Abstract

With the increasing importance of Natural Language Processing (NLP) tools, their implications on the propagation of societal biases become more and more relevant. In this context, the analysis of political bias in manually written and automatically generated text is a relatively understudied field. Political bias refers to the preference or prejudice towards one political ideology over another. To increase the discourse in this subject area, we analyze contemporary studies on detecting and mitigating political bias in this literature review. We further discuss the benefits and potential drawbacks of the considered methods and look at the ethical considerations involved with political bias in NLP, before we give suggestions for future studies.

1 Introduction

With the rising integration of Natural Language Processing (NLP) models in everyday applications, their effects on the propagation of societal biases become increasingly relevant. In this context, not only political bias detection but also mitigation approaches are needed to expose and alleviate such biases.

The definition of political bias across studies is inconsistent. According to Chen et al. (2020), politically unbiased means to "report on an event without taking a political position, characterization, or terminology" while Gangula et al. (2019) define it as not "selectively publishing articles to specifically choosing to highlight some events, parties and leaders". In the review at hand, we base our political bias definition on that of Chen et al. (2020). Political objectivity in our context hence means that an event is being reported without taking a political stance and without adapting an ideology-specific terminology, i.e. to be politically unbiased, as well as fair with regards to the report of original facts rather than opinionated statements (Chen et al., 2020; Ad Fontes Media, 2021). For example, a phrase like 'death tax' would be considered politically biased towards the conservative party in the United States where the term is used to describe a tax that is imposed on property that gets transferred to another person after the owner’s death. On the other hand, liberals, who are in favor of this concept, call it 'estate tax' (Graetz, 2016).

Looking at bias in a machine-learning context, previous studies found that it can be exhibited in multiple components in NLP systems such as pre-trained word embeddings or training data (Zhao et al., 2018; Bolukbasi et al., 2016; Caliskan et al., 2017). In this paper, we are specifically interested in which detection and mitigation approaches exist to deal with such biased sub-components and to prevent the amplification of political bias through text.

The topic is relevant because objective reporting is necessary for an unbiased societal discourse on potentially controversial topics that in turn can shape the political agenda as well as corresponding initiatives on a national and international level (Dardis et al., 2008). While NLP models can be used to identify political bias in human-generated text, they can also be the source of said bias in generative language models. Especially given the large amount of data online, automatic detection and mitigation methods are necessary since the manual identification of political bias becomes increasingly infeasible.

In the paper at hand, we elaborate on the occurrence of political bias, corresponding ethical considerations as well as the necessity for future research in the field in Section 2. Due to the different nature of the detection and mitigation
task in NLP, we decided to split our corresponding review into two separate sections: In Section 3, we analyze approaches to detect political bias before we examine the current state of political bias mitigation in NLP in Section 4. An overview of the considered studies is given in Table 1. We conclude our work with an overview of future research directions in Section 5.

We make two contributions to the current research:

- To the best of our knowledge, we put together the first review of political bias in NLP which builds a basis for future discussions in the field.
- We critically discuss current detection as well as debiasing methods to identify optimization potential and future research directions.

2 Background

Recently, ethical implications of bias in NLP have been the focus of research efforts (Sheng et al., 2020; Bordia and Bowman, 2019). Politically biased texts can be both human- and machine-made. In both cases, the consumers of such texts can be influenced in their decisions and perceptions of the world and hence need to be aware of potential ideology-specific tendencies exhibited by such texts. In addition, consumers need to be able to access unbiased, fair reports about their topic of interest.

Any form of bias can be categorized into two different categories: allocation or representation bias (Crawford, 2017). The former means that certain groups are being preferred in the allocation of resources. In an NLP context, this occurs when models perform better on the majority data. On the other hand, representation bias occurs when considered subgroups (e.g., a specific political ideology) are associated with specific concepts in parameters or embeddings. Both the application and the representation bias can deepen political misconceptions and hence have implications for the national political agenda and respective initiatives. Hence, the increasing use of NLP models poses the risk of propagating and amplifying damaging stereotypes in society.

