org/10.1609/aaai.v33i01.330120. (cited on pages 42, 53, and 96)

Blei, D. M.; Ng, A. Y.; and Jordan, M. I., 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3, Jan (2003), 993–1022. (cited on page 28)

Boykoff, M. T. and Boykoff, J. M., 2004. Balance as bias: Global warming and the us prestige press. *Global environmental change*, 14, 2 (2004), 125–136. (cited on pages 96 and 98)

Brandow, R.; Mitze, K.; and Rau, L. F., 1995. Automatic condensation of electronic publications by sentence selection. *Information Processing & Management*, 31, 5 (1995), 675–685. (cited on page 65)

Brodersen, K. H.; Ong, C. S.; Stephan, K. E.; and Buhmann, J. M., 2010. The balanced accuracy and its posterior distribution. In *International Conference on Pattern Recognition*. (cited on pages 67 and 105)

Byrd, R. H.; Lu, P.; Nocedal, J.; and Zhu, C., 1995. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, (1995). (cited on pages 63 and 67)

Cao, Z.; Wei, F.; Dong, L.; Li, S.; and Zhou, M., 2015. Ranking with recursive neural networks and its application to multi-document summarization. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15 (Austin, Texas, 2015), 2153–2159. AAAI Press. (cited on pages 13, 14, 15, and 78)

Carbonell, J. and Goldstein, J., 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *ACM SIGIR Conference on Research and Development in Information Retrieval*. (cited on pages 11, 37, and 55)

Chen, C.; Buntine, W.; Ding, N.; Xie, L.; and Du, L., 2014. Differential topic models. *IEEE transactions on pattern analysis and machine intelligence*, 37, 2 (2014), 230–242. (cited on page 28)

Chen, J. and Zhuge, H., 2018. Abstractive text-image summarization using multi-modal attentional hierarchical rnn. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 4046–4056. (cited on page 119)

Chen, W.-F.; Al-Khatib, K.; Stein, B.; and Wachsmuth, H., 2020a. Detecting media bias in news articles using gaussian bias distributions. *arXiv preprint arXiv:2010.10649*, (2020). (cited on page 98)

Chen, W.-F.; Al-Khatib, K.; Wachsmuth, H.; and Stein, B., 2020b. Analyzing political bias and unfairness in news articles at different levels of granularity. *arXiv preprint arXiv:2010.10652*, (2020). (cited on page 98)

Chen, W.-F.; Syed, S.; Stein, B.; Hagen, M.; and Potthast, M., 2020c. Abstractive snippet generation. In *Proceedings of The Web Conference 2020*, WWW '20 (Taipei, Taiwan, 2020), 1309–1319. Association for Computing Machinery, New York, NY, USA. doi: 10.1145/3366423.3380206. URL https://doi.org/10.1145/3366423.3380206. (cited on pages 21 and 22)

Cheng, J. and Lapata, M., 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)*, 484–494. Association for Computational Linguistics, Berlin, Germany. doi: 10.18653/v1/P16-1046. URL https://www.aclweb.org/anthology/P16-1046. (cited on page 27)

Cho, S.; Lebanoff, L.; Foroosh, H.; and Liu, F., 2019. Improving the similarity measure of determinantal point processes for extractive multi-document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1027–1038. Association for Computational Linguistics, Florence, Italy. doi: 10.18653/v1/P19-1098. URL https://www.aclweb.org/anthology/P19-1098. (cited on pages 12, 14, 15, 87, 88, and 89)

Christensen, J.; Mausam; Soderland, S.; and Etzioni, O., 2013. Towards coherent multi-document summarization. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1163–1173. Association for Computational Linguistics, Atlanta, Georgia. URL https://www.aclweb.org/anthology/N13-1136. (cited on page 119)

Chu, E. and Liu, P., 2019. Meansum: a neural model for unsupervised multi-document abstractive summarization. In *International Conference on Machine Learning*, 1223–1232. PMLR. (cited on pages 13, 15, 25, and 119)

Cohan, A.; Dernoncourt, F.; Kim, D. S.; Bui, T.; Kim, S.; Chang, W.; and Goharian, N., 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)*, 615–621. Association for Computational Linguistics, New Orleans, Louisiana. doi: 10.18653/v1/N18-2097. URL https://www.aclweb.org/anthology/N18-2097. (cited on pages 21 and 22)

Cohen, K. B.; Dang, H. T.; de Waard, A.; Yadav, P.; and Vanderwende, L., 2014. Tac 2014 biomedical summarization task description. (cited on page 21)

Conroy, J.; Davis, S. T.; Kubina, J.; Liu, Y.-K.; O'Leary, D. P.; and Schlesinger, J. D., 2013. Multilingual summarization: Dimensionality reduction and a step towards optimal term coverage. In *Proceedings of the MultiLing 2013 Workshop on Multilingual Multi-document Summarization*, 55–63. Association for Computational Linguistics, Sofia, Bulgaria. URL https://www.aclweb.org/anthology/W13-3108. (cited on pages 88 and 89)

Conroy, J. M. and O'leary, D. P., 2001. Text summarization via hidden markov models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, 406–407. ACM. (cited on page 12)

Cornuéjols, G.; Nemhauser, G.; and Wolsey, L., 1983. The uncapicitated facility location problem. Technical report, Cornell University Operations Research and Industrial Engineering. (cited on page 37)

Cortes, C.; Mohri, M.; and Rostamizadeh, A., 2010. Two-stage learning kernel algorithms. In *Proceedings of the 27th Annual International Conference on Machine Learning (ICML 2010)*. URL http://www.cs.nyu.edu/~mohri/pub/align.pdf. (cited on pages 6, 76, 77, 82, 83, and 118)

Cortes, C. and Vapnik, V., 1995. Support-vector networks. *Machine learning*, 20, 3 (1995), 273–297. (cited on pages 31 and 105)

D'Alessio, D. and Allen, M., 2000. Media bias in presidential elections: A meta-analysis. *Journal of communication*, 50, 4 (2000), 133–156. (cited on pages 96 and 98)

Daners, D., 2008. Introduction to functional analysis. *The University of Sydney*, (2008). (cited on pages 29, 33, and 80)

Dang, H. T., 2005. Overview of duc 2005. (cited on pages 16, 21, and 25)

Dang, H. T., 2006a. Duc 2005: Evaluation of question-focused summarization systems. In *Proceedings of the Workshop on Task-Focused Summarization and Question Answering*, SumQA '06 (Sydney, Australia, 2006), 48–55. Association for Computational Linguistics, USA. (cited on page 16)

Dang, H. T., 2006b. Overview of duc 2007. (cited on pages 16 and 21)

Dang, H. T., 2007. Overview of duc 2006. (cited on page 21)

Dang, H. T. and Owczarzak, K., 2008. Overview of the tac 2008 update summarization task. In *TAC*. (cited on pages 3, 5, 17, 21, and 25)

Dang, H. T. and Owczarzak, K., 2009. Overview of the tac 2009 summarization task. In *TAC*. (cited on pages 5, 21, and 25)

