Summarizing User-generated Textual Content: Motivation and Methods for Fairness in Algorithmic Summaries

ABHISEK DASH, Indian Institute of Technology Kharagpur, India
ANURAG SHANDILYA, Indian Institute of Technology Kharagpur, India
ARINDAM BISWAS, Indian Institute of Technology Kharagpur, India
KRIPABANDHU GHOSH, Tata Research Development and Design Centre, India
SAPTARSHI GHOSH, Indian Institute of Technology Kharagpur, India
ABHIJNAN CHAKRABORTY, Max Planck Institute for Software Systems, Germany

As the amount of user-generated textual content grows rapidly, text summarization algorithms are increasingly being used to provide users a quick overview of the information content. Traditionally, summarization algorithms have been evaluated only based on how well they match human-written summaries (e.g. as measured by ROUGE scores). In this work, we propose to evaluate summarization algorithms from a completely new perspective that is important when the user-generated data to be summarized comes from different socially salient user groups, e.g. men or women, Caucasians or African-Americans, or different political groups (Republicans or Democrats). In such cases, we check whether the generated summaries *fairly represent* these different social groups. Specifically, considering that an *extractive* summarization algorithm selects a subset of the textual units (e.g. microblogs) in the original data for inclusion in the summary, we investigate whether this selection is *fair* or not. Our experiments over real-world microblog datasets show that existing summarization algorithms often represent the socially salient user-groups very differently compared to their distributions in the original data. More importantly, some groups are frequently *under-represented* in the generated summaries, and hence get far less exposure than what they would have obtained in the original data. To reduce such adverse impacts, we propose novel fairness-preserving summarization algorithms which produce high-quality summaries while ensuring fairness among various groups. To our knowledge, this is the first attempt to produce fair text summarization, and is likely to open up an interesting research direction.

CCS Concepts: • **Information systems → Summarization**; • **Human-centered computing → Social media**.

Additional Key Words and Phrases: Text summarization; Extractive summarization; Fair summarization; Fairness in algorithmic decision making; Group fairness

We thank the anonymous reviewers whose suggestions helped to improve the paper. We acknowledge the human annotators who developed the gold standard summaries for the datasets used in this study. This research was supported in part by a European Research Council (ERC) Advanced Grant for the project "Foundations for Fair Social Computing", funded under the European Union's Horizon 2020 Framework Programme (grant agreement no. 789373). A Dash is supported by a Fellowship from Tata Consultancy Services.

Authors' addresses: Abhisek Dash, Indian Institute of Technology Kharagpur, India; Anurag Shandilya, Indian Institute of Technology Kharagpur, India; Arindam Biswas, Indian Institute of Technology Kharagpur, India; Kripabandhu Ghosh, Tata Research Development and Design Centre, India; Saptarshi Ghosh, Indian Institute of Technology Kharagpur, India; Abhijnan Chakraborty, Max Planck Institute for Software Systems, Germany.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.
2573-0142/2019/11-ART172 $15.00
https://doi.org/10.1145/3359274

ACM Reference Format:
Abhisek Dash, Anurag Shandilya, Arindam Biswas, Kripabandhu Ghosh, Saptarshi Ghosh, and Abhijnan Chakraborty. 2019. Summarizing User-generated Textual Content: Motivation and Methods for Fairness in Algorithmic Summaries. *Proc. ACM Hum.-Comput. Interact.* 3, CSCW, Article 172 (November 2019), 28 pages. https://doi.org/10.1145/3359274

1 INTRODUCTION

Recently, there has been an explosion in the amount of user-generated information on the Web. To help Web users deal with the information overload, text summarization algorithms are commonly used to get a quick overview of the textual information. Recognizing the business opportunities, many startups have mushroomed recently to offer content summarization services. For example, Agolo (`agolo.com/splash`) provides a summarization platform to get the most relevant information from both public and private documents. Aylien (`aylien.com/text-api/summarization`) or Resoomer (`resoomer.com`) present relevant points and topics from a piece of text. Multiple smartphone apps (e.g. News360, InShorts) have also been launched to provide short summaries of news stories.

A large number of text summarization algorithms have been devised, including algorithms to summarize a single large document, as well as for summarizing a set of documents (e.g. a set of microblogs or tweets); interested readers can check [3] for a survey on summarization algorithms. Most of these summarization algorithms are *extractive* in nature, i.e. they form the summary by extracting some of the textual units in the input [31] (e.g. individual sentences in a document, or individual tweets in a set of tweets). Additionally, some *abstractive* algorithms have also been devised, that attempt to generate natural language summaries [3]. In this paper, we restrict our focus to the more prevalent extractive summarization.

Extractive summarization algorithms essentially perform a selection of a (small) subset of the textual units in the input, for inclusion in the summary, based on some measure of the relative quality or importance of the textual units. Traditionally, these algorithms are judged based on how closely the algorithmic summary matches gold standard summaries that are usually written by human annotators. To this end, measures such as ROUGE scores are used to evaluate the goodness of algorithmic summaries [41]. The underlying assumption behind this traditional evaluation criteria is that *the data to be summarized is homogeneous, and the sole focus of summarization algorithms should be to identify summary-worthy information.*

However, user-generated content constitutes a large chunk of information generated on the Web today, and such content is often heterogeneous, coming from users belonging to different social groups. For example, on social media, different user groups (e.g. men and women, Republicans and Democrats) discuss socio-political issues, and it has been observed that different social groups often express very different opinions on the same topic or event [15]. Hence, while summarizing such heterogeneous user-generated data, one needs to check whether the summaries are properly representing the opinions of these different social groups. Since the textual units (e.g. tweets) that are included in the summary get much more exposure than the rest of the information (similar to how top-ranked search results get much more exposure than other documents [8, 68]), if a particular group is under-represented in the summary, their opinion will get much less exposure than the opinion of other groups.

Therefore, in this paper, we propose to look at summarization algorithms from a completely new perspective, and investigate whether the selection of the textual units in the summary is fair, i.e. *whether the generated summary fairly represents every social group in the input data.* We experiment with three datasets of tweets generated by different user groups (men and women, pro-republican and pro-democratic users). We find that most existing summarization algorithms do *not* fairly

represent different groups in the generated summaries, even though the tweets written by these groups are of comparable textual quality. More worryingly, some groups are found to be systemically under-represented in the process. Note that we, by no means, claim such under-representation to be intentionally caused by the existing algorithms. Rather it is most likely an inadvertent perpetuation of the metrics that the algorithms are trying to optimize. Since the applications of summarization algorithms may extend from product reviews to citizen journalism, the question of whether existing algorithms are fair and how we can potentially improve them become even more important.

Having observed that existing summarization algorithms do not give fair summaries in most cases, we next attempt to develop algorithms for fair summarization. Recently, there have been multiple research works attempting to incorporate fairness in machine learning algorithms [24, 33, 38]. Primarily, there are three ways in which these research works make fairness interventions in an existing system – *pre-processing, in-processing* and *post-processing*, depending on whether the interventions are applied at the input, algorithm or the output stage [26]. Following this line of work, in this paper, we develop three novel fairness-preserving summarization algorithms which select highly relevant textual units in the summary while maintaining fairness in the process. Our proposed in-processing algorithm is based on constrained sub-modular optimization (where the fairness criteria are applied as constraints). The post-processing algorithm is based on fair ranking of textual units based on some goodness measure, and the pre-processing approach groups the tweets on the basis of their association to different classes, and then summarizes each group separately to generate fair summaries. Extensive evaluations show that our proposed algorithms are able to generate summaries having quality comparable to state-of-the-art summarization algorithms (which often do not generate fair summaries), while being fair to different user groups.

In summary, we make the following contributions in this paper: (1) ours is one of the first attempts to consider the notion of fairness in summarization, and the first work on fair summarization of textual information; (2) we show that, while summarizing content generated by different user groups, existing summarization algorithms often do not represent the user groups fairly; and (3) we propose summarization algorithms that produce summaries that are of good quality as well as fair according to different fairness notions, including equal representation, proportional representation, and so on. We have made the implementation of our fair summarization algorithms and our datasets publicly available at https://github.com/ad93/FairSumm.

We believe that this work will be an important addition to the growing literature on incorporating fairness in algorithmic systems. Generation of fair summaries would not only benefit the end users of the summaries, but also many downstream applications that use the summaries of crowdsourced information, e.g., summary-based opinion classification and rating inference systems [44].

The rest of the paper is structured as follows. Section 2 gives a background on summarization and discusses related works. Section 3 describes the datasets we use throughout the paper. Thereafter, we motivate the need for fair summarization in Section 4. Section 5 introduces some possible notions of fairness in summarization, and Section 6 shows how existing text summarization algorithms do not adhere to these fairness notions. In Section 7, we discuss a principled framework for achieving fairness in summarization, followed by details of three fair summarization algorithms in Sections 8 and 9. We evaluate the performance of our proposed algorithms in Section 10. Finally, we conclude the paper, discussing some limitations of the proposed algorithms and possible future directions.

2 BACKGROUND AND RELATED WORK

In this section, we discuss two strands of prior works that are relevant to our paper. First, we focus on text summarization. Then, we relate this paper to prior works on bias and fairness in information systems.

2.1 Text Summarization

Text summarization is a well-studied problem in Natural Language Processing, where the task is to produce a fluent and informative summary given a piece of text or a collection of text documents. A large number of text summarization algorithms have been proposed in literature; the reader can refer to [3, 31] for surveys. As discussed in the introduction, there are two variants of summarization algorithms – extractive and abstractive summarization algorithms. While most classical summarization algorithms were unsupervised, the recent years have seen the proliferation of many supervised neural network-based models for summarization; the reader can refer to [21] for a survey on neural summarization models. To contextualise our work, next we discuss different types of extractive text summarization algorithms in the literature.

Single-Document Summarization: Traditional single document extractive summarization deals with extraction of useful information from a single document. A series of single-document summarization algorithms have been proposed [25, 28, 29, 36, 45, 50, 51]. We will describe some of these algorithms in Section 6. One of the most commonly used class of summarization algorithms is centered around the popular TF-IDF model [61]. Different works have used TF-IDF based similarities for summarization [2, 57]. Additionally, there has been a series of works where summarization has been treated as a sub-modular optimization problem [5, 43]. One of the fair summarization algorithms proposed in this work, is also based on a sub-modular constrained optimization framework, and uses the notion of TF-IDF similarity.

Multi-Document Summarization: Multi-document extractive summarization deals with extraction of information from multiple documents (pieces of text) written about the same topic. For instance, NeATS [42] is a multi-document summarization system that, given a collection of newspaper articles as input, generates a summary in three stages – content selection, filtering, and presentation. Hub/Authority [72] is another multi-document summarization system which uses the Markov Model to order the sub-topics that the final summary should contain, and then outputs the summary according to the sentence ranking score of all sentences within one sub-topic. Generic Relation Extraction (GRE) [32] is another multi-document text summarization approach, which aims to build systems for relation identification and characterization that can be transferred across domains and tasks without modification of model parameters. Celikyilmaz *et al.* [11] described multi-document summarization as a prediction problem based on a two-phase hybrid model and proposed a hierarchical topic model to discover the topic structures of all sentences. Wong *et al.* [66] proposed a semi-supervised method for extractive summarization, by co-training two classifiers iteratively. In each iteration, the unlabeled training sentences with top scores are included in the labeled training set, and the classifiers are trained on the new training data.

Summarization of User Generated Text on Social Media: With the proliferation of user generated textual content on social media (e.g., Twitter, Facebook), a number of summarization algorithms have been developed specifically for such content. For instance, Carenini *et al.* [10] proposed a novel summarization algorithm that summarizes e-mail conversations using fragment quotation graph and clue words. Nichols *et al.* [52] described an algorithm that generates a journalistic summary of an event using only status updates from Twitter as information source. They used temporal cues to find important moments within an event and a sentence ranking method to extract the most relevant sentences describing the event. Rudra *et al.* [60] proposed a summarization algorithm for tweets posted during disaster events. Kim *et al.* [37] used narrative theory as a framework for identifying the links between social media content and designed crowdsourcing tasks to generate summaries of events based on commonly used narrative templates. Zhang *et al.* [71] proposed a recursive summarization workflow where they design a summary tree that enables readers to digest the entire abundance of posts. Zhang *et al.* [70] developed Tilda, which allows participants

of a discussion to collectively tag, group, link, and summarize chat messages in a variety of ways, such as by adding emoji reactions to messages or leaving written notes.