Darwish, K.; Stefanov, P.; Aupetit, M.; and Nakov, P., 2020. Unsupervised user stance detection on twitter. In *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 14, 141–152. (cited on pages 99 and 100)

Daumé III, H. and Marcu, D., 2006. Bayesian query-focused summarization. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, 305–312. Association for Computational Linguistics, Sydney, Australia. doi: 10.3115/1220175.1220214. URL https://www.aclweb.org/anthology/P06-1039. (cited on page 16)

Demner-Fushman, D. and Lin, J., 2006. Answer extraction, semantic clustering, and extractive summarization for clinical question answering. ACL-44, 841–848. ACL. doi: 10.3115/1220175.1220281. URL https://doi.org/10.3115/1220175.1220281. (cited on page 1)

Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L., 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, 248–255. Ieee. (cited on page 6)

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K., 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)*, 4171–4186. Association for Computational Linguistics, Minneapolis, Minnesota. doi: 10.18653/v1/N19-1423. URL https://www.aclweb.org/anthology/N19-1423. (cited on pages 96, 104, and 113)

Dey, K.; Shrivastava, R.; and Kaushik, S., 2018. Topical stance detection for twitter: A two-phase lstm model using attention. In *European Conference on Information Retrieval*, 529–536. Springer. (cited on page 99)

Diakopoulos, N.; Elgesem, D.; Salway, A.; Zhang, A.; and Hofland, K., 2015. Compare clouds: Visualizing text corpora to compare media frames. In *Proceedings of IUI Workshop on Visual Text Analytics*, 193–202. (cited on page 28)

Dias, G.; Alves, E.; and Lopes, J. G. P., 2007. Topic segmentation algorithms for text summarization and passage retrieval: An exhaustive evaluation. In *Proceedings of the 22nd National Conference on Artificial Intelligence - Volume 2*, AAAI'07 (Vancouver, British Columbia, Canada, 2007), 1334–1339. AAAI Press. (cited on page 85)

Dictionary, C., 2021a. Cambridge dictionary: distinguishable. URL https://dictionary.cambridge.org/dictionary/english/distinguishable. (cited on page 100)

Dictionary, C., 2021b. Cambridge dictionary: stance. URL https://dictionary.cambridge.org/dictionary/english/ideology. (cited on page 96)

Dictionary, C., 2021c. Cambridge dictionary: stance. URL https://dictionary.cambridge.org/dictionary/english/stance. (cited on page 97)

Dole, J. A.; Duffy, G. G.; Roehler, L. R.; and Pearson, P. D., 1991. Moving from the old to the new: Research on reading comprehension instruction. *Review of educational research*, 61, 2 (1991), 239–264. (cited on page 1)

Dong, R.; Sun, Y.; Wang, L.; Gu, Y.; and Zhong, Y., 2017. Weakly-guided user stance prediction via joint modeling of content and social interaction. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, 1249–1258. (cited on pages 99 and 100)

Dornaika, F. and Aldine, I. K., 2015. Decremental sparse modeling representative selection for prototype selection. *Pattern Recognition*, 48, 11 (2015), 3714–3727. (cited on page 4)

Duan, Y. and Jatowt, A., 2019. Across-time comparative summarization of news articles. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, 735–743. (cited on pages 5, 17, and 96)

Ebrahimi, J.; Dou, D.; and Lowd, D., 2016. A joint sentiment-target-stance model for stance classification in tweets. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 2656–2665. (cited on page 99)

Edmundson, H. P., 1969. New methods in automatic extracting. *Journal of the ACM (JACM)*, 16, 2 (1969), 264–285. (cited on page 11)

Erkan, G. and Radev, D. R., 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *J. Artif. Int. Res.*, 22, 1 (Dec. 2004), 457–479. (cited on pages 5, 10, 12, 14, 15, 57, 61, 77, 78, 85, 88, 89, and 92)

Fabbri, A.; Li, I.; She, T.; Li, S.; and Radev, D., 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*, 1074–1084. Association for Computational Linguistics, Florence, Italy. doi: 10.18653/v1/P19-1102. URL https://www.aclweb.org/anthology/P19-1102. (cited on pages 15, 88, and 89)

Galgani, F.; Compton, P.; and Hoffmann, A., 2012a. Citation based summarisation of legal texts. In *Pacific Rim International Conference on Artificial Intelligence*, 40–52. Springer. (cited on page 22)

Galgani, F.; Compton, P.; and Hoffmann, A., 2012b. Combining different summarization techniques for legal text. In *Proceedings of the Workshop on Innovative Hybrid Approaches to the Processing of Textual Data*, HYBRID '12 (Avignon, France, 2012), 115–123. Association for Computational Linguistics, USA. (cited on page 21)

Ganesan, K.; Zhai, C.; and Han, J., 2010. Opinosis: A graph based approach to abstractive summarization of highly redundant opinions. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, 340–348. Coling 2010 Organizing Committee, Beijing, China. URL https://www.aclweb.org/anthology/C10-1039. (cited on pages 21 and 22)

Garimella, K.; Morales, G. D. F.; Gionis, A.; and Mathioudakis, M., 2018. Quantifying controversy on social media. *Transactions on Social Computing*, (2018). (cited on pages 42 and 45)

Gelbukh, A.; Alexandrov, M.; Bourek, A.; and Makagonov, P., 2003. Selection of representative documents for clusters in a document collection. *Natural language processing and information systems*, (2003). (cited on page 57)

Genest, P.-E. and Lapalme, G., 2012. Fully abstractive approach to guided summarization. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 354–358. (cited on page 16)

Gillick, D. and Favre, B., 2009. A scalable global model for summarization. In *Proceedings of the Workshop on Integer Linear Programming for Natural Language Processing*, 10–18. Association for Computational Linguistics, Boulder, Colorado. URL https://www.aclweb.org/anthology/W09-1802. (cited on pages 12, 14, 15, 77, 78, and 87)

Gillick, D.; Favre, B.; Hakkani-Tür, D.; Bohnet, B.; Liu, Y.; and Xie, S., 2009. The icsi/utd summarization system at tac 2009. In *TAC*. (cited on pages 18, 78, 84, 85, 88, 89, 90, and 92)

Goldstein, J.; Kantrowitz, M.; Mittal, V.; and Carbonell, J., 1999. Summarizing text documents: Sentence selection and evaluation metrics. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99 (Berkeley, California, USA, 1999), 121–128. Association for Computing Machinery, New York, NY, USA. (cited on pages 11, 15, 23, 24, 77, and 78)

Gormley, C. and Tong, Z., 2015. *Elasticsearch: the definitive guide: a distributed real-time search and analytics engine*. " O'Reilly Media, Inc.". (cited on page 47)

Graham, T.; Bruns, A.; Angus, D.; Hurcombe, E.; and Hames, S., 2020. # istandwithdan versus# dictatordan: the polarised dynamics of twitter discussions about victoria's covid-19 restrictions. *Media International Australia*, (2020), 1329878X20981780. (cited on page 98)

Graham, Y., 2015. Re-evaluating automatic summarization with BLEU and 192 shades of ROUGE. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 128–137. Association for Computational Linguistics, Lisbon, Portugal. doi: 10.18653/v1/D15-1013. URL https://www.aclweb.org/anthology/D15-1013. (cited on pages 24 and 26)

Gretton, A.; Borgwardt, K. M.; Rasch, M. J.; Schölkopf, B.; and Smola, A., 2012a. A kernel two-sample test. 13, null (Mar. 2012), 723–773. (cited on pages 6, 29, 32, 33, 54, 76, 78, 80, and 102)

Gretton, A.; Sejdinovic, D.; Strathmann, H.; Balakrishnan, S.; Pontil, M.; Fukumizu, K.; and Sriperumbudur, B. K., 2012b. Optimal kernel choice for large-scale two-sample tests. In *Advances in neural information processing systems*, 1205–1213. Citeseer. (cited on page 119)

Grusky, M.; Naaman, M.; and Artzi, Y., 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)*, 708–719. Association for Computational Linguistics, New Orleans, Louisiana. doi: 10.18653/v1/N18-1065. URL https://www.aclweb.org/anthology/N18-1065. (cited on pages 21, 22, and 27)

Haghighi, A. and Vanderwende, L., 2009. Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 362–370. Association for Computational Linguistics, Boulder, Colorado. URL https://www.aclweb.org/anthology/N09-1041. (cited on pages 12, 14, 15, 55, and 78)

Hamborg, F.; Donnay, K.; and Gipp, B., 2019. Automated identification of media bias in news articles: an interdisciplinary literature review. *International Journal on Digital Libraries*, 20, 4 (2019), 391–415. (cited on page 98)

He, L.; Li, W.; and Zhuge, H., 2016. Exploring differential topic models for comparative summarization of scientific papers. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, 1028–1038. (cited on pages 17 and 28)

Hofstetter, C. R., 1976. *Bias in the news: Network television coverage of the 1972 election campaign*. The Ohio State University Press. (cited on page 98)

Hong, K.; Conroy, J.; Favre, B.; Kulesza, A.; Lin, H.; and Nenkova, A., 2014. A repository of state of the art and competitive baseline summaries for generic news summarization. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*, 1608–1616. European Languages Resources Association (ELRA), Reykjavik, Iceland. URL http://www.lrec-conf.org/proceedings/lrec2014/pdf/1093_Paper.pdf. (cited on pages 22, 25, 87, and 88)

Hong, K. and Nenkova, A., 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*, 712–721. Association for Computational Linguistics, Gothenburg, Sweden. doi: 10.3115/v1/E14-1075. URL https://www.aclweb.org/anthology/E14-1075. (cited on pages 12, 14, 15, 77, 78, 87, 88, and 89)

Hovy, E. H.; Lin, C.-Y.; Zhou, L.; and Fukumoto, J., 2006. Automated summarization evaluation with basic elements. In *LREC*, vol. 6, 899–902. Citeseer. (cited on pages 5 and 23)

Hsu, W.-T.; Lin, C.-K.; Lee, M.-Y.; Min, K.; Tang, J.; and Sun, M., 2018. A unified model for extractive and abstractive summarization using inconsistency loss. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 132–141. Association for Computational Linguistics, Melbourne, Australia. doi: 10.18653/v1/P18-1013. URL https://www.aclweb.org/anthology/P18-1013. (cited on page 27)

Hua, T.; Lu, C.-T.; Choo, J.; and Reddy, C. K., 2020. Probabilistic topic modeling for comparative analysis of document collections. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, 14, 2 (2020), 1–27. (cited on page 28)

Huang, X.; Wan, X.; and Xiao, J., 2011. Comparative news summarization using linear programming. In *Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. (cited on pages 5, 17, 54, 55, and 96)

Johnson, K. and Goldwasser, D., 2016. Identifying stance by analyzing political discourse on twitter. In *Proceedings of the First Workshop on NLP and Computational Social Science*, 66–75. (cited on pages 97, 99, and 100)

Jones, K. and Galliers, J., 1995. Evaluating natural language processing systems: An analysis and review. *Evaluating Natural Language Processing Systems*, (1995). (cited on pages 23 and 64)

Joulin, A.; Grave, E.; Bojanowski, P.; and Mikolov, T., 2016. Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*, (2016). (cited on page 66)

Kaufman, L. and Rousseeuw, P., 1987. *Clustering by means of medoids*. North-Holland. (cited on page 67)

Kedzie, C.; McKeown, K.; and Daumé III, H., 2018. Content selection in deep learning models of summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 1818–1828. Association for Computational Linguistics, Brussels, Belgium. doi: 10.18653/v1/D18-1208. URL https://www.aclweb.org/anthology/D18-1208. (cited on pages 14, 22, 79, 90, and 92)

Kempe, D.; Kleinberg, J.; and Tardos, E., 2003. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '03 (Washington, D.C., 2003), 137–146. Association for Computing Machinery, New York, NY, USA. doi: 10.1145/956750.956769. URL https://doi.org/10.1145/956750.956769. (cited on page 35)

Kim, B.; Khanna, R.; and Koyejo, O., 2016. Examples are not enough, learn to criticize! criticism for interpretability. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, NIPS'16 (Barcelona, Spain, 2016), 2288–2296. Curran Associates Inc., Red Hook, NY, USA. (cited on pages 4, 23, 24, 25, 26, 29, 34, 35, 37, 38, 54, 56, 60, 64, 65, 67, 72, 83, and 121)

Kim, B.; Kim, H.; and Kim, G., 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)*, 2519–2531. Association for Computational Linguistics, Minneapolis, Minnesota. doi: 10.18653/v1/N19-1260. URL https://www.aclweb.org/anthology/N19-1260. (cited on pages 21, 22, and 27)

Kim, B.; Rudin, C.; and Shah, J. A., 2014. The bayesian case model: A generative approach for case-based reasoning and prototype classification. *Advances in neural information processing systems*, 27 (2014). (cited on page 4)

Kim, G.; Moon, S.; and Sigal, L., 2015a. Joint photo stream and blog post summarization and exploration. In *IEEE Conference on Computer Vision and Pattern Recognition*. (cited on page 19)

Kim, H.; Choo, J.; Kim, J.; Reddy, C. K.; and Park, H., 2015b. Simultaneous discovery of common and discriminative topics via joint nonnegative matrix factorization. In *Proceedings*

of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 567–576. (cited on page 28)

Kingma, D. P. and Ba, J., 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, (2014). (cited on page 113)

Kirchhoff, K. and Bilmes, J., 2014. Submodularity for data selection in machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 131–141. (cited on page 38)

Kleppmann, M., 2017. *Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems*. " O'Reilly Media, Inc.". (cited on page 47)

Kornilova, A. and Eidelman, V., 2019. BillSum: A corpus for automatic summarization of US legislation. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, 48–56. Association for Computational Linguistics, Hong Kong, China. doi: 10.18653/v1/D19-5406. URL https://www.aclweb.org/anthology/D19-5406. (cited on pages 21 and 22)