2.2 Bias and Fairness in Information Filtering Algorithms

Bias in applications on user-generated content: Powerful computational resources along with the enormous amount of data from social media sites has driven a growing school of works that uses a combination of machine learning, natural language processing, statistics and network science for decision making. In [6], Baeza-Yates has discussed how human perceptions and societal biases creep into social media, and how different algorithms fortify them. These observations raise questions of bias in the decisions derived from such analyses. Friedman *et al.* [27] broadly categorized these biases into 3 different classes, and essentially were the first to propose a framework for comprehensive understanding of the biases. Several recent works have investigated different types of biases (demographic, ranking, position biases etc.) and their effects on online social media [9, 14, 15]. Our observations in this work show that summaries generated by existing algorithms (which do not consider fairness) can lead to biases towards/against socially salient demographic groups.

Rooney Rule: The notion of implicit bias has been an important component in understanding discrimination in activities such as hiring, promotion, and school admissions. Research on implicit bias hypothesizes that when people evaluate others – e.g., while hiring for a job – their unconscious biases about membership in particular groups can have an effect on their decision-making, even when they have no deliberate intention to discriminate against members of these groups. To this end, the *Rooney Rule* was proposed hoping to reduce the adverse effects of such implicit biases. The Rooney Rule is a National Football League policy in the USA, that requires league teams to interview ethnic-minority candidates for head coaching and senior football operation jobs. Roughly speaking, it requires that while recruiting for a job opening, one of the candidates interviewed must come from an underrepresented group. As per [19], there are two variants of the Rooney rule. The 'soft' affirmative action programs encompass outreach attempts like minority recruitment and counseling etc., while the 'hard' affirmative action programs usually include explicit preferences or quotas that reserve a specific number of openings exclusively for members of the preferred group.

In the context of summarization, any summarization algorithm will adhere to the 'soft' variant of Rooney Rule, since all the textual units (be it from majority or minority groups) have candidature to enter the summary. However, existing summarization algorithms are *not* guaranteed to adhere to the 'hard' variant of the Rooney rule. The algorithms proposed in this paper (detailed in Sections 8 and 9) are guaranteed to also cohere to the 'hard' variant of the Rooney Rule since they maintain a specific level of representation of various social groups in the final summary.

Fairness in information filtering algorithms: Given that information filtering algorithms (search, recommendation, summarization algorithms) have far-reaching social and economic consequences in today's world, fairness and anti-discrimination have been recent inclusions in the algorithm design perspective [24, 33, 38]. There have been several recent works on defining and achieving different notions of fairness [35, 39, 67, 69] as well as on removing the existing unfairness from different methodologies [34, 40, 69]. Different fairness-aware algorithms have been proposed to achieve group and/or individual fairness for tasks such as clustering [17], classification [67], ranking [68], matching [65], recommendation [16] and sampling [12].

To our knowledge, only two prior works have looked into fairness in summarization. Celis *et al.* proposed a methodology to obtain fair and diverse summaries [13]. They applied their determinantal point process based algorithm on an image dataset and a categorical dataset (having several attributes), and not on textual data. The problem of fair *text* summarization was first introduced

in our prior work [62], which showed that many existing text summarization algorithms do not generate fair summaries; however, no algorithm for fair summarization was proposed in [62]. To our knowledge, ours is the first work to propose algorithms for fair summarization of textual data.

3 DATASETS USED

Since our focus in this paper is to understand the need for fairness while summarizing user-generated content, we consider datasets containing tweets posted by different groups of users, e.g. different gender groups, or groups of users with different political leanings. Specifically, we use the following three datasets throughout this paper.

(1) **Claritin dataset**: Patients undergoing medication often post the consequences of using different drugs on social media, especially highlighting the side-effects they endure [53]. Claritin (loratadine) is an anti-allergic drug that reduces the effects of natural chemical histamine in the body, which can produce symptoms of sneezing, itching, watery eyes and runny nose. However, this drug may also have some adverse effects on the patients.

To understand the sentiments of people towards Claritin and different side-effects caused by it, tweets posted by users about Claritin were collected, analyzed and later publicly released by 'Figure Eight' (erstwhile CrowdFlower). This dataset contains tweets in English about the effects of the drug. Each tweet is annotated with the gender of the user (male/female/unknown) posting it [18]. Initial analyses on these tweets reveal that women mentioned some serious side effects of the drug (e.g. heart palpitations, shortness of breathe, headaches) while men did not [18]. From this dataset, we ignored those tweets for which the gender of the user is unknown. We also removed exact duplicate tweets, since they do not have any meaningful role in summarization. Finally, we have 4,037 tweets in total, of which 1,532 (37.95%) are written by men, and 2,505 (62.05%) by women.

(2) **US-Election dataset**: This dataset consists of tweets related to the 2016 US Presidential Election collected by the website TweetElect (https://badrit.com/work/Tweetelect) during the period from September 1, 2016 to November 8, 2016 (the election day) [20]. TweetElect used an initial set of 38 keywords related to the election (including all candidate names and common hashtags about the participating parties) for filtering relevant tweets. Subsequently, state-of-the-art adaptive filtering methods were used to expand the set of keywords with additional terms that emerged over time [47], and their related tweets were added to the collection.

In this dataset released by Darwish *et al.* [20], each tweet is annotated as supporting or attacking one of the presidential candidates (Donald Trump and Hillary Clinton) or neutral or attacking both. For simplicity, we grouped the tweets into three classes: (i) *Pro-Republican*: tweets which support Trump and / or attack Clinton, (ii) *Pro-Democratic*: tweets which support Clinton and / or attack Trump, and (iii) *Neutral*: tweets which are neutral or attack both candidates. After removing duplicates, we have 2,120 tweets, out of which 1,309 (61.74%) are Pro-Republican, 658 (31.04%) tweets are Pro-Democratic, and remaining 153 (7.22%) are Neutral tweets.

(3) **MeToo dataset**: We collected a set of tweets related to the #MeToo movement in October 2018. We initially collected 10,000 English tweets containing the hashtag #MeToo using the Twitter Search API [1]. After removing duplicates, we were left with 3,982 distinct tweets. We asked three human annotators to examine the name and bio of the Twitter accounts who posted the tweets. The annotators observed three classes of tweets based on who posted the tweets – tweets posted by male users, tweets posted by female users, and tweets posted by organizations (mainly news media agencies). Also, there were many tweets for which the annotators could not understand the type/gender of the user posting the tweet. For purpose of this study, we decided to focus only on

those tweets for which all the annotators were certain that they were written by men or women. In total, we had 488 such tweets, out of which 213 are written by men and 275 are written by women.

In summary, two of our datasets contain tweets posted by two social groups (men and women) which the other dataset contains three categories of tweets (pro-democratic, pro-republican and neutral tweets, presumably written by users having the corresponding political leanings).

Human-generated summaries for evaluation: The traditional way of evaluating the 'goodness' of a summary is to match it with one or more human-generated summaries (gold standard), and then compute ROUGE scores [41]. ROUGE scores are between $[0, 1]$, where a higher ROUGE score means a better algorithmic summary that has higher levels of 'similarity' with the gold standard summaries. Specifically, the similarity is computed in terms of common unigrams (in case of ROUGE-1) or common bigrams (in case of ROUGE-2) between the algorithmic summary and the human-generated summaries. For creating the gold standard summaries, we asked three human annotators to summarize the datasets. Each annotator is well-versed with the use of social media like Twitter, is fluent in English, and none is an author of this paper. The annotators were asked to generate extractive summaries independently, i.e., without consulting one another. We use these three human-generated summaries for the evaluation of algorithmically-generated summaries, by computing the average ROUGE-1 and ROUGE-2 Recall and F_1 scores [41].

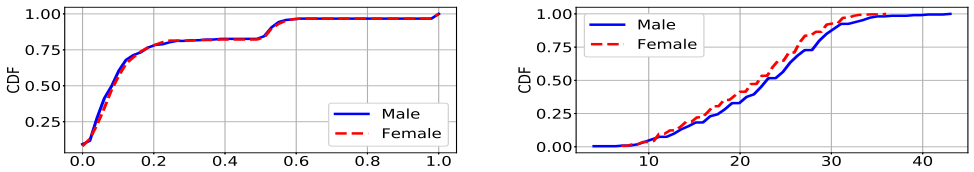
4 WHY DO WE NEED FAIR SUMMARIES?

Traditionally, summarization algorithms have only considered including (in the summary) those textual units (tweets, in our case) whose contents are most 'summary-worthy'. In contrast, in this paper, we argue for giving a fair chance to textual units written by different social groups to appear in the summary. Before making this argument, two questions need to be investigated –
(1) Are the tweets written by different social groups of comparable textual quality? If not, someone may argue for discarding lower quality tweets generated by a specific user group.
(2) Do the tweets written by different social groups actually reflect different opinions? This question is important since, if the opinions of the different groups are not different, then it can be argued that selecting tweets of any group (for inclusion in the summary) is sufficient.
We attempt to answer these two questions in this section.

4.1 Are tweets written by different social groups of comparable quality?

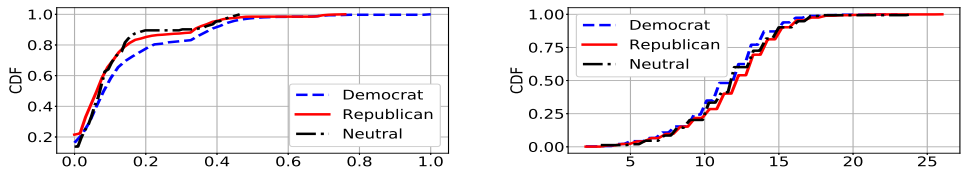
We use three measures for estimating the textual quality of individual tweets. (i) First, the NAVA words (nouns, adjectives, verbs, adverbs) are known to be the most informative words in an English text [49]. Hence we consider the count of NAVA words in a tweet as a measure of its textual quality. We consider two other measures of textual quality that are specific to the application of text summarization – (ii) ROUGE-1 precision and (iii) ROUGE-2 precision scores. Put simply, the ROUGE-1 (ROUGE-2) precision score of a tweet measures what fraction of the unigrams (bigrams) in the tweet appears in the gold standard summaries for the corresponding dataset (as described in Section 3). Thus, these scores specifically measure the utility of selecting a particular tweet for inclusion in the summary.

For a particular dataset, we compare the distributions of the three scores – ROUGE-1 precision score, ROUGE-2 precision score, and count of NAVA words – for the subsets of tweets written by different user groups. For all cases, we found that ROUGE-1 precision scores and ROUGE-2 precision scores show similar trends; hence we report only the ROUGE-2 precision scores. Figure 1(a) and Figure 1(b) respectively compare the distributions of ROUGE-2 precision scores and NAVA word counts among the tweets written by male and female users in the MeToo dataset. We find that the distributions are very close to each other, thus implying that the tweets written by both groups



(a) ROUGE-2 Precision (b) NAVA word count

Fig. 1. **Comparing textual quality of individual tweets of the two user groups in MeToo dataset – distributions of (a) ROUGE-2 Precision scores and (b) Count of NAVA words, of individual tweets.**



(a) ROUGE-2 Precision (b) NAVA word count

Fig. 2. **Comparing textual quality of individual tweets of the three groups in US-Election dataset – distributions of (a) ROUGE-2 Precision scores and (b) Count of NAVA words.**

are of comparable textual quality. Similarly, Figure 2 shows that, in the US-Election dataset, the pro-democratic, pro-republican and neutral tweets are of comparable textual quality. The textual quality of the tweets written by male and female users in the Claritin dataset are also very similar – the mean number of NAVA words are 8.19 and 8.61 respectively for tweets written by male and female users, while the mean ROUGE-2 Precision scores are 0.22 for male and 0.20 for female (detailed results omitted for brevity). All these values show that the textual quality is very similar for the different groups of tweets, across all the three datasets.

4.2 Do tweets written by different user groups reflect different opinion?

To answer this question, we asked our human annotators (those who prepared the gold standard summaries) to observe the tweets written by different user groups in the datasets. For all three datasets, the annotators observed that the tweets posted by different social groups mostly contain very different information/opinion.