Koupaee, M. and Wang, W. Y., 2018. Wikihow: A large scale text summarization dataset. *CoRR*, abs/1810.09305 (2018). URL http://arxiv.org/abs/1810.09305. (cited on page 21)

Kovach, B. and Rosenstiel, T., 2014. *The elements of journalism: What newspeople should know and the public should expect*. Three Rivers Press (CA). (cited on page 1)

Krause, A. and Golovin, D., 2014. Submodular function maximization. (cited on pages 20, 35, 36, 39, and 40)

Krause, A.; Leskovec, J.; Guestrin, C.; VanBriesen, J.; and Faloutsos, C., 2008. Efficient sensor placement optimization for securing large water distribution networks. *Journal of Water Resources Planning and Management*, 134, 6 (2008), 516–526. (cited on page 35)

Kreps, J.; Narkhede, N.; Rao, J.; et al., 2011. Kafka: A distributed messaging system for log processing. In *Proceedings of the NetDB*, vol. 11, 1–7. (cited on page 47)

Krikorian, R., 2013. New tweets per second record, and how! URL https://web.archive.org/web/20190212224215/https://blog.twitter.com/engineering/en_us/a/2013/new-tweets-per-second-record-and-how.html. (cited on page 1)

Kryscinski, W.; Keskar, N. S.; McCann, B.; Xiong, C.; and Socher, R., 2019. Neural text summarization: A critical evaluation. 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)*, 540–551. Association for Computational Linguistics, Hong Kong, China. doi: 10.18653/v1/D19-1051. URL https://www.aclweb.org/anthology/D19-1051. (cited on pages 26 and 27)

Kulesza, A. and Taskar, B., 2012. *Determinantal Point Processes for Machine Learning*. Now Publishers Inc., Hanover, MA, USA. ISBN 1601986289. (cited on pages 12, 14, 15, 77, 78, 81, 87, 88, and 89)

Kupiec, J.; Pedersen, J.; and Chen, F., 1999. A trainable document summarizer. (1999). (cited on page 12)

Lacoste-Julien, S.; Sha, F.; and Jordan, M. I., 2009. Disclda: Discriminative learning for dimensionality reduction and classification. In *Advances in neural information processing systems*, 897–904. (cited on page 28)

Lahoti, P.; Garimella, K.; and Gionis, A., 2018. Joint non-negative matrix factorization for learning ideological leaning on twitter. In *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*, 351–359. (cited on pages 99 and 100)

Landis, J. R. and Koch, G. G., 1977. The measurement of observer agreement for categorical data. *biometrics*, (1977). (cited on page 73)

Le, T. and Akoglu, L., 2019. Contravis: Contrastive and visual topic modeling for comparing document collections. In *The World Wide Web Conference*, 928–938. (cited on page 28)

Le, T. and Lauw, H. W., 2016. Word clouds with latent variable analysis for visual comparison of documents. (2016). (cited on page 28)

Lebanoff, L.; Song, K.; and Liu, F., 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*, 4131–4141. Association for Computational Linguistics, Brussels, Belgium. doi: 10.18653/v1/D18-1446. URL https://www.aclweb.org/anthology/D18-1446. (cited on page 27)

Li, C.; Liu, F.; Weng, F.; and Liu, Y., 2013. Document summarization via guided sentence compression. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 490–500. (cited on page 16)

Li, J.; Li, L.; and Li, T., 2012a. Multi-document summarization via submodularity. *Applied Intelligence*, 37, 3 (2012), 420–430. (cited on pages 4, 16, 17, 36, 54, and 55)

Li, L.; Zhou, K.; Xue, G.-R.; Zha, H.; and Yu, Y., 2009a. Enhancing diversity, coverage and balance for summarization through structure learning. In *International Conference on World Wide Web*. (cited on pages 12, 14, 54, and 55)

Li, P.; Ma, J.; and Gao, S., 2012b. Learning to summarize web image and text mutually. In *Proceedings of the 2Nd ACM International Conference on Multimedia Retrieval*. (cited on page 19)

Li, S.; Wang, W.; and Zhang, Y., 2009b. Tac 2009 update summarization of icl. In *TAC*. (cited on pages 87, 88, and 92)

Li, Y.; Swersky, K.; and Zemel, R., 2015. Generative moment matching networks. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15 (Lille, France, 2015), 1718–1727. JMLR.org. (cited on page 33)

Lin, C.-Y., 1999. Training a selection function for extraction. In *Proceedings of the eighth international conference on Information and knowledge management*, 55–62. ACM. (cited on page 12)

Lin, C.-Y., 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, 74–81. Association for Computational Linguistics, Barcelona, Spain. URL https://www.aclweb.org/anthology/W04-1013. (cited on pages 5, 23, 24, 25, 26, 54, 64, 73, 86, 87, and 116)

Lin, C.-Y. and Hovy, E., 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, 150–157. (cited on pages 25 and 54)

Lin, H. and Bilmes, J., 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*, 912–920. Association for Computational Linguistics, Los Angeles, California. URL https://www.aclweb.org/anthology/N10-1134. (cited on pages 12, 14, 17, 36, 40, 60, 77, 78, 83, and 86)

Lin, H. and Bilmes, J., 2011. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, 510–520. Association for Computational Linguistics, Portland, Oregon, USA. URL https://www.aclweb.org/anthology/P11-1052. (cited on pages 5, 12, 14, 15, 16, 35, 36, 37, 54, 56, 57, 77, and 78)

Lin, H. and Bilmes, J., 2012. Learning mixtures of submodular shells with application to document summarization. UAI'12 (Catalina Island, CA, 2012), 479–490. AUAI Press, Arlington, Virginia, USA. (cited on pages 12, 14, 15, 78, 88, and 89)

Lin, H.; Bilmes, J.; and Xie, S., 2009. Graph-based submodular selection for extractive summarization. In *2009 IEEE Workshop on Automatic Speech Recognition & Understanding*, 381–386. IEEE. (cited on pages 37, 59, and 60)

Liu, D. C. and Nocedal, J., 1989. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45, 1-3 (1989), 503–528. (cited on page 87)

Liu, F.; Ge, S.; and Wu, X., 2021. Competence-based multimodal curriculum learning for medical report generation. 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)*, 3001–3012. (cited on page 119)

Liu, Y. and Lapata, M., 2019a. Hierarchical transformers for multi-document summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 5070–5081. Association for Computational Linguistics, Florence, Italy. doi: 10.18653/v1/P19-1500. URL https://www.aclweb.org/anthology/P19-1500. (cited on pages 15, 26, and 27)

Liu, Y. and Lapata, M., 2019b. 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)*, 3730–3740. Association for Computational Linguistics, Hong Kong, China. doi: 10.18653/v1/D19-1387. URL https://www.aclweb.org/anthology/D19-1387. (cited on pages xix, 13, 15, 26, 27, 84, 86, 89, 90, and 119)

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V., 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, (2019). (cited on pages 104 and 113)

Louviere, J. J.; Flynn, T. N.; and Marley, A. A. J., 2015. *Best-worst scaling: Theory, methods and applications*. Cambridge University Press. (cited on page 26)