For instance, Table 1 shows some sample tweets written by male and female users in the MeToo dataset, along with some of the hashtags that are frequently posted by male and female users (highlighted). We observe that most tweets written by women support the #MeToo movement, and give examples of relevant experiences of themselves or of other women. On the other hand, many of the tweets written by male users point out undesirable side-effects of the movement, and call for gender equality.

Similarly, in the US-Election dataset, the pro-republican tweets criticize Hillary Clinton and/or support the policies of Donald Trump (e.g., '*We must not let #CrookedHillary take her criminal scheme into the Oval Office. #DrainTheSwamp*'), while the pro-democratic tweets have the opposite opinion (e.g. '*Yes America. This is the election where Hillary's cough gets more furious coverage than Trump asking people to shoot her #InterrogateTrump*'). The neutral tweets either give only information (and no opinion), or criticize both Clinton and Trump. For the Claritin dataset as well, there is large difference in opinion among the tweets written by male and female users – the female users

Tweets on #MeToo from male users	Tweets on #MeToo from female users
If a woman shares a **#metoo** without evidence, it's taken to be true coz it's a women's testimony, a man coming out with **#HeToo** story, people would be doubtful, & question the evidences, the intent & will never except the man as victim. **#misandry** must be understood. **#SpeakUpMan**	If a woman is unveiled it gives a man the right 2 demand sexual favors.When it comes 2 sexual harassment in Islamic Republic it is always your fault if U dont wear hijab. Women using camera to expose sexual harassment. **#MyCameraIsMyWeapon** is like **#MeToo** movement in Iran
Instead of arresting this women @CPMumbaiPolice taking common man coz its **#MeToo #MeTooIndia #MeToo4Publicity** This is why **#FeminismIsCancer #feminismIsMisandry #CrimeByWomen**	Whatever happens to you in your life, you always have the choice to rise above your challenges. Choose NOT to be a victim. **#feminism #metoo**
Pain knows no gender. When it hurts, it hurts equally, whether its a man or woman. Why there is discrimination on Gender. Every person deserves dignified treatment and happy life. **#MeToo #MeToo4Publicity**	ONLY 40 charges and thousands of cries for help. Too many are victim to **#UberRape** and their voices aren't being heard. **#TimesUp #Metoo**
When Settlement amount is the motive by falsely charging a man' it's called **#MeToo** Pls tk action on ppl filing **#FakeCases** & bring **#GenderNeutralLaws #MeToo4publicity #MensCommission.**	A long term solution would be the exact opposite of the two suggested here - gender sensitisation, not segregation so that exchange between different genders is normalised instead of being stigmatised further. **#MeToo**

Table 1. **Example tweets containing the hashtags that are most frequently posted by male and female users, in the MeToo dataset. Even though all tweets have high textual quality, the opinions expressed by the two groups of users are quite diverse.**

criticize the drug much more than the male users (details omitted for brevity). Thus, it is clear that tweets posted by different social groups often reflect very different opinions.

4.3 Need for fairness in summarization

The fact that tweets written by different social groups are of very similar quality/merit implies that all groups should have 'equality of opportunity' [59] for their opinions to be reflected in the summary. This fact, coupled with the diversity in opinion of the different groups, calls for a fair representation of the opinions of different groups in the summary. This is similar in spirit to the need for fairness in top crowdsourced recommendations [16] or top search results [8]. Since the tweets that get included in the summary are likely to get much more exposure than the rest of the information (just like how top search and recommendation results get much more exposure [8, 16]), under-representation of any of the social groups in the summary can severely suppress their opinion. These factors advocate the need for fair summaries when data generated by various social groups is being summarized.

5 NOTIONS OF FAIR SUMMARIZATION

Having established the need for fair summarization, we now define two fairness notions that are applicable in the context of summarization. Essentially, when the input data (e.g. tweets) are generated by users belonging to different social groups, we require the summaries to *fairly represent* these groups. Next, we consider two notions for fairness in representation.

5.1 Equal Representation

The notion of equality finds its roots in the field of morality and justice, which advocates for the redress of undeserved inequalities (e.g. inequalities of birth or due to natural endowment) [58]. Formal equality suggests that when two people or two groups of people have equal status in at least one normatively relevant aspect, they must be treated equally [30]. In terms of selection, equal representation requires that the number of representatives from different classes in the society having comparable relevance has to be equal.

In the context of user-generated content, we observed that different sections of the society have different opinion on the same topic, either because of their gender or ideological leaning [4]. However, if we consider the textual quality, i.e. their candidature for inclusion in the summary, then tweets from both the groups are comparable (as discussed in section 4). Thus, the notion of equal representation requires that a summarization algorithm will be fair if different groups generating the input data are represented equally in the output summary. Given the usefulness of summaries in many downstream applications, this notion of fairness ensures equal exposure to the opinions of different socially salient groups.

5.2 Proportional Representation

Often it may not be possible to equally represent different user groups in the summary, especially if the input data contains very different proportions from different groups. Hence, we consider another notion of fairness: *Proportional Representation* (also known as *Statistical Parity* [46]). Proportional representation requires that the representation of different groups in the selected set should be proportional to their distribution in the input data.

In certain scenarios such as hiring for jobs, relaxations of this notion are often used. For instance, the U.S. Equal Employment Opportunity Commission uses a variant of Proportional Representation to determine whether a company's hiring policy is biased against (has any adverse impact on) a demographic group [7]. According to this policy, a particular class c is **under-represented** in the selected set (or **adversely impacted**), if the fraction of selected people belonging to class c is less than 80% of the fraction of selected people from the class having the highest selection rate.

In the context of summarization, Proportional Representation requires that the proportion of content from different user groups in the summary should be same as in the original input. A relaxed notion of proportional fairness is one which would ensure *no adverse impact* in the generated summary. In other words, 'no adverse impact' requires that the fraction of textual units from any class, that is selected for inclusion in the summary, *should not be* less than 80% of the fraction of selected units from the class having the highest selection rate (in the summary). These notions of fairness ensure that the probability of selecting an item is **independent** of which user group generated it.

It should be noted that, we are *not* advocating for any particular notion of fairness to be better in the context of summarization. We also note that different applications may require different types of fairness. Hence, in this work, we propose mechanisms that can accommodate different notions of fairness, including the ones stated above, and produce fair summaries accordingly.

6 DO EXISTING ALGORITHMS PRODUCE FAIR SUMMARIES?

Having discussed the need for fair summarization, we now check whether existing algorithms generate fair summaries.

6.1 Summarization algorithms

We consider a set of well-known extractive summarization algorithms, that select a subset of the textual units for inclusion in the summary. Some of the methods are unsupervised (the traditional methods) and some are recent supervised neural models.

Unsupervised summarization algorithms: We consider six well-known summarization algorithms. These algorithms generally estimate an importance score for each textual unit (sentence / tweet) in the input, and the k textual units having the highest importance scores are selected to generate a summary of length k.

(1) Cluster-rank [28] which clusters the textual units to form a cluster-graph, and uses graph algorithms (e.g., PageRank) to compute the importance of each unit.

(2) DSDR [36] which measures the relationship between the textual units using linear combinations and reconstructions, and generates the summary by minimizing the reconstruction error.

(3) LexRank [25], which creates a graph representation based on similarity of the units, where edges are placed depending on the intra-unit cosine similarity, and then computes the importance of textual units using eigenvector centrality on this graph.

(4) LSA [29], which constructs a terms-by-units matrix, and estimates the importance of the textual units based on Singular Value Decomposition on the matrix.

(5) LUHN [45], which derives a 'significance factor' for each textual unit based on occurrences and placements of frequent words within the unit.

(6) SumBasic [51], which uses frequency-based selection of textual units, and reweights word probabilities to minimize redundancy.

Supervised neural summarization algorithms: With the recent popularity of neural network based models, the state of the art techniques for summarization have shifted to data-driven supervised algorithms [21]. We have considered two recently proposed extractive neural summarization models, proposed in [50]:

(7) SummaRuNNer-RNN, a Recurrent Neural Network based sequence model that provides a binary label to each textual unit: – a label of 1 implies that the textual unit can be part of the summary, while 0 implies otherwise. Each label has an associated confidence score. The summary is generated by picking textual units labeled 1 in decreasing order of their confidence score.

(8) SummaRuNNer-CNN is a variant of the above model where the sentences are fed to a two layer Convolutional Neural Network (CNN) architecture before using GRU-RNN in the third layer. For both the SummaRuNNer models, the authors have made the pretrained models available[1] which are trained on the CNN/Daily Mail news articles corpus[2]. We directly used the pretrained models for the summarization.

6.2 Verifying if the summaries are fair

We applied the summarization algorithms stated above on the datasets described in Section 3, to obtain summaries of length 50 tweets each. Table 2 shows the results of summarizing the Claritin dataset, while Table 3 and Table 4 show the results for the US-Election and MeToo datasets respectively. In all cases, shown are the numbers of tweets of the different classes in the whole dataset (first row), and in the summaries generated by the different summarization algorithms (subsequent rows), and the average ROUGE-1 and ROUGE-2 Recall and F_1 scores of the summaries.

We check whether the generated summaries are fair, according to the fairness notions of equal representation, proportional representation and the principle of *'no adverse impact'* [7] (which were

[1]https://github.com/hpzhao/SummaRuNNer

[2]https://github.com/deepmind/rc-data

Method	Nos. of tweets		ROUGE-1		ROUGE-2	
	Female	Male	Recall	F_1	Recall	F_1
Whole data	2,505 (62%)	1,532 (38%)	NA	NA	NA	NA
ClusterRank	33 (66%)	17 (34%)†*	0.437	0.495	0.161	0.183
DSDR	31 (62%)	19 (38%)*	0.302	0.425	0.144	0.203
LexRank	34 (68%)	16 (32%)$^{\#\dagger\star}$	0.296	0.393	0.114	0.160
LSA	35 (70%)	15 (30%)$^{\#\dagger\star}$	0.515	0.504	0.151	0.147
LUHN	34 (68%)	16 (32%)$^{\#\dagger\star}$	0.380	0.405	0.128	0.136
SumBasic	27 (54%)$^{\#\dagger}$	23 (46%)*	0.314	0.434	0.108	0.149
SummaRNN	33 (66%)	17 (34%)†*	0.342	0.375	0.126	0.147
SummaCNN	30 (60%)†	20 (40%)*	0.377	0.409	0.126	0.146

Table 2. **Results of summarizing the Claritin dataset: Number of tweets posted by the two user groups, in the whole dataset and in summaries of length 50 tweets generated by different algorithms. Also given are ROUGE-1 and ROUGE-2 Recall and F_1 scores of each summary. The symbols ⋆, † and # respectively indicate under-representation of a group according to the fairness notions of equal representation, proportional representation, and 'no adverse impact' [7].**

Method	Nos. of tweets			ROUGE-1		ROUGE-2	
	Pro Rep	Pro Dem	Neutral	Recall	F_1	Recall	F_1
Whole data	1,309 (62%)	658 (31%)	153 (7%)	NA	NA	NA	NA
ClusterRank	32 (64%)	15 (30%)*	3 (6%)*	0.247	0.349	0.061	0.086
DSDR	28 (56%)$^{\#\dagger}$	19 (38%)	3 (6%)$^{\#\star}$	0.215	0.331	0.067	0.104
LexRank	27 (54%)$^{\#\dagger}$	20 (40%)	3 (6%)$^{\#\star}$	0.252	0.367	0.078	0.114
LSA	24 (48%)$^{\#\dagger}$	20 (40%)$^{\#}$	6 (12%)*	0.311	0.404	0.083	0.108
LUHN	34 (68%)	13 (26%)$^{\#\dagger\star}$	3 (6%)$^{\#\star}$	0.281	0.375	0.085	0.113
SumBasic	27 (54%)$^{\#\dagger}$	23 (46%)	0 (0%)$^{\#\dagger\star}$	0.200	0.311	0.051	0.080
SummaRNN	34 (68%)	15 (30%)*	1 (2%)$^{\#\dagger\star}$	0.347	0.436	0.120	0.160
SummaCNN	32 (64%)	17 (34%)	1 (2%)$^{\#\dagger\star}$	0.337	0.423	0.108	0.145

Table 3. **Results of summarizing the US-Election dataset: Number of tweets of the three groups in the whole data and summaries of length 50 tweets generated by different algorithms. The symbols ⋆, † and # denote under-representation of the corresponding group, similar to Table 2.**

Method	Nos. of tweets		ROUGE-1		ROUGE-2	
	Female	Male	Recall	F_1	Recall	F_1
Whole data	275 (56.3%)	213 (43.7%)	NA	NA	NA	NA
ClusterRank	24 (48%)$^{\#\dagger\star}$	26 (52%)	0.550	0.560	0.216	0.223
DSDR	32 (64%)	18 (36%)$^{\#\dagger\star}$	0.233	0.358	0.092	0.141
LexRank	34 (68%)	16 (32%)$^{\#\dagger\star}$	0.285	0.414	0.105	0.153
LSA	20 (40%)$^{\#\dagger\star}$	30 (60%)	0.511	0.534	0.175	0.183
LUHN	22 (44%)$^{\#\dagger\star}$	28 (56%)	0.520	0.522	0.219	0.184
SumBasic	27 (54%)†	23 (46%)*	0.464	0.499	0.216	0.229
SummaRNN	23 (46%)$^{\#\dagger\star}$	27 (54%)	0.622	0.636	0.385	0.394
SummaCNN	23 (46%)$^{\#\dagger\star}$	27 (54%)	0.622	0.636	0.385	0.394

Table 4. **Results of summarizing the MeToo dataset: Number of tweets of the two classes, in the whole dataset and in summaries of length 50 tweets generated by different algorithms. The symbols ⋆, † and # denote under-representation of the corresponding group, similar to Table 2.**

explained in Section 5). We find under-representation of particular groups of users in the summaries generated by many of the algorithms; these cases are marked in Table 2, Table 3 and Table 4 with the symbols ⋆ (where equal representation is violated), † (where proportional representation is violated) and # (cases where there is adverse impact). Especially, the minority groups are under-represented in most of the cases.

We repeated the experiments for summaries of lengths other than 50 as well, such as for $100, 200, \ldots, 500$ (details omitted due to lack of space). We observed several cases where the same algorithm includes very different proportions of tweets of various groups, while generating summaries of different lengths.

Thus, there is no guarantee of fairness in the summaries generated by the existing summarization algorithms – one or more groups are often under-represented in the summaries, even though the quality of the tweets written by different groups are quite similar (as was shown in Section 4).

7 ACHIEVING FAIRNESS IN SUMMARIZATION

Recently, there has been a flurry of research activities focusing on fairness issues in algorithmic decision making systems, with the main emphasis on classification algorithms [24, 67, 69]. Approaches proposed in these works can be broadly categorised into three types [26]: *pre-processing*, *in-processing* and *post-processing*, based on the stage where the fairness intervention is applied. To achieve fairness, pre-processing approaches attempt to change the input data/representation, in-processing approaches change the underlying algorithm itself, and post-processing methods change the outputs of the algorithm before they get used in downstream applications.

Following this line of work, in this paper, we develop three novel fairness-preserving summarization algorithms (adhering to the principles of pre-, in- and post-processing) which select highly relevant textual units in the summary while maintaining fairness in the process. Next, we discuss the key ideas behind the proposed algorithms. Each of the algorithms will be explained in detail in subsequent sections.

(1) Pre-processing: As mentioned above, pre-processing approaches attempt to change the input to the algorithms to make the outcome fair. The idea originated from classification algorithms where the biases in the training data may get translated into the learned model, and hence by making the training data or the input unbiased, the algorithm can be made non-discriminatory. In our context, to ensure fair representation, we propose a pre-processing technique *ClasswiseSumm* (described in Section 9.1), where we first group tweets on the basis of their association to different classes. Then, we propose to summarize each group separately using any state-of-the-art algorithm, and generate $\{l_1, l_2, \ldots\}$ length summaries for different groups, where the lengths $\{l_1, l_2, \ldots\}$ would be determined based on the fairness objective. Finally, these individual summaries would be combined to generate the final fair summary.

(2) In-processing: In-processing methods work by changing the underlying learning algorithms and making them adhere to the fairness objectives (for instance, by putting additional fairness constraints). Our proposed algorithm *FairSumm* (detailed in Section 8) is one such algorithm, where we summarize using a constrained sub-modular optimization, with the fairness criteria applied as matroid constraints to an objective function ensuring goodness of the summary.

(3) Post-processing: The third approach for bringing fairness into algorithmic systems is by modifying the outputs of an algorithm to achieve the desired results for different groups. Intervention at the output stage becomes necessary when the summarization algorithm is already decided, and there is no option to change its working. For example, in our context, if some organization intends to stick to its proprietary summarization algorithm, then post-processing on the generated summaries (or the ranked list of textual units) becomes necessary to produce fair summaries. Hence, we propose *ReFaSumm* (**Re**ranking **Fa**irly the **Summ**arization outputs) where we attempt to fairly re-rank the outputs generated by existing summarization algorithms (detailed in Section 9.2).

8 FAIRSUMM: IN-PROCESSING ALGORITHM FOR FAIR SUMMARIZATION

Our proposed in-processing algorithm, named `FairSumm`, treats summarization as a constrained optimization problem of an objective function. The objective function is designed so that optimizing it is likely to result in a good quality summary, while the fairness requirements are applied as constraints which must be obeyed during the optimization process.

Some notations: Let V denote the set of textual units (e.g., tweets) that is to be summarized. Our goal is to find a subset S ($\subseteq V$) such that $|S| \leq k$, where k (an integer) is the desired length of the summary (specified as an input),

8.1 Formulating summarization as an optimization problem

We need an *objective function* for extractive summarization, optimizing which is likely to lead to a good summary. Following the formulation by Lin *et al.* [43], we consider two important aspects of an extractive text summarization algorithm, viz. *Coverage* and *Diversity reward*, described below.

Coverage: Coverage refers to amount of information covered in the summary S. Clearly, the summary cannot contain the information in all the textual units. We consider the summary S to cover the information contained in a particular textual unit $i \in V$ if either S contains i, or if S contains another textual unit $j \in V$ that is very similar to i. Here we assume a notion of similarity $sim(i, j)$ between two textual units $i \in V$ and $j \in V$, which can be measured in various ways. Thus, the coverage will be measured by a function – say, $\mathcal{L}$ – whose generic form can be

$$\mathcal{L}(S) = \sum_{i \in S, j \in V} sim(i, j) \tag{1}$$

Thus, $\mathcal{L}(S)$ measures the overall similarity of the textual units included in the summary S with all the textual units in the input collection V.

Diversity reward: The purpose of this aspect is to avoid redundancy and reward diverse information in the summary. Usually, it is seen that the input set of textual units can be partitioned into groups, where each group contains textual units that are very similar to each other. A popular way of ensuring diversity in a summary is to partition the input set into such groups, and then select a representative element from each group [23].

Specifically, let us consider that the set V of textual units is partitioned into K groups. Let $P_1, P_2, \ldots, P_K$ comprise a partition of V. That is, $\cup_i P_i = V$ (V is formed by the *union* of all P_i) and $P_i \cap P_j = \emptyset$ (P_i, P_j have no element in common) for all $i \neq j$. For instance, the partitioning $P_1, P_2, \ldots, P_K$ can be achieved by clustering the set V using any clustering algorithm (e.g., K-means), based on the similarity of items as measured by $sim(i, j)$.

Then, to reduce redundancy and increase diversity in the summary, including textual units from different partitions needs to be rewarded. Let the associated function for diversity reward be denoted as $\mathcal{R}$. A generic formulation of $\mathcal{R}$ is

$$\mathcal{R}(S) = \sum_{i=1}^{K} \sqrt{\sum_{j \in P_i \cap S} r_j} \tag{2}$$

where r_j is a suitable function that estimates the importance of adding the textual unit $j \in V$ to the summary. The function r_j is called a 'singleton reward function' since it estimates the reward of adding the singleton element $j \in V$ to the summary S. One possible way to define this function is by measuring the average similarity of j to the other textual units in V. Mathematically,

$$r_j = \frac{1}{N} \sum_{i \in V} sim(i, j) \tag{3}$$

Justifying the functional forms of Coverage and Diversity Reward: We now explain the significance of the functional form of $\mathcal{L}(S)$ in Equation 1 and $\mathcal{R}(S)$ in Equation 2. We give only an intuitive explanation here; more mathematical details are given in the Supplementary Information accompanying the paper[3].

The functions $\mathcal{L}(S)$ and $\mathcal{R}(S)$ are designed to be 'monotonic non-decreasing submodular' functions (or 'monotone submodular' functions), since such functions are easier to optimize. A monotonic non-decreasing function is one that does not decrease (usually increases) as the set over which the function is employed grows. A submodular function has the property of *diminishing returns* which intuitively means that as the set (over which the function is employed) grows, the increment of the function decreases.

$\mathcal{L}$ is monotone submodular. $\mathcal{L}$ is monotonic since coverage increases by the addition of a new sentence in the summary. At the same time, $\mathcal{L}$ is submodular since the increase in $\mathcal{L}$ would be more when a sentence is added to a shorter summary, than when it is added to a longer summary.

Also $\mathcal{R}$ is a monotone submodular function. The diversity of a summary increases considerably only for the initial growth of the set (when new, 'novel' elements are added to the summary) and stabilizes later on, and thus prevents the incorporation of similar elements (redundancy) in the summary. $\mathcal{R}(S)$ rewards diversity since there is more benefit in selecting a textual unit from a partition (cluster) that does not yet have any of its elements included in the summary. As soon as any one element from a cluster P_i is included in the summary, the other elements in P_i start having diminishing gains, due to the square root function in Equation 2.

Combining Coverage and Diversity reward: While constructing a summary, both coverage and diversity are important. Only maximizing coverage may lead to lack of diversity in the resulting summary and vice versa. So, we define our objective function for summarization as follows:

$$\mathcal{F} = \lambda_1 \mathcal{L} + \lambda_2 \mathcal{R} \tag{4}$$

where $\lambda_1, \lambda_1 \geq 0$ are the weights given to coverage and diversity respectively.

Our proposed fairness-preserving summarization algorithm will maximize $\mathcal{F}$ in keeping with some fairness constraints. Note that $\mathcal{F}$ is monotone submodular since it is a non-negative linear combination of two monotone submodular functions $\mathcal{L}$ and $\mathcal{R}$. We have chosen $\mathcal{F}$ such that it is monotone submodular, since there exist standard algorithms to efficiently optimize such functions (as explained later in the section).

8.2 Proposed fair summarization scheme

Our proposed scheme is based on the concept of *matriods* that are typically used to generalize the notion of liner independence in matrices [55]. Specifically, we utilize a special type of matroids, called *partition matroids*. We give here a brief, intuitive description of our method. More details can be found in the Supplementary Information.

Brief background on matroids and related topics: In mathematical terms, a matroid is a pair $\mathcal{M} = (\mathcal{Z}, \mathcal{I})$, defined over a finite set $\mathcal{Z}$ (called the ground set) and a family of sets $\mathcal{I}$ (called the independent sets), that satisfies the three properties:

(1) $\emptyset$ (empty set) $\in \mathcal{I}$.
(2) If $Y \in \mathcal{I}$ and $X \subseteq Y$, then $X \in \mathcal{I}$.
(3) If $X \in \mathcal{I}$, $Y \in \mathcal{I}$ and $|Y| > |X|$, then there exists $e \in Y \setminus X$ such that $X \cup \{e\} \in \mathcal{I}$.

Condition (1) simply means that $\mathcal{I}$ can contain the empty set, i.e., the empty set is *independent*. Condition (2) means that every subset of an independent set is also independent. Condition (3)

[3]http://cse.iitkgp.ac.in/~saptarshi/docs/DashEtAl-CSCW2019-fair-summarization-SuppleInfo.pdf

means that if X is independent and there exists a larger independent set Y, X can be extended to a larger independent set by adding an element in Y but *not in* X.[4]

Partition matroids refer to a special type of matroids where the ground set $\mathcal{Z}$ is partitioned into s disjoint subsets $\mathcal{Z}_1, \mathcal{Z}_2, ..., \mathcal{Z}_s$ for some s, and $\mathcal{I} = \{S \mid S \subseteq \mathcal{Z}$ and $|S \cap \mathcal{Z}_i| \leq c_i$, for all $i = 1, 2, ..., s\}$ for some given parameters $c_1, c_2, ..., c_s$. Thus, S is a subset of Z that contains at least c_i items from the partition $\mathcal{Z}_i$ (for all i), and $\mathcal{I}$ is the family of all such subsets.