Luhn, H. P., 1958. The automatic creation of literature abstracts. *IBM Journal of research and development*, 2, 2 (1958), 159–165. (cited on page 11)

Mahasseni, B.; Lam, M.; and Todorovic, S., 2017. Unsupervised video summarization with adversarial lstm networks. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 202–211. (cited on page 19)

Mani, I., 2001. Summarization evaluation: An overview. In *Summarization evaluation: An overview*. (cited on page 25)

Mani, I. and Bloedorn, E., 1997. Multi-document summarization by graph search and matching. AAAI'97/IAAI'97 (Providence, Rhode Island, 1997), 622–628. AAAI Press. (cited on pages 11 and 64)

Mani, I.; House, D.; Klein, G.; Hirschman, L.; Firmin, T.; and Sundheim, B., 1999. The TIPSTER SUMMAC text summarization evaluation. In *Ninth Conference of the European Chapter of the Association for Computational Linguistics*, 77–85. Association for Computational Linguistics, Bergen, Norway. URL https://www.aclweb.org/anthology/E99-1011. (cited on pages 23, 25, 64, 65, and 106)

Mcauliffe, J. D. and Blei, D. M., 2008. Supervised topic models. In *Advances in neural information processing systems*, 121–128. (cited on page 28)

McKeown, K. and Radev, D. R., 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 (Seattle, Washington, USA, 1995), 74–82. Association for Computing Machinery, New York, NY, USA. doi: 10.1145/215206.215334. URL https://doi.org/10.1145/215206.215334. (cited on page 11)

Mendes, P. N.; Jakob, M.; García-Silva, A.; and Bizer, C., 2011. Dbpedia spotlight: Shedding light on the web of documents. In *Proceedings of the 7th International Conference on Semantic Systems*, I-Semantics '11 (Graz, Austria, 2011), 1–8. Association for Computing Machinery, New York, NY, USA. doi: 10.1145/2063518.2063519. URL https://doi.org/10.1145/2063518.2063519. (cited on pages 43, 47, and 85)

Mercer, J., 1909. Xvi. functions of positive and negative type, and their connection the theory of integral equations. *Philosophical transactions of the royal society of London. Series A, containing papers of a mathematical or physical character*, 209, 441-458 (1909), 415–446. (cited on page 30)

Meyer, R., 2016. How many stories do newspapers publish per day? URL https://web.archive.org/web/20190408131646/https://www.theatlantic.com/technology/archive/2016/05/how-many-stories-do-newspapers-publish-per-day/483845/. (cited on page 1)

Mintz, M.; Bills, S.; Snow, R.; and Jurafsky, D., 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, 1003–1011. Association for Computational Linguistics, Suntec, Singapore. URL https://www.aclweb.org/anthology/P09-1113. (cited on page 49)

Mirzasoleiman, B.; Badanidiyuru, A.; and Karbasi, A., 2016. Fast constrained submodular maximization: Personalized data summarization. In *International Conference on Machine Learning*. (cited on pages 35, 36, and 54)

Mirzasoleiman, B.; Jegelka, S.; and Krause, A., 2017. Streaming non-monotone submodular maximization: Personalized video summarization on the fly. *arXiv preprint arXiv:1706.03583*, (2017). (cited on pages 19 and 35)

Mitrovic, M.; Kazemi, E.; Zadimoghaddam, M.; and Karbasi, A., 2018. Data summarization at scale: A two-stage submodular approach. In *International Conference on Machine Learning*. (cited on pages 36, 54, and 56)

Mohammad, S.; Kiritchenko, S.; Sobhani, P.; Zhu, X.; and Cherry, C., 2016. SemEval-2016 task 6: Detecting stance in tweets. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, 31–41. Association for Computational Linguistics, San Diego, California. doi: 10.18653/v1/S16-1003. URL https://www.aclweb.org/anthology/S16-1003. (cited on pages 97, 99, 100, 112, and 113)

Morris, A. H.; Kasper, G. M.; and Adams, D. A., 1992. The effects and limitations of automated text condensing on reading comprehension performance. *Information Systems Research*, 3, 1 (1992), 17–35. (cited on pages 23, 25, and 26)

Muandet, K.; Fukumizu, K.; Sriperumbudur, B.; Schölkopf, B.; et al., 2017. Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends® in Machine Learning*, 10, 1-2 (2017), 1–141. (cited on pages 29, 31, 32, and 83)

Mullainathan, S. and Shleifer, A., 2002. Media bias. Technical report, National Bureau of Economic Research. (cited on pages 96 and 98)

Murray, G.; Kleinbauer, T.; Poller, P.; Becker, T.; Renals, S.; and Kilgour, J., 2009. Extrinsic summarization evaluation: A decision audit task. *ACM Transactions on Speech and Language Processing (TSLP)*, 6, 2 (2009), 1–29. (cited on page 25)

Nallapati, R.; Xiang, B.; and Zhou, B., 2016a. Sequence-to-sequence rnns for text summarization. *CoRR*, abs/1602.06023 (2016). (cited on page 119)

Nallapati, R.; Zhai, F.; and Zhou, B., 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, AAAI'17 (San Francisco, California, USA, 2017), 3075–3081. AAAI Press. (cited on pages 10, 13, 15, 26, 57, and 78)

Nallapati, R.; Zhou, B.; Gulcehre, C.; Xiang, B.; et al., 2016b. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*, (2016). (cited on pages 5, 10, 13, 15, 21, and 22)

Narayan, S.; Cohen, S. B.; and Lapata, M., 2018a. 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*, 1797–1807. Association for Computational Linguistics, Brussels, Belgium. doi: 10.18653/v1/D18-1206. URL https://www.aclweb.org/anthology/D18-1206. (cited on pages 21, 22, 27, and 119)

Narayan, S.; Cohen, S. B.; and Lapata, M., 2018b. Ranking sentences for extractive summarization with reinforcement learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 1747–1759. Association for Computational Linguistics, New Orleans, Louisiana. doi: 10.18653/v1/N18-1158. URL https://www.aclweb.org/anthology/N18-1158. (cited on pages 26 and 27)

Narayan, S.; Papasarantopoulos, N.; Cohen, S. B.; and Lapata, M., 2017. Neural extractive summarization with side information. *arXiv preprint arXiv:1704.04530*, (2017). (cited on page 27)

Narayanan, H., 1997. *Submodular functions and electrical networks*, vol. 54. Elsevier. (cited on page 35)

Nemhauser, G. L.; Wolsey, L. A.; and Fisher, M. L., 1978. An analysis of approximations for maximizing submodular set functions—i. *Mathematical Programming*, 14, 1 (1978), 265–294. (cited on pages 15, 35, 39, 78, and 83)

Nenkova, A.; Passonneau, R.; and McKeown, K., 2007. The pyramid method: Incorporating human content selection variation in summarization evaluation. *ACM Transactions on Speech and Language Processing*, (2007). (cited on pages 5, 23, 25, and 54)

Nenkova, A. and Vanderwende, L., 2005. The impact of frequency on summarization. *Microsoft Research, Redmond, Washington, Tech. Rep. MSR-TR-2005*, 101 (2005). (cited on pages 14, 15, and 78)

Neto, J. L.; Santos, A. D.; Kaestner, C. A.; Alexandre, N.; Santos, D.; et al., 2000. Document clustering and text summarization. (2000). (cited on page 85)

Nguyen, D. Q.; Vu, T.; and Nguyen, A. T., 2020. Bertweet: A pre-trained language model for english tweets. *arXiv preprint arXiv:2005.10200*, (2020). (cited on page 113)

Oelke, D.; Strobelt, H.; Rohrdantz, C.; Gurevych, I.; and Deussen, O., 2014. Comparative exploration of document collections: a visual analytics approach. In *Computer Graphics Forum*, vol. 33, 201–210. Wiley Online Library. (cited on page 28)

Over, P., 2001. Introduction to duc-2001: an intrinsic evaluation of generic news text summarization systems. (2001). (cited on pages 21, 23, and 25)

Over, P., 2003. An introduction to duc 2003: Intrinsic evaluation of generic news text summarization systems. In *Proceedings of Document Understanding Conference 2003*. (cited on pages 5, 16, and 21)

Over, P.; Dang, H.; and Harman, D., 2007. Duc in context. *Inf. Process. Manage.*, 43, 6 (Nov. 2007), 1506–1520. doi: 10.1016/j.ipm.2007.01.019. URL https://doi.org/10.1016/j.ipm.2007.01.019. (cited on pages 5, 16, 23, 25, and 26)

Over, P. and Liggett, W., 2002. Introduction to duc-2002: an intrinsic evaluation of generic news text summarization systems. (2002). (cited on pages 21 and 25)

Over, P. and Yen, J., 2004. An introduction to duc-2004. *National Institute of Standards and Technology*, (2004). (cited on pages 5, 16, and 21)

Owczarzak, K. and Dang, H. T., 2010. Overview of the tac 2010 summarization track. In *Proceedings of the Third Text Analysis Conference, Gaithersburg, Maryland, USA. National Institute of Standards and Technology. http://www.nist.gov/tac/publications*. (cited on pages 16 and 21)

Owczarzak, K. and Dang, H. T., 2011. Overview of the tac 2011 summarization track. In *Proceedings of the Fourth Text Analysis Conference, Gaithersburg, Maryland, USA. National Institute of Standards and Technology. http://www.nist.gov/tac/publications*. (cited on page 21)

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J., 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 311–318. (cited on pages 25 and 73)

Parveen, D.; Ramsl, H.-M.; and Strube, M., 2015. Topical coherence for graph-based extractive summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 1949–1954. Association for Computational Linguistics, Lisbon, Portugal. doi: 10.18653/v1/D15-1226. URL https://www.aclweb.org/anthology/D15-1226. (cited on page 27)

Patil, G. P. and Rao, C. R., 1978. Weighted distributions and size-biased sampling with applications to wildlife populations and human families. *Biometrics*, (1978), 179–189. (cited on page 80)

Patricia Aires, V.; G. Nakamura, F.; and F. Nakamura, E., 2019. A link-based approach to detect media bias in news websites. In *Companion Proceedings of The 2019 World Wide Web Conference*, 742–745. (cited on pages 49 and 98)

Paul, M., 2009. Cross-collection topic models: Automatically comparing and contrasting text. *Urbana*, 51 (2009), 61801. (cited on page 28)

Paulus, R.; Xiong, C.; and Socher, R., 2017. A deep reinforced model for abstractive summarization. *CoRR*, abs/1705.04304 (2017). (cited on pages 13, 15, and 27)

Pearson, K., 1901. Liii. on lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 2, 11 (1901), 559–572. (cited on page 31)

Pennington, J.; Socher, R.; and Manning, C., 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1532–1543. Association for Computational Linguistics, Doha, Qatar. doi: 10.3115/v1/D14-1162. URL https://www.aclweb.org/anthology/D14-1162. (cited on pages 66 and 104)

Peyrard, M., 2019. A simple theoretical model of importance for summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 1059–1073. Association for Computational Linguistics, Florence, Italy. doi: 10.18653/v1/P19-1101. URL https://www.aclweb.org/anthology/P19-1101. (cited on page 18)

Pilault, J.; Li, R.; Subramanian, S.; and Pal, C., 2020. On extractive and abstractive neural document summarization with transformer language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 9308–9319. (cited on pages 13 and 119)

Pöttker, H., 2003. News and its communicative quality: The inverted pyramid - when and why did it appear? *Journalism Studies*, (2003). (cited on pages 66 and 104)

Radev, D. R.; Jing, H.; Styś, M.; and Tam, D., 2004. Centroid-based summarization of multiple documents. *Information Processing & Management*, 40, 6 (2004), 919–938. (cited on pages 12, 14, 15, 77, and 78)

Raiber, F. and Kurland, O., 2010. On identifying representative relevant documents. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, 99–108. (cited on page 57)

Rajadesingan, A. and Liu, H., 2014. Identifying users with opposing opinions in twitter debates. In *International conference on social computing, behavioral-cultural modeling, and prediction*, 153–160. Springer. (cited on page 100)

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P., 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, (2016). (cited on page 6)

Reimers, N. and Gurevych, I., 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics. URL https://arxiv.org/abs/1908.10084. (cited on page 104)

Ren, X.; Lv, Y.; Wang, K.; and Han, J., 2017. Comparative document analysis for large text corpora. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, 325–334. (cited on page 28)

Ren, Z.; Inel, O.; Aroyo, L.; and De Rijke, M., 2016. Time-aware multi-viewpoint summarization of multilingual social text streams. In *ACM International on Conference on Information and Knowledge Management*. (cited on pages 18 and 55)

Ren, Z.; Liang, S.; Meij, E.; and de Rijke, M., 2013. Personalized time-aware tweets summarization. In *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*, 513–522. (cited on page 18)

Rieskamp, J., 2022. Contrastive argument summarization using supervised and unsupervised machine learning. (2022). (cited on pages 5 and 61)

Rochan, M.; Ye, L.; and Wang, Y., 2018. Video summarization using fully convolutional sequence networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 347–363. (cited on page 19)

Ross, B. H. and Makin, V. S., 1999. Prototype versus exemplar models in cognition. *The nature of cognition*, (1999), 205–241. (cited on page 4)

Rücklé, A. and Gurevych, I., 2017. Real-time news summarization with adaptation to media attention. In *Recent Advances in Natural Language Processing, RANLP*. (cited on pages 18 and 55)

Rush, A. M.; Chopra, S.; and Weston, J., 2015. A neural attention model for abstractive sentence summarization. *CoRR*, abs/1509.00685 (2015). (cited on pages 13, 15, 21, and 22)

Saggion, H.; Radev, D.; Teufel, S.; and Lam, W., 2002. Meta-evaluation of summaries in a cross-lingual environment using content-based metrics. In *COLING 2002: The 19th International Conference on Computational Linguistics*. URL https://www.aclweb.org/anthology/C02-1073. (cited on pages 23 and 24)