Consider that we have a set of control variables z_j (e.g., 'gender', 'political leaning'). Each item in $\mathcal{Z}$ has a particular value for each z_j. Also consider that z_j takes t_j distinct values, e.g., the control variable 'gender' takes the two distinct values 'male' and 'female', while the control variable 'political leaning' takes the values 'Democrat', 'Republican' and 'Neutral'.

For each control variable z_j, we can partition $\mathcal{Z}$ into t_j disjoint subsets $\mathcal{Z}_{j1}, \mathcal{Z}_{j2}, ..., \mathcal{Z}_{jt_j}$, each corresponding to a particular value of this control variable. We now define a partition matriod $\mathcal{M}_j = (\mathcal{Z}, \mathcal{I}_j)$ such that

$$\mathcal{I}_j = \{S \mid S \subseteq \mathcal{Z} \text{ and } |S \cap Z_{ji}| \leq c_j, \text{for all } i = 1, 2, \ldots, t_j\}$$

for some given parameters $c_1, c_2, ..., c_{t_j}$.

Now, for a given *submodular* objective function f, a submodular optimization under the partition matriod constraints with P control variables can be designed as follows:

$$Maximize_{S \subseteq \mathcal{Z}} \quad f(S) \qquad (5)$$
$$\text{subject to } S \in \bigcap_{j=1}^{P} \mathcal{I}.$$

A prior work by Du *et al.* [22] has established that this submodular optimization problem under the matroid constraints can be solved efficiently with provable guarantees (see [22] for details).

Formulating the fair summarization problem: In the context of the fair summarization problem, the ground set is V (= $\mathcal{Z}$), the set of all textual units (sentences/tweets) which we look to summarize. The control variables are analogous to the sensitive attributes with respect to which fairness is to be ensured, such as 'gender' or 'political leaning'. In this work, we consider only one sensitive attribute for a particular dataset (the gender of a user for the Claritin and MeToo datasets, and political leaning for the US-Election dataset). Let the corresponding control variable be z, and let z take t distinct values (e.g., $t = 2$ for the Claritin and MeToo datasets, and $t = 3$ for the US-Election dataset). Note that, the formulation can be extended to multiple sensitive attributes (control variables) as well.

Each textual unit in V is associated with a class, i.e., a particular value of the control variable z (e.g., is posted either by a male or a female). Let $Z_1, Z_2, ..., Z_t$ ($Z_i \subseteq V$, for all i) be disjoint subsets of the textual units from the t classes, each associated with a distinct value of z. We now define a partition matroid $\mathcal{M} = (V, \mathcal{I})$ in which V is partitioned into disjoint subsets $Z_1, Z_2, ..., Z_t$ and

$$\mathcal{I} = \{S \mid S \subseteq V \text{ and } |S \cap Z_i| \leq c_i, i = 1, 2, ..., t\}$$

for some given parameters $c_1, c_2, ..., c_t$. In other words, I will contain all the sets S containing at most c_i textual units from Z_i, $i = 1, 2, ..., t$.

Now we add the fairness constraints. Outside the purview of the matroid constraints, we maintain the restriction that c_i's are chosen such that
(1) $\sum_{i=1}^{t} c_i = k$ (the desired length of the summary S), and
(2) a desired fairness criterion is maintained in S. For instance, if equal representation of all classes in the summary is desired, then $c_i = \frac{k}{t}$ for all i.
We now express our fairness-constrained summarization problem as follows:

$$Maximize_{S \subseteq V} \quad \mathcal{F}(S) \qquad (6)$$

[4]For details, refer to http://www-math.mit.edu/~goemans/18433S09/matroid-notes.pdf

Algorithm 1 : FairSumm (in-processing approach for fair summarization)

1: Set $d = max_{z \in V} \mathcal{F}(\{z\})$.
2: Set $w_t = \frac{d}{(1+\delta)^t}$ for $t = 0, \ldots, l$ where $l = argmin_i \ [w_i \leq \frac{\delta d}{N}]$, and $w_{l+1} = 0$.
3: Set $G = \emptyset$
4: **for** $t = 0, 1, \ldots, l, l+1$ **do**
5: **for** each $z \in V$ and $G \cup \{z\} \in I$ **do**
6: **if** $\mathcal{F}(G \cup \{z\})$ - $\mathcal{F}(G) \geq w_t$ **then**
7: Set $G \leftarrow G \cup \{z\}$
8: **end if**
9: **end for**
10: **end for**
11: Output G as the summary

subject to $S \in \mathcal{I}$.

where the objective function $\mathcal{F}(S)$ is as stated in Equation 4. Given that $\mathcal{F}$ is a submodular function (as explained earlier in this section), the algorithm proposed by Du et al. [22] is suitable to solve this constrained submodular optimization problem.

An example: Let us illustrate the formulation of the fair summarization problem with an example. Assume that we are applying the equal representation fairness notion over the MeToo dataset, and we want a summary of length $k = 50$ tweets. Then, the control variable z corresponds to the sensitive attribute 'gender' which takes $t = 2$ values ('male' and 'female') for this particular dataset. The set of tweets V will be partitioned into two disjoint subsets Z_1 and Z_2 which will comprise the tweets posted by male and female users respectively. To enforce equal representation fairness constraint, we will set the parameters $c_1 = 25$ and $c_2 = 25$ (since we want equal number of tweets from Z_1 and Z_2 in the summary). Thus, I contains all the possible sets S that contain at most 25 tweets written by male users and 25 tweets written by female users. Each such S is a valid summary (that satisfies the fairness constraints). Solving the optimization problem in Equation 6 will give us that summary S for which $\mathcal{F}(S)$ will be maximum, i.e. for which coverage and diversity reward will be the highest.

Algorithm for fair summarization: Algorithm 1 presents the algorithm to solve this constrained submodular optimization problem, based on the algorithm developed by Du et al. [22]. The G output by Algorithm 1 is the solution of Equation 6. We now briefly describe the steps of Algorithm 1.

In Step 1, the maximum value of the objective function $\mathcal{F}$ that can be achieved for a text unit z ($\in V$) is calculated and stored in d. The purpose of this step is to compute the maximum value of $\mathcal{F}$ for a single text unit z and set a selection threshold (to be described shortly) with respect to this value. This step will help in the subsequent selection of textual units for the creation of the summary to be stored in G. w_t (defined in Step 2) is such a threshold at the t^{th} time step. w_t is updated (decreased by division with a factor $1 + \delta$) for $t = 0, 1, \ldots, l$. l is the minimum value of i for which $w_i \leq \frac{\delta d}{N}$ holds (see Du et al. [22] for details) and w_{l+1} is set to zero. In Step 3, G (the set that will contain the summary) is initialized as an empty set. Note that G is supposed to be an independent set according to the definition of matroid given earlier in this section. By condition (1) in the definition of matroids (stated earlier in this section), an empty set is independent. Step (4) iterates through the different values of t. Step (5) tests, for each z (text element) $\in V$, if G remains an independent set by the inclusion of z. Only those z's are chosen in this step whose inclusion expands G (already an independent set) to another independent set. Step (6) selects a z (permitted by Step (5)) for inclusion in G if $\mathcal{F}(G \cup \{z\})$ - $\mathcal{F}(G) \geq w_t$. This z is added to G in Step (7). That is, z

is added to G if the increment of $\mathcal{F}$ by the addition of z is not less than the threshold w_t. For $t = 0$, $w_t = d$, that is, the maximum value of $\mathcal{F}$ for any z ($\in V$). This means, the z which maximizes $\mathcal{F}$ is added to G. Note that, there can be multiple z's for which $\mathcal{F}$ is maximized. In that case, the tie is broken arbitrarily. The remaining z's may or may not be added to G based on the threshold value.

Another important point to note is that, our chosen $\mathcal{F}$ (see equation (4)) is designed to maximize both coverage and diversity. So, even if multiple z's satisfy Step (5), they may not be added to G in Step (7) if they contain redundant information. The value of w_t is relaxed for the subsequent values of t to allow text elements z producing relatively lower increments of $\mathcal{F}$ to be considered for possible inclusion in G. $w_{l+1} = 0$ indicates that for the final value of t, at least one text unit z which does not decrement $\mathcal{F}$ is added to G. This ensures that the coverage of the summary produced is not compromised while preserving diversity. This process (Steps (5) to (7)) is repeated for $t = 0, 1, \ldots, l, l + 1$ resulting in the final output G.

The reason for the efficiency of Algorithm 1 is the fact that this algorithm does *not* perform exhaustive evaluation of all the possible submodular functions evolving in the intermediate steps of the algorithm. The reduction in the number of steps in the algorithm is achieved mainly by decreasing w_t geometrically by a factor of $1 + \delta$. In addition, multiple elements z can be added to G for a single threshold which also expedites the culmination of the algorithm.

9 PRE AND POST-PROCESSING MECHANISMS FOR FAIR SUMMARIZATION

In this section, we discuss our proposed pre-processing and post-processing summarization algorithms to produce fair summaries.

9.1 ClasswiseSumm: Pre-processing algorithm for fair summarization

We now describe a simple pre-procesing algorithm for fair summarization. Suppose that the textual units in the input belong to t classes $Z_1, Z_2, \ldots, Z_t$, and to conform to a desired fairness notion, the summary should have c_i units from class Z_i, $i = 1, 2, \ldots, t$ (using the same notations as in Section 8). The simplest way to generate a fair summary is to *separately summarize the textual units belonging to each class* Z_i, to produce a summary of length c_i, and finally to combine all the t summaries to obtain the final summary of length k. We refer to this method as the **ClasswiseSumm** method. Specifically, in this work, we use our proposed algorithm FairSumm, without any fairness constraints, to summarize each class separately. However, any other summarization algorithm can be used to summarize each class separately.

9.2 ReFaSumm: Post-processing algorithm for fair summarization

In this section, we discuss our proposed post-processing mechanism for generating fair summaries, which can be used along with any existing summarization algorithm. Many summarization algorithms (including the ones stated in Section 6) generate an importance score of each textual unit in the input. The textual units are then ranked in decreasing order of this importance score, and the top-ranked k units are selected to form the summary. Hence, if the ranked list of the textual units can be made fair (according to some desired fairness notion), then selecting the top k from this fair ranked list can generate a fair summary. We refer to this algorithm as ReFaSumm (**Re**-ranking **Fa**irly the **Summ**arization output).

Fairness in ranking systems is an important problem that has been addressed recently by some works [8, 68]. We adopt the fair-ranking methodology developed by Zehlike *et al.* [68] to generate fair summaries. The fair-ranking scheme in [68] considers a **two-class** setting, with a 'majority class' and a 'minority class' for which fairness has to be ensured adhering to a *ranked group fairness criterion* . Their proposed ranking algorithm (named *FA*IR* [68]) ensures that the proportion of the candidates/items from the minority class in a ranked list never falls below a certain specified

threshold. Specifically, two fairness criteria are ensured – *selection utility* which means every selected item is more qualified than those not selected, and *ordering utility* which means for every pair of selected candidates, either the more qualified is ranked above or the difference in their qualifications is small [68].

We propose to use the algorithm in [68] for fair extractive text summarization as follows. Note that this scheme is only applicable to cases where there are two groups (e.g., the Claritin and MeToo datasets). We consider that group to be the majority class which has the higher number of textual units (tweets) in the input data, while the group having lesser textual units in the input is considered the minority class.

Input and Parameter settings: The algorithm takes as input a set of textual units (to be summarized); the other input parameters (k, q_i, g_i and p) taken by the algorithm in [68] are set as follows.
• *Qualification (q_i) of a candidate:* In our summarization setting, this is the goodness value of a textual unit in the data to be summarized. We set this value to the importance score computed by some standard summarization algorithm (e.g., the ones discussed in Section 6) that ranks the text units by their importance scores.
• *Expected size (k) of the ranking:* The expected number of textual units in the summary (k).
• *Indicator variable (g_i) indicating if the candidate is protected:* We consider that group to be the minority class which has the lesser number of textual units in the input data. All tweets posted by the minority group are marked as 'protected'.
• *Minimum proportion (p) of protected candidates:* We will set this value in the open interval]0, 1[(0 and 1 excluded) so that a particular notion of fairness is ensured in the summary. For instance, if we want equal representation of both classes in the summary, we will set $p = 0.5$.
• *Adjusted significance level (α_c):* We regulate this parameter in the open interval]0, 1[.

Working of the algorithm: Two priority queues P_0 (for the textual units of the majority class) and P_1 (for the textual units of the minority class), each with capacity k, are set to empty. P_0 and P_1 are initialized by the goodness values (q_i) of the majority and minority textual units respectively. Then a ranked group fairness table is created which calculates the minimum number of minority textual units at each rank, given the parameter setting. If this table determines that a textual unit from the minority class needs to added to the summary (being generated), the algorithm adds the best element from P_1 to the summary S; otherwise it adds the overall best textual unit (from $P_0 \cup P_1$) to S. Thus a fair summary (S) of desired length k is generated, adhering to a particular notion of fairness (decided by the parameter setting).

Note that since the *FA*IR* algorithm provides fair ranking for two classes only [68], we look to apply this algorithm for summarization of data containing tweets from exactly two social groups (i.e., the Claritin and MeToo datasets only). It is an interesting future work to design a fair ranking algorithm for more than two classes, and then to use the algorithm for summarizing data from more than two social groups.

10 EXPERIMENTS AND EVALUATION

We now experiment with different methodologies of generating fair summaries, over the three datasets described in Section 3.

Pre-processing the datasets: We performed standard pre-processing on the datasets including stopword removal, and stemming using Porter Stemmer. All summarization algorithms were executed on these pre-processed datasets.

10.1 Parameter settings of algorithms

The following parameter settings are used.

• For all datasets, we generate all summaries of $k = 50$ tweets.
• The proposed algorithm (FairSumm) uses a similarity function $sim(i, j)$ to measure the similarity between two tweets i and j. We experimented with the following two similarity functions:
TFIDFsim – we compute TF-IDF scores for each word (unigram) in a dataset, and hence obtain a TF-IDF vector for each textual unit. The similarity $sim(i, j)$ is computed as the cosine similarity between the TF-IDF vectors of i and j.
Embedsim – we obtain an embedding (a vector of dimension 300) for each distinct word in a dataset, either by training Word2vec [48] on the dataset, or by considering pre-trained GloVe embeddings [56]. We obtain an embedding for a tweet by taking the mean embedding of all words contained in the tweet. $sim(i, j)$ is computed as the cosine similarity between the embeddings of tweets i and j.
We found that the performance of the FairSumm algorithm is very similar for both the similarity measures. Hence, we report results for the *TFIDFsim* similarity measure.
• For our post-processing algorithm (the one based on fair ranking), the value of the parameter α_c needs to be decided (see Section 9.2). We try different values of α_c in the interval $]0, 1[$ using grid search, and finally use $\alpha_c = 0.5$ since this value obtained the best ROUGE scores on the Claritin and MeToo datasets.

10.2 Baselines

To our knowledge, there is no existing fair text summarization algorithm. Hence we consider the standard text summmarization algorithms stated in Section 6 as baselines. Additionally, we have used our proposed FairSumm algorithm (described in Section 8) without considering any fairness constraints as a separate baseline. We call this summarization algorithm *DiCoSumm*, a summarization algorithm that is optimized for both diversity and coverage, but not considering any fairness constraints. Next, we compare the performance of the different fair summarization algorithms with all the baselines.

10.3 Results and Insights

We now describe the results of applying various fair summarization algorithms over the three datasets. Some sample summaries obtained by using various algorithms are given in the Supplementary Information.

To evaluate the quality of summaries, we compute ROUGE-1 and ROUGE-2 Recall and F_1 scores by matching the algorithmically generated summaries with the gold standard summaries (described in Section 3). Table 5 reports the results of summarizing the Claritin dataset. We compute summaries without any fairness constraint, and considering the two fairness notions of *equal representation* and *proportional representation* (explained in Section 5). In each case, we state the number of tweets in the summary from the two user groups, and the ROUGE scores of the summary. Similarly, Table 6 and Table 7 report the results for the MeToo dataset and the US-Election dataset respectively.

The FairSumm algorithm (in-processing algorithm) and the ClasswiseSumm algorithm (pre-processing algorithm) are executed over all three datasets. For the two-class Claritin and MeToo datasets, we also apply our post-processing methodology (stated in Section 9.2) where a fair ranking scheme is used for fair summarization (results in Table 5 and Table 6). Specifically, we use our post-processing methodology over the existing summarization algorithms described in Section 6 such as ClusterRank, LexRank, SummaRNN, SummaCNN, etc. The resulting fair summarization algorithms are denoted as Fair-ClusRank, Fair-LexRank, Fair-SummaRNN, Fair-SummaCNN, and so on. Note that, for generating a fixed length summary, the neural models use only the textual units labeled with 1, ranked as per their confidence scores. Hence, in Fair-SummaRNN and Fair-SummaCNN methods, we have considered the ranked list of only those textual units that are labeled with 1.

Approach	Algorithm	Nos. of tweets		ROUGE-1		ROUGE-2	
		Female	Male	Recall	F_1	Recall	F_1
	Whole data	2,505 (62%)	1,532 (38%)				
Baselines (which do not consider fairness)							
	ClusterRank	33	17	0.437	0.495	0.161	0.183
	DSDR	31	19	0.302	0.425	0.144	0.203
	LexRank	34	16	0.296	0.393	0.114	0.160
	LSA	35	15	0.515	0.504	0.151	0.147
	LUHN	34	16	0.380	0.405	0.128	0.136
	SumBasic	27	23	0.314	0.434	0.108	0.149
	SummaRNN	33	17	0.342	0.375	0.126	0.147
	SummaCNN	30	20	0.377	0.409	0.126	0.146
	DiCoSumm	37	13	0.548	0.545	0.172	0.171
Fairness: Equal representation							
In-processing	FairSumm	25	25	**0.560**	**0.552**	**0.188**	**0.185**
Pre-processing	ClasswiseSumm	25	25	0.545	0.538	0.172	0.170
Post-processing (ReFaSumm used with existing summarization algorithms)	Fair-ClusRank	25	25	0.433	0.481	0.135	0.162
	Fair-DSDR	25	25	0.285	0.400	0.139	0.206
	Fair-LexRank	25	25	0.290	0.370	0.110	0.153
	Fair-LSA	25	25	0.513	0.493	0.114	0.109
	Fair-LUHN	25	25	0.415	0.429	0.114	0.118
	Fair-SumBasic	25	25	0.314	0.436	0.111	0.154
	Fair-SummaRNN	25	25	0.356	0.410	0.126	0.154
	Fair-SummaCNN	25	25	0.356	0.410	0.126	0.154
Fairness: Proportional representation							
In-processing	FairSumm	31	19	**0.572**	**0.568**	**0.206**	**0.202**
Pre-processing	ClasswiseSumm	31	19	0.550	0.541	0.180	0.173
Post-processing (ReFaSumm)	Fair-ClusRank	31	19	0.439	0.483	0.133	0.159
	Fair-DSDR	31	19	0.302	0.425	0.145	0.204
	Fair-LexRank	31	19	0.312	0.406	0.115	0.160
	Fair-LSA	31	19	0.502	0.487	0.118	0.115
	Fair-LUHN	31	19	0.426	0.435	0.119	0.121
	Fair-SumBasic	31	19	0.318	0.435	0.116	0.159
	Fair-SummaRNN	31	19	0.340	0.394	0.120	0.147
	Fair-SummaCNN	31	19	0.340	0.394	0.120	0.147

Table 5. **Summarizing the Claritin dataset: Number of tweets written by the two user groups, in the whole dataset and the summaries of length 50 tweets generated by different algorithms. Also given are the ROUGE-1 and ROUGE-2 Recall and F_1 scores of each summary.**

Insights from the results: We make the following observations from the results shown in Table 5, Table 6 and Table 7.

• **In-processing and post-processing methods perform better than pre-processing:** Across all datasets, the in-processing FairSumm algorithm achieves higher ROUGE scores than ClasswiseSumm, considering the same fairness notion. Note that in the ClasswiseSumm approach, the same FairSumm algorithm is used on each class separately. Hence, the pre-processing approach of separately summarizing each class leads to relatively poor summaries, as compared to the in-processing FairSumm methodology. This difference in performance is probably because, if similar tweets / opinions are posted by different social groups, ClasswiseSumm can include multiple similar posts in the summary, thereby leading to redundancy in the summary. On the other hand, FairSumm optimizes coverage and diversity across all textual units taken together, thereby avoiding redundancy in the summary.

Approach	Algorithm	Nos. of tweets		ROUGE-1		ROUGE-2	
		Female	Male	Recall	F_1	Recall	F_1
	Whole data	275 (56.3%)	213 (43.7%)				
Baselines (which do not consider fairness)							
	ClusterRank	24	26	0.550	0.560	0.216	0.223
	DSDR	32	18	0.233	0.358	0.092	0.141
	LexRank	34	16	0.285	0.414	0.105	0.153
	LSA	20	30	0.511	0.534	0.175	0.183
	LUHN	22	28	0.520	0.522	0.219	0.184
	SumBasic	27	23	0.464	0.499	0.216	0.229
	SummaRNN	23	27	0.622	0.636	0.385	0.394
	SummaCNN	23	27	0.622	0.636	0.385	0.394
	DiCoSumm	30	20	0.563	0.569	0.229	0.249
Fairness: Equal representation							
In-processing	FairSumm	25	25	0.616	0.613	0.285	0.296
Pre-processing	ClasswiseSumm	25	25	0.587	0.569	0.189	0.196
Post-processing (ReFaSumm)	Fair-ClusRank	25	25	0.499	0.532	0.186	0.198
	Fair-DSDR	25	25	0.558	0.574	0.157	0.162
	Fair-LexRank	25	25	0.511	0.564	0.209	0.230
	Fair-LSA	25	25	0.556	0.541	0.196	0.191
	Fair-LUHN	25	25	0.527	0.537	0.207	0.211
	Fair-SumBasic	25	25	0.541	0.567	0.180	0.189
	Fair-SummaRNN	25	25	0.623	0.629	0.371	0.375
	Fair-SummaCNN	25	25	**0.623**	**0.629**	**0.371**	**0.375**
Fairness: Proportional representation							
In-processing	FairSumm	28	22	**0.631**	**0.648**	0.311	0.338
Pre-processing	ClasswiseSumm	28	22	0.605	0.622	0.279	0.298
Post-processing (ReFaSumm)	Fair-ClusRank	28	22	0.499	0.528	0.174	0.184
	Fair-DSDR	28	22	0.565	0.577	0.168	0.172
	Fair-LexRank	28	22	0.518	0.564	0.210	0.228
	Fair-LSA	28	22	0.560	0.544	0.197	0.191
	Fair-LUHN	28	22	0.533	0.541	0.213	0.216
	Fair-SumBasic	28	22	0.546	0.569	0.190	0.198
	Fair-SummaRNN	28	22	0.622	0.636	0.385	0.394
	Fair-SummaCNN	28	22	0.621	0.636	**0.385**	**0.394**

Table 6. **Summarizing the MeToo dataset: Number of tweets written by the two user groups, in the whole dataset and the summaries of length 50 tweets generated by different algorithms. Also given are the ROUGE-1 and ROUGE-2 Recall and F_1 scores of each summary.**

The in-processing FairSumm algorithm and some of the post-processing approaches achieve comparable performances. For instance, while FairSumm performs decidedly better than all other algorithms for the Claritin dataset, the post-processing approaches Fair-SummaCNN and Fair-SummaRNN perform better in most cases over the MeToo dataset.

Note that the performances of the pre-processing and post-processing algorithms depend on that of the original summarization algorithm that is used. As such, the pre-processing and post-processing algorithms can be useful in situations where an existing summarization algorithm is preferred, e.g., when a firm has a proprietary summarization algorithm.