Sakai, T. and SparckJones, K., 2001. Generic summaries for indexing in information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, 190–198. ACM. (cited on page 1)

Salton, G.; Singhal, A.; Mitra, M.; and Buckley, C., 1997. Automatic text structuring and summarization. *Information Processing & Management*, 33, 2 (1997), 193–207. (cited on page 1)

Sandhaus, E., 2008. The new york times annotated corpus. *Linguistic Data Consortium, Philadelphia*, 6, 12 (2008), e26752. (cited on pages 21 and 22)

Sanh, V.; Debut, L.; Chaumond, J.; and Wolf, T., 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, (2019). (cited on page 104)

Savov, P.; Jatowt, A.; and Nielek, R., 2020. Identifying breakthrough scientific papers. *Information Processing & Management*, 57, 2 (2020), 102168. (cited on page 60)

Schiffman, B. and McKeown, K., 2005. Context and learning in novelty detection. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, 716–723. (cited on page 18)

Schroff, F.; Kalenichenko, D.; and Philbin, J., 2015. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 815–823. (cited on page 104)

See, A.; Liu, P. J.; and Manning, C. D., 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*, (2017). (cited on pages 13 and 15)

Sejdinovic, D. and Gretton, A., 2012. What is an rkhs? *Lecture Notes*, (2012). (cited on pages 29, 30, and 31)

Shahaf, D.; Guestrin, C.; and Horvitz, E., 2012. Trains of thought: Generating information maps. In *Proceedings of the 21st international conference on World Wide Web*, 899–908. ACM. (cited on page 18)

Sharma, D.; Deshpande, A.; and Kapoor, A., 2015. On greedy maximization of entropy. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15 (Lille, France, 2015), 1330–1338. JMLR.org. (cited on page 37)

Sharma, E.; Huang, L.; Hu, Z.; and Wang, L., 2019a. An entity-driven framework 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)*, 3280–3291. Association for Computational Linguistics, Hong Kong, China. doi: 10.18653/v1/D19-1323. URL https://www.aclweb.org/anthology/D19-1323. (cited on page 27)

Sharma, E.; Li, C.; and Wang, L., 2019b. BIGPATENT: A large-scale dataset for abstractive and coherent summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2204–2213. Association for Computational Linguistics, Florence, Italy. doi: 10.18653/v1/P19-1212. URL https://www.aclweb.org/anthology/P19-1212. (cited on pages 21 and 22)

Shen, D.; Sun, J.-T.; Li, H.; Yang, Q.; and Chen, Z., 2007. Document summarization using conditional random fields. In *IJCAI*, vol. 7, 2862–2867. (cited on page 12)

Shin, M.; Kim, D.; Lee, J. H.; Bista, U.; and Xie, L., 2019. Visualizing graph differences from social media streams. WSDM '19, 806–809. ACM. doi: 10.1145/3289600.3290616. URL http://doi.acm.org/10.1145/3289600.3290616. (cited on page 28)

Shorten, C. and Khoshgoftaar, T. M., 2019. A survey on image data augmentation for deep learning. *Journal of Big Data*, 6, 1 (2019), 1–48. (cited on page 120)

Simon, I.; Snavely, N.; and Seitz, S. M., 2007. Scene summarization for online image collections. In *International Conference on Computer Vision*. (cited on pages 18, 37, and 56)

Smith, L. M.; Zhu, L.; Lerman, K.; and Kozareva, Z., 2013. The role of social media in the discussion of controversial topics. In *2013 International Conference on Social Computing*, 236–243. IEEE. (cited on page 42)

Smola, A.; Gretton, A.; Song, L.; and Schölkopf, B., 2007. A hilbert space embedding for distributions. In *International Conference on Algorithmic Learning Theory*, 13–31. Springer. (cited on page 32)

Soboroff, I. and Harman, D., 2003. Overview of the trec 2003 novelty track. In *TREC*, 38–53. Citeseer. (cited on page 16)

Sriperumbudur, B. K.; Fukumizu, K.; Gretton, A.; Lanckriet, G. R. G.; and Schölkopf, B., 2009a. Kernel choice and classifiability for rkhs embeddings of probability distributions. In *Proceedings of the 22nd International Conference on Neural Information Processing Systems*, NIPS'09 (Vancouver, British Columbia, Canada, 2009), 1750–1758. Curran Associates Inc., Red Hook, NY, USA. (cited on pages 34, 59, 82, and 102)

Sriperumbudur, B. K.; Fukumizu, K.; Gretton, A.; Schölkopf, B.; and Lanckriet, G. R., 2009b. On integral probability metrics,\phi-divergences and binary classification. *arXiv preprint arXiv:0901.2698*, (2009). (cited on page 33)

Sriperumbudur, B. K.; Gretton, A.; Fukumizu, K.; Schölkopf, B.; and Lanckriet, G. R., 2010. Hilbert space embeddings and metrics on probability measures. *Journal of Machine Learning Research*, 11, Apr (2010), 1517–1561. (cited on page 83)

Stefanov, P.; Darwish, K.; Atanasov, A.; and Nakov, P., 2020. Predicting the topical stance and political leaning of media using tweets. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 527–537. (cited on pages 49, 98, 99, and 100)

Steinwart, I., 2001. On the influence of the kernel on the consistency of support vector machines. *Journal of machine learning research*, 2, Nov (2001), 67–93. (cited on page 83)

Storms, G.; De Boeck, P.; and Ruts, W., 2000. Prototype and exemplar-based information in natural language categories. *Journal of Memory and Language*, 42, 1 (2000), 51–73. (cited on page 4)

Sun, Q.; Wang, Z.; Zhu, Q.; and Zhou, G., 2018. Stance detection with hierarchical attention network. In *Proceedings of the 27th international conference on computational linguistics*, 2399–2409. (cited on page 99)

Szegedy, C.; Ioffe, S.; Vanhoucke, V.; and Alemi, A., 2017. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 31. (cited on page 46)

Tombros, A. and Sanderson, M., 1998. Advantages of query biased summaries in information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, 2–10. (cited on page 64)

Trabelsi, A. and Zaiane, O., 2018. Unsupervised model for topic viewpoint discovery in online debates leveraging author interactions. In *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 12. (cited on page 100)

Truong, Q.-T. and Lauw, H., 2019. Multimodal review generation for recommender systems. In *The World Wide Web Conference*, 1864–1874. (cited on page 119)

Tschiatschek, S.; Iyer, R. K.; Wei, H.; and Bilmes, J. A., 2014. Learning mixtures of submodular functions for image collection summarization. In *Advances in neural information processing systems*, 1413–1421. (cited on pages 18, 35, 36, 37, 57, and 59)

Varma, V.; Pingali, P.; Katragadda, R.; Krishna, S.; Ganesh, S.; Sarvabhotla, K.; Garapati, H.; Gopisetty, H.; Reddy, V. B.; Reddy, K.; et al., 2009. Iiit hyderabad at tac 2009. In *TAC*. (cited on pages 14, 18, 77, 78, 87, 88, 90, and 92)