• **Proposed algorithms are generalizable to different fairness notions:** Table 5, Table 6 and Table 7 demonstrate that the proposed algorithms are generalizable to various fairness notions. We demonstrate summaries conforming to equal representation and proportional representation for all three datasets. Additionally, Table 7 shows different summaries that can be generated using

Approach	Algorithm	Nos. of tweets			ROUGE-1		ROUGE-2	
		Pro-Rep	Pro-Dem	Neutral	Recall	F_1	Recall	F_1
	Whole data	1,309 (62%)	658 (31%)	153 (7%)				
Baselines (which do not consider fairness)								
	ClusterRank	32	15	3	0.247	0.349	0.061	0.086
	DSDR	28	19	3	0.215	0.331	0.067	0.104
	LexRank	27	20	3	0.252	0.367	0.078	0.114
	LSA	24	20	6	0.311	0.404	0.083	0.108
	LUHN	34	13	3	0.281	0.375	0.085	0.113
	SumBasic	27	23	0	0.200	0.311	0.051	0.080
	SummaRNN	34	15	1	0.347	0.436	0.120	0.160
	SummaCNN	32	17	1	0.337	0.423	0.108	0.145
	DiCoSumm	34	12	4	0.359	0.460	0.074	0.091
Fairness: Equal representation								
In-processing	FairSumm	17	17	16	**0.368**	**0.467**	**0.078**	**0.096**
Pre-processing	ClasswiseSumm	16	16	18	0.363	0.467	0.071	0.088
Fairness: Proportional representation								
In-processing	FairSumm	31	15	4	**0.376**	**0.490**	**0.094**	**0.116**
Pre-processing	ClasswiseSumm	30	15	5	0.367	0.454	0.081	0.100
Fairness: No Adverse Impact								
In-processing	FairSumm	29	17	4	0.371	0.484	0.086	0.102
	FairSumm	30	16	4	0.372	0.489	0.087	0.109
	FairSumm	31	15	4	**0.376**	**0.490**	**0.094**	**0.116**
	FairSumm	31	16	3	0.371	0.477	0.085	0.096
	FairSumm	32	15	3	0.371	0.473	0.085	0.093

Table 7. **Summarizing the US-Election dataset: Number of tweets of the three classes, in the whole dataset and the summaries of length 50 tweets generated by different algorithms. Also given are the ROUGE-1 and ROUGE-2 Recall and F_1 scores of each summary.**

FairSumm considering the 'no adverse impact' fairness notion (such rows are omitted from other tables for brevity).

In general, summaries conforming to proportional representation achieve higher ROUGE scores than summaries conforming to other fairness notions, probably because the human assessors intuitively attempt to represent different opinions (coming from different social groups) in a similar proportion in the gold standard summaries as what occurs in the input data (even though they were not told anything about ensuring fairness while writing the gold standard summaries).

• **Ensuring fairness does not lead to much degradation in summary quality:** For all three datasets, we observe that FairSumm with fairness constraints always achieves higher ROUGE scores than DiCoSumm (without fairness constraints). Also, we can compare the performances of the existing summarization algorithms (e.g., DSDR, LexRank, SummaRNN) without any fairness constraint, and after their outputs are made fair using the methodology in Section 9.2. We find that the performances in the two scenarios are comparable to each other. In fact, for a few cases, the ROUGE scores marginally improve after the summaries generated by an algorithm are made fair, over those of the original summary generated by the same algorithm. Thus, *making summaries fair does not lead to much degradation in summary quality* (as measured by ROUGE scores).

Overall, the results signify that, the proposed fair summarization algorithms can not only ensure various fairness notions in the summaries, but also can generate summaries that achieve comparable (or better) ROUGE scores than many well-known summarization algorithms (which often do not generate fair summaries, as demonstrated in Section 6).

Error Rate	Nos. of tweets		ROUGE-1		ROUGE-2	
	Female	Male	Recall	F_1	Recall	F_1
Whole data	275 (56.3%)	213 (43.7%)				
Fairness: Equal representation						
0 %	25	25	**0.616**	**0.613**	**0.285**	**0.296**
10 %	27	23	0.612	0.604	0.285	0.286
20 %	29	21	0.596	0.582	0.282	0.287
30 %	29	21	0.591	0.590	0.282	0.287
Fairness: Proportional representation						
0 %	28	22	**0.631**	**0.648**	**0.311**	**0.338**
10 %	29	21	0.624	0.633	0.304	0.317
20 %	30	20	0.615	0.610	0.297	0.302
30 %	31	19	0.614	0.610	0.293	0.300

Table 8. **Effect of degraded information of gender in MeToo dataset: Number of tweets written by the two user groups, in the whole dataset and the summaries of length 50 tweets generated by FairSumm algorithm. Also given are the ROUGE-1 and ROUGE-2 Recall and F_1 scores of each summary. Each result is averaged over 100 random experiments for each of the error rates.**

10.4 Effects of degraded information of demographic details

Our proposed algorithms assume the availability of class information of the textual units under consideration e.g., gender information for the tweets in Claritin and MeToo datasets, and ideological leaning for tweets in the US-Election dataset. In cases where such class information is not available, we need to resort to inference mechanisms. For instance, there are multiple methodologies to infer demographic details of people from their writing styles with a high level of accuracy [54, 63, 64]. Moreover, in social media, additional information can be utilized to infer demographic details of their users, such as user names and profile pictures [15]. Similarly, sentiment analysis can be deployed to infer the opinions. However, such inferences may not be absolutely perfect, and the accuracy of these inference methodologies will eventually decide how well the predicted labels replicate the true class labels. In this section, we check how the inaccuracy in the inference mechanism may impact the performance of the proposed FairSumm algorithm.

Experimental Setup: We performed these experiments for all the three datasets, and obtained qualitatively similar results. Hence, for brevity, we are reporting the experimental results only on the MeToo dataset. We assume the existence of a classifier which can infer the gender information of the authors of textual posts (required for our FairSumm algorithm), with certain level of error, for the MeToo dataset. To simulate the effect of a classifier with $x\%$ error rate, we change the class labels (i.e., consider the inferred label to be 'male' if the true label was 'female', and vice versa) of *randomly selected* $x\%$ of the tweets in the dataset. Now this degraded information of gender labels is given as input to the FairSumm algorithm to create the fair summaries. We experiment with error rates $x = 10\%, 20\%, 30\%$. For every error rate, we repeat the experiment 100 times, and then report the average results over all experiments. Note that, for checking the fairness property (i.e., the proportion of tweets from various groups in the summaries), we use the true labels (and not the inferred labels).

Observations: Table 8 reports the results obtained over the MeToo dataset, with increasing amount of noise/error in the demographic labels. As expected, the error/noise in the demographic inference has some effects on the fairness property that the summaries are meant to satisfy. With increasing error rate in the prediction of the class labels, the summaries deviate further from the desired fairness criterion. However, it is interesting to note that even with degraded demographic information, the

summaries generated by FairSumm have better fairness property than the summaries generated by many of the baseline algorithms. Also it is evident from Table 8 that the degradation in the availability of demographic details does *not* affect the quality of the summaries much (as measured by ROUGE scores). In fact, even with the degraded gender information, the FairSumm algorithm outperforms many of the existing summarization algorithms in terms of ROUGE scores.

Hence, we can conclude that, though the accuracy of the inference methodology is an important factor when the actual class information is not present (especially in order to achieve the fairness goals), the quality of summaries produced by FairSumm is generally robust to such noise in the inferred labels.

11 CONCLUDING DISCUSSION

To our knowledge, this work is the first attempt to consider fairness in textual summarization. Through experiments on several user-generated microblog datasets, we show that existing algorithms often produce summaries that are not fair, even though the text written by different social groups are of comparable quality. Note that, we do *not* claim the existing algorithms to be intentionally biased towards/against any social group. Since these algorithms attempt to optimize only one metric (e.g., textual quality of the summary), the unfairness comes as a side-effect. We further propose algorithms to generate high-quality summaries that conform to various standard notions of fairness (implementations available at https://github.com/ad93/FairSumm.git). These algorithms will help in addressing the concern that using a (inadvertently) 'biased' summarization algorithm can reduce the visibility of the voice/opinion of certain social groups in the summary. Moreover, downstream applications that use the summaries (e.g., for opinion classification and rating inference [44]) would benefit from a fair summary.

Limitations and future directions: There are potential limitations of the fair summarization algorithms presented in this paper. All three algorithms need as input the class (e.g., socially salient group) information to which each textual unit belongs. Where such class information is not readily available, we need to infer these information, which may impact the fairness objectives to some extent, as discussed in Section 10.4. A major limitation of ReFaSumm comes from the fair ranking algorithm being applied in the post-processing phase. The FA*IR framework is designed for scenarios where only two socially salient groups are defined (e.g., male and female). Hence our post-processing algorithm is presently applicable only in such cases. We plan to extend ReFaSumm to more than two classes in future work.

We also note that, in certain special cases, the fair summarization algorithms developed in this work may lead to degradation in the summary quality. For example, let us assume a scenario where we are summarizing tweets posted by two equally-sized groups of users, e.g., group A and group B, and the fairness objective is to achieve *Equal Representation*, i.e., both groups should have the same number of tweets in the summary. Now, if the variability of opinions within the groups are different – e.g., everyone from group A has the same opinion on an issue, while people in group B have many varied opinions on the same issue – then the proposed method will not generate a good summary because there will be redundant tweets posted by users of group A, while some of the diverse opinions posted by users of group B may not be included in the summary. Hence, if the distribution of opinions is very different from the distribution of people belonging to different social groups, then the summaries may not be of good quality. This situation leads us to an interesting question of whether to look for *fair representation across demographic groups*, or for *fair representation across the different opinions* – a question that we would like to investigate in future work.

Going beyond summarization, deciding the social grouping is often a normative question in many socio-technical applications, which requires decisions at the policy level. In some applications,

legal doctrines may suggest what should be the social grouping (which often emerges from long deliberations and historical contexts); whereas, in other cases, the corresponding online platforms (such as social media sites like Facebook or Twitter) may have their own guidelines to decide groups they want to be considerate about. Similar to most of the recent algorithms trying to incorporate group fairness (including fair classification algorithms [24, 67, 69]), the algorithms proposed in this paper also consider the grouping to be given apriori. However, questions on what constitute the right grouping are important, and should be more widely discussed in the research community.

Finally, looking at a higher level, fairness-preserving information filtering algorithms like the ones proposed in this paper are of significant societal importance. Today social media sites are the gateway of information for a large number of people worldwide; and the algorithms (search, recommendation, sampling, summarization etc.) deployed in these sites act as the gatekeepers. If these algorithms lack the sense of embedded ethics or civic responsibilities, they may not be fully suitable to curate information for the heterogeneous society. Thus, incorporating fairness in algorithms is the need of the hour. As discussed in Section 2, recent research works are taking correct steps in that direction. Likewise, we believe that our work will open up multiple interesting research problems on fair summarization, such as extending the concept of fairness to abstractive summaries, or estimating user preferences for fair summaries in various applications, and will be an important addition to the emerging literature on fairness in algorithmic decision making.

REFERENCES

[1] 2019. Twitter Search API. https://developer.twitter.com/en/docs/tweets/search/api-reference/get-search-tweets.html. (2019).

[2] Rasim M Alguliev, Ramiz M Aliguliyev, Makrufa S Hajirahimova, and Chingiz A Mehdiyev. 2011. MCMR: Maximum coverage and minimum redundant text summarization model. *Expert Systems with Applications* 38, 12 (2011).

[3] Mehdi Allahyari, Seyed Amin Pouriyeh, Mehdi Assefi, Saeid Safaei, Elizabeth D. Trippe, Juan B. Gutierrez, and Krys Kochut. 2017. Text Summarization Techniques: A Brief Survey. *CoRR* (2017). http://arxiv.org/abs/1707.02268

[4] Mahmoudreza Babaei, Juhi Kulshrestha, Abhijnan Chakraborty, Fabrício Benevenuto, Krishna P Gummadi, and Adrian Weller. 2018. Purple feed: Identifying high consensus news posts on social media. In *Proc. AAAI/ACM AIES*.

[5] Ashwinkumar Badanidiyuru, Baharan Mirzasoleiman, Amin Karbasi, and Andreas Krause. 2014. Streaming submodular maximization: Massive data summarization on the fly. In *Proc. ACM KDD*.

[6] Ricardo Baeza-Yates. 2018. Bias on the web. *Commun. ACM* 61, 6 (2018).

[7] Dan Biddle. 2006. *Adverse Impact and Test Validation: A Practitioner's Guide to Valid and Defensible Employment Testing*. Routledge.

[8] Asia J. Biega, Krishna P. Gummadi, and Gerhard Weikum. 2018. Equity of Attention: Amortizing Individual Fairness in Rankings. In *Proc. ACM SIGIR*.

[9] Francesco Bonchi, Sara Hajian, Bud Mishra, and Daniele Ramazzotti. 2017. Exposing the probabilistic causal structure of discrimination. *International Journal of Data Science and Analytics* 3, 1 (2017).

[10] Giuseppe Carenini, Raymond T Ng, and Xiaodong Zhou. 2007. Summarizing email conversations with clue words. In *Proc. ACM WWW*.

[11] Asli Celikyilmaz and Dilek Hakkani-Tur. 2010. A hybrid hierarchical model for multi-document summarization. In *Proc. ACL*.

[12] Elisa L Celis, Amit Deshpande, Tarun Kathuria, and Nisheeth K Vishnoi. 2016. How to be Fair and Diverse? *CoRR* (2016). http://arxiv.org/abs/1610.07183

[13] Elisa L. Celis, Vijay Keswani, Damian Straszak, Amit Deshpande, Tarun Kathuria, and Nisheeth K. Vishnoi. 2018. Fair and Diverse DPP-based Data Summarization. *CoRR* (2018). http://arxiv.org/abs/1802.04023

[14] Abhijnan Chakraborty, Saptarshi Ghosh, Niloy Ganguly, and Krishna P Gummadi. 2016. Dissemination biases of social media channels: On the topical coverage of socially shared news. In *Proc. AAAI ICWSM*.

[15] Abhijnan Chakraborty, Johnnatan Messias, Fabricio Benevenuto, Saptarshi Ghosh, Niloy Ganguly, and Krishna P Gummadi. 2017. Who makes trends? understanding demographic biases in crowdsourced recommendations.

[16] Abhijnan Chakraborty, Gourab K Patro, Niloy Ganguly, Krishna P Gummadi, and Patrick Loiseau. 2019. Equality of Voice: Towards Fair Representation in Crowdsourced Top-K Recommendations. In *Proc. ACM FAT**.

[17] Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, and Sergei Vassilvitskii. 2017. Fair Clustering Through Fairlets. In *Proc. NeurIPS*.

[18] claritin-dataset 2013. Discovering Drug Side Effects with Crowdsourcing. (2013). https://www.crowdflower.com/discovering-drug-side-effects-with-crowdsourcing/.
[19] Brian W Collins. 2007. Tackling unconscious bias in hiring practices: The plight of the Rooney rule. *NYU Law Review* 82 (2007).
[20] K. Darwish, W. Magdy, and Zanouda T. 2017. Trump vs. Hillary: What Went Viral During the 2016 US Presidential Election. In *Proc. SocInfo*.
[21] Yue Dong. 2018. A Survey on Neural Network-Based Summarization Methods. *CoRR* (2018). http://arxiv.org/abs/1804.04589
[22] Nan Du, Yingyu Liang, Maria-Florina Balcan, and Le Song. 2013. Continuous-Time Influence Maximization for Multiple Items. *CoRR* abs/1312.2164 (2013). http://arxiv.org/abs/1312.2164
[23] Soumi Dutta, Vibhash Chandra, Kanav Mehra, Asit Kr. Das, Tanmoy Chakraborty, and Saptarshi Ghosh. 2018. Ensemble Algorithms for Microblog Summarization. *IEEE Intelligent Systems* 33, 3 (2018), 4–14.
[24] Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. 2012. Fairness through awareness. In *Proc. ACM ITCS*.
[25] Günes Erkan and Dragomir R. Radev. 2004. LexRank: Graph-based Lexical Centrality As Salience in Text Summarization. *Journal of Artificial Intelligence Res.* 22, 1 (2004).
[26] Sorelle A Friedler, Carlos Scheidegger, Suresh Venkatasubramanian, Sonam Choudhary, Evan P Hamilton, and Derek Roth. 2019. A comparative study of fairness-enhancing interventions in machine learning. In *Proc. ACM FAT**.
[27] Batya Friedman and Helen Nissenbaum. 1996. Bias in computer systems. *ACM Transactions on Information Systems* 14, 3 (1996), 330–347.
[28] Nikhil Garg and others. 2009. Clusterrank: a graph based method for meeting summarization. In *Proc. INTERSPEECH*.
[29] Yihong Gong and Xin Liu. 2001. Generic Text Summarization Using Relevance Measure and Latent Semantic Analysis. In *Proc. ACM SIGIR*.
[30] Stefan Gosepath. 2011. Equality. In *The Stanford Encyclopedia of Philosophy*, Edward N. Zalta (Ed.).
[31] Vishal Gupta and Gurpreet Singh Lehal. 2010. A Survey of Text Summarization Extractive Techniques. *IEEE Journal of Emerging Technologies in Web Intelligence* 2, 3 (2010).
[32] Ben Hachey. 2009. Multi-document summarisation using generic relation extraction. In *Proc. EMNLP*.
[33] Sara Hajian, Francesco Bonchi, and Carlos Castillo. 2016. Algorithmic bias: From discrimination discovery to fairness-aware data mining. In *Proc. ACM KDD*.
[34] Sara Hajian, Josep Domingo-Ferrer, and Oriol Farràs. 2014. Generalization-based privacy preservation and discrimination prevention in data publishing and mining. *Data Mining and Knowledge Discovery* 28 (2014).
[35] Moritz Hardt, Eric Price, Nati Srebro, and others. 2016. Equality of opportunity in supervised learning. In *Proc. NeurIPS*.
[36] Zhanying He and others. 2012. Document Summarization Based on Data Reconstruction. In *Proc. AAAI*.
[37] Joy Kim and Andres Monroy-Hernandez. 2016. Storia: Summarizing social media content based on narrative theory using crowdsourcing. In *Proc. ACM CSCW*.
[38] Jon Kleinberg, Jens Ludwig, Sendhil Mullainathan, and Cass R Sunstein. 2018. Discrimination in the Age of Algorithms. *Journal of Legal Analysis* 10 (2018).
[39] Jon Kleinberg and Sendhil Mullainathan. 2019. *Simplicity Creates Inequity: Implications for Fairness, Stereotypes, and Interpretability*. Technical Report. National Bureau of Economic Research.
[40] Jon Kleinberg and Manish Raghavan. 2018. Selection problems in the presence of implicit bias. *CoRR* (2018). http://arxiv.org/abs/1801.03533
[41] Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Proc. Workshop on Text Summarization Branches Out, ACL*.
[42] Chin-Yew Lin and Eduard Hovy. 2002. From single to multi-document summarization. In *Proc. ACL*.
[43] Hui Lin and Jeff Bilmes. 2011. A Class of Submodular Functions for Document Summarization. In *Proc. ACL (HLT '11)*.
[44] Elena Lloret, Horacio Saggion, and Manuel Palomar. 2010. Experiments on summary-based opinion classification. In *Proc. NAACL HLT*.
[45] H. P. Luhn. 1958. The Automatic Creation of Literature Abstracts. *IBM J. Res. Dev.* 2, 2 (1958).
[46] Binh Thanh Luong, Salvatore Ruggieri, and Franco Turini. 2011. k-NN as an implementation of situation testing for discrimination discovery and prevention. In *Proc. ACM KDD*.
[47] Walid Magdy and Tamer Elsayed. 2016. Unsupervised adaptive microblog filtering for broad dynamic topics. *Information Processing & Management* 52, 4 (2016).
[48] T. Mikolov, W.T. Yih, and G. Zweig. 2013. Linguistic Regularities in Continuous Space Word Representations. In *Proc. NAACL HLT*.
[49] George A. Miller. WordNet: A Lexical Database for English. *Communications of the ACM* (????).
[50] Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. SummaRuNNer: A Recurrent Neural Network Based Sequence Model for Extractive Summarization of Documents. In *Proc. AAAI*.

[51] Ani Nenkova and Lucy Vanderwende. 2005. *The impact of frequency on summarization.* Technical Report. Microsoft Research.
[52] Jeffrey Nichols, Jalal Mahmud, and Clemens Drews. 2012. Summarizing sporting events using twitter. In *Proc. ACM IUI.*
[53] Karen O'Connor, Pranoti Pimpalkhute, Azadeh Nikfarjam, Rachel Ginn, Karen L Smith, and Graciela Gonzalez. Pharmacovigilance on twitter? Mining tweets for adverse drug reactions. In *AMIA Annual Symposium proceedings.*
[54] Alexandra Olteanu, Ingmar Weber, and Daniel Gatica-Perez. 2016. Characterizing the demographics behind the # blacklivesmatter movement. In *Proc. 2016 AAAI Spring Symposium Series.*
[55] James G. Oxley. 2006. *Matroid Theory (Oxford Graduate Texts in Mathematics).* Oxford University Press, Inc.
[56] Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global Vectors for Word Representation. In *Proc. EMNLP.*
[57] Dragomir R. Radev, Eduard Hovy, and Kathleen McKeown. 2002. Introduction to the Special Issue on Summarization. *Comput. Linguist.* 28, 4 (2002).
[58] John Rawls. 2009. *A theory of justice.* Harvard University Press.
[59] John E Roemer. 2009. *Equality of opportunity.* Harvard University Press.
[60] Koustav Rudra, Subham Ghosh, Niloy Ganguly, Pawan Goyal, and Saptarshi Ghosh. 2015. Extracting situational information from microblogs during disaster events: a classification-summarization approach. In *Proc. ACM CIKM.*
[61] Gerard Salton. 1989. Automatic text processing: The transformation, analysis, and retrieval of. *Addison-Wesley* (1989).
[62] Anurag Shandilya, Kripabandhu Ghosh, and Saptarshi Ghosh. 2018. Fairness of Extractive Text Summarization. In *Companion Proceedings of The Web Conference (WWW).* 97–98.
[63] Shafiza Mohd Shariff, Mark Sanderson, and Xiuzhen Zhang. 2016. Correlation analysis of reader's demographics and tweet credibility perception. In *Proc. ECIR.*
[64] Luke Sloan, Jeffrey Morgan, William Housley, Matthew Williams, Adam Edwards, Pete Burnap, and Omer Rana. 2013. Knowing the Tweeters: Deriving sociologically relevant demographics from Twitter. *Sociological research online* (2013).
[65] Tom Sühr, Asia J Biega, Meike Zehlike, Krishna P Gummadi, and Abhijnan Chakraborty. 2019. Two-Sided Fairness for Repeated Matchings in Two-Sided Markets: A Case Study of a Ride-Hailing Platform. In *Proc. ACM KDD.*
[66] Kam-Fai Wong, Mingli Wu, and Wenjie Li. 2008. Extractive summarization using supervised and semi-supervised learning. In *Proc. COLING.*
[67] Muhammad Bilal Zafar, Isabel Valera, Manuel Gomez Rogriguez, and Krishna P Gummadi. 2017. Fairness Constraints: Mechanisms for Fair Classification. In *Proc. AISTAS.*
[68] Meike Zehlike, Francesco Bonchi, Carlos Castillo, Sara Hajian, Mohamed Megahed, and Ricardo Baeza-Yates. 2017. FA*IR: A Fair Top-k Ranking Algorithm. In *Proc. ACM CIKM.*
[69] Rich Zemel, Yu Wu, Kevin Swersky, Toni Pitassi, and Cynthia Dwork. 2013. Learning fair representations. In *Proc. ICML.*
[70] Amy X Zhang and Justin Cranshaw. 2018. Making sense of group chat through collaborative tagging and summarization. *Proc. PACM-HCI* 2, CSCW (2018).
[71] Amy X Zhang, Lea Verou, and David Karger. 2017. Wikum: Bridging discussion forums and wikis using recursive summarization. In *Proc. ACM CSCW.*
[72] Junlin Zhang, Le Sun, and Quan Zhou. 2005. A cue-based hub-authority approach for multi-document text summarization. In *Proc. IEEE NLP–KE.*

Received April 2019; revised June 2019; accepted August 2019