Verkamp, J.-P. and Gupta, M., 2013. Five incidents, one theme: Twitter spam as a weapon to drown voices of protest. In *USENIX Workshop on Free and Open Communications on the Internet*. (cited on pages 43 and 45)

Vinyals, O.; Fortunato, M.; and Jaitly, N., 2015. Pointer networks. *arXiv preprint arXiv:1506.03134*, (2015). (cited on page 13)

Völske, M.; Potthast, M.; Syed, S.; and Stein, B., 2017. TL;DR: Mining Reddit to learn automatic summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, 59–63. Association for Computational Linguistics, Copenhagen, Denmark. doi: 10.18653/v1/W17-4508. URL https://www.aclweb.org/anthology/W17-4508. (cited on pages 21 and 22)

Wan, X.; Yang, J.; and Xiao, J., 2007. Towards an iterative reinforcement approach for simultaneous document summarization and keyword extraction. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, 552–559. Association for Computational Linguistics, Prague, Czech Republic. URL https://www.aclweb.org/anthology/P07-1070. (cited on page 85)

Wang, D.; Zhu, S.; Li, T.; and Gong, Y., 2012. Comparative document summarization via discriminative sentence selection. *ACM Transactions on Knowledge Discovery from Data*, (2012). (cited on pages 4, 17, 54, and 55)

Wang, Z.; Duan, Z.; Zhang, H.; Wang, C.; Tian, L.; Chen, B.; and Zhou, M., 2020. Friendly topic assistant for transformer based abstractive summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 485–497. (cited on page 13)

Wei, K.; Iyer, R.; and Bilmes, J., 2015. Submodularity in data subset selection and active learning. In *International Conference on Machine Learning*. (cited on pages 35, 37, 54, 56, 58, 59, and 61)

Wei, W.; Zhang, X.; Liu, X.; Chen, W.; and Wang, T., 2016. pkudblab at semeval-2016 task 6: A specific convolutional neural network system for effective stance detection. In *Proceedings of the 10th international workshop on semantic evaluation (SemEval-2016)*, 384–388. (cited on page 99)

White, D. M., 1950. The "gate keeper": A case study in the selection of news. *Journalism quarterly*, 27, 4 (1950), 383–390. (cited on page 98)

Williams, A., 1975. Unbiased study of television news bias. *Journal of Communication*, 25, 4 (1975), 190–199. (cited on page 98)

Xu, H.; Wang, J.; Hua, X.-S.; and Li, S., 2011. Hybrid image summarization. In *Proceedings of the 19th ACM International Conference on Multimedia*, MM '11 (Scottsdale, Arizona, USA, 2011), 1217–1220. Association for Computing Machinery, New York, NY, USA. doi: 10.1145/2072298.2071978. URL https://doi.org/10.1145/2072298.2071978. (cited on pages 18 and 57)

Xu, J. and Durrett, G., 2019. Neural extractive text summarization with syntactic compression. *arXiv preprint arXiv:1902.00863*, (2019). (cited on page 27)

Yan, R.; Wan, X.; Otterbacher, J.; Kong, L.; Li, X.; and Zhang, Y., 2011. Evolutionary timeline summarization: a balanced optimization framework via iterative substitution. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, 745–754. (cited on page 18)

Yasunaga, M.; Zhang, R.; Meelu, K.; Pareek, A.; Srinivasan, K.; and Radev, D., 2017. Graph-based neural multi-document summarization. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, 452–462. Association for Computational Linguistics, Vancouver, Canada. doi: 10.18653/v1/K17-1045. URL https://www.aclweb.org/anthology/K17-1045. (cited on pages 10, 13, 14, 21, 22, 27, 57, 77, 78, 84, 86, 87, 88, 89, and 120)

Zaheer, M.; Reddi, S.; Sachan, D.; Kale, S.; and Kumar, S., 2018. Adaptive methods for nonconvex optimization. In *Advances in Neural Information Processing Systems 31* (Eds. S. Bengio; H. Wallach; H. Larochelle; K. Grauman; N. Cesa-Bianchi; and R. Garnett), 9793–9803. Curran Associates, Inc. URL http://papers.nips.cc/paper/8186-adaptive-methods-for-nonconvex-optimization.pdf. (cited on page 87)

Zandt, D. V., 2015. Media bias/fact check. URL https://mediabiasfactcheck.com. (cited on pages xviii, 48, and 49)

Zarrella, G. and Marsh, A., 2016. Mitre at semeval-2016 task 6: Transfer learning for stance detection. *arXiv preprint arXiv:1606.03784*, (2016). (cited on pages 99 and 113)

Zhang, F., 2006. *The Schur complement and its applications*, vol. 4. Springer Science & Business Media. (cited on page 40)

Zhang, J.; Du, P.; Xu, H.; and Cheng, X., 2009. Ictgrasper at TAC2009: temporal preferred update summarization. In *Proceedings of the Second Text Analysis Conference, TAC 2009, Gaithersburg, Maryland, USA, November 16-17, 2009*. (cited on pages 78, 87, 88, and 90)

Zhang, J.; Zhao, Y.; Saleh, M.; and Liu, P., 2020. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, 11328–11339. PMLR. (cited on pages 26 and 27)

Zhang, L.; Li, L.; Shen, C.; and Li, T., 2015. Patentcom: A comparative view of patent document retrieval. In *Proceedings of the 2015 SIAM International Conference on Data Mining*, 163–171. SIAM. (cited on pages 1, 5, 17, and 96)

Zhang, R. and Tetreault, J., 2019. This email could save your life: Introducing the task of email subject line generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 446–456. Association for Computational Linguistics, Florence, Italy. doi: 10.18653/v1/P19-1043. URL https://www.aclweb.org/anthology/P19-1043. (cited on pages 21, 22, and 27)

Zhang, R.; You, O.; and Li, W., 2011. Guided summarization with aspect recognition. In *TAC*. (cited on page 16)

Zhang, X.; Wei, F.; and Zhou, M., 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*, 5059–5069. Association for Computational Linguistics, Florence, Italy. doi: 10.18653/v1/P19-1499. URL https://www.aclweb.org/anthology/P19-1499. (cited on page 27)

Zhao, L.; Wu, L.; and Huang, X., 2009. Using query expansion in graph-based approach for query-focused multi-document summarization. *Information processing & management*, 45, 1 (2009), 35–41. (cited on page 16)

Zhou, Q.; Yang, N.; Wei, F.; Huang, S.; Zhou, M.; and Zhao, T., 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)*, 654–663. Association for Computational Linguistics, Melbourne, Australia. doi: 10.18653/v1/P18-1061. URL https://www.aclweb.org/anthology/P18-1061. (cited on page 27)

Zhu, J.; Li, H.; Liu, T.; Zhou, Y.; Zhang, J.; and Zong, C., 2018. Msmo: Multimodal summarization with multimodal output. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, 4154–4164. (cited on page 119)

Zhu, J.; Zhou, Y.; Zhang, J.; Li, H.; Zong, C.; and Li, C., 2020. Multimodal summarization with guidance of multimodal reference. In *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, 9749–9756. (cited on page 119)