The most prominent example of an area where political bias can occur is the media. At the article level, published texts might implicitly convey the author’s or the news outlet’s political ideology, i.e., exhibit a right or left bias. In an extreme form of said ideology-specific tendencies, articles can be classified as propaganda (Rashkin et al., 2017; Da San Martino et al., 2019). Any form of media could be biased so that people are not aware of it. For example, word choices and the selective or misrepresentative reporting of events can influence the reader’s perception. A relevant ethical consideration in this context is whether, and if so, in what way, politically biased reporting should be exposed, a) to allow media organizations to stay credible and b) to give people the control over which content they consume and which texts influence their opinions. In this context, the political bias of a medium is further essential to detect so-called fake news (Horne et al., 2018), or to fact-check a claim (Nguyen et al., 2018) and hence to ensure that the reader is either informed about the reliability of the respective source or that the bias is mitigated in the first place.

Political bias is also relevant in the virtual space: In online communities such as social networking sites, complex profiling of users that include psychological characteristics, demographics and meta-data has occurred. Such profiles were subsequently used to micro-target users with politically biased content to gain some form of political advantage (Lazer et al., 2018; Vosoughi et al., 2018). Another aspect of the online realm impacted by political bias is hate speech detection. Hate speech has become an increasing issue in online communities, and detection methods for this phenomenon were developed. However, said models can be impacted by undesired political bias in the training data, which can negatively impact the performance of hate speech classifiers (Wich et al., 2020). This unfavorable effect, in turn, can lead to issues regarding the freedom of speech or to hampering the social discourse if articles are falsely identified as politically-biased hate speech. On the other hand, false identification as non-hate-speech could negatively impact the attacked people, so both misclassifications need to be addressed.

Finally, large-scale language models have re-
<table>
<thead>
<tr>
<th>Study</th>
<th>Purpose</th>
<th>Method</th>
<th>Data Source(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iyyer et al. (2014)</td>
<td>Detection</td>
<td>RNN</td>
<td>Convote &amp; subset of the Ideological Books Corpus</td>
</tr>
<tr>
<td>Chen et al. (2017)</td>
<td>Detection</td>
<td>Opinion-aware Knowledge Graph</td>
<td>Convote &amp; subset of the Ideological Books Corpus &amp; a collection of political Tweets</td>
</tr>
<tr>
<td>Chen et al. (2020)</td>
<td>Detection</td>
<td>RNN &amp; Reverse Feature Analysis</td>
<td>allsides.com &amp; adfontesmedia.com &amp; a collection of manually labeled articles</td>
</tr>
<tr>
<td>Baly et al. (2020)</td>
<td>Detection</td>
<td>Multi-Task Ordinal Regression</td>
<td>allsides.com</td>
</tr>
<tr>
<td>Liu et al. (2021)</td>
<td>Mitigation</td>
<td>Reinforced Calibration</td>
<td>Media Cloud API &amp; survey data from the Pew Research Center</td>
</tr>
</tbody>
</table>

Table 1: Overview of the studies considered along with their purpose, the respective employed method as well as the data source(s) used.

Recently been the focus of research efforts to advance human-like text generation (Zhang et al., 2020; Peng et al., 2020). Other applications of such models are machine translations (Zhu et al., 2020). Given that these language models have been trained on sizeable unsupervised text corpora – for example, GPT-2 (Radford et al., 2019) was trained on 8 million web pages –, they can potentially inherit the (political) bias that was present in the training data and propagate it in the subsequently generated text. This propagation can lead to the amplification of political bias through such models in society, and hence a potentially unethical influence on public opinion.

Our paper is the first review of methods to detect and mitigate political bias in NLP. We provide a basis for future discussions and suggest research directions to advance the current state of the field.

### 3 Political Bias Detection

As seen in the last section, political bias is a phenomenon that can affect people in their opinion formation. Due to the increasing volume of distributed text online, people are more exposed to politically biased work. In addition, the rate of information dissemination in the online realm is much faster, and manual detection of political bias is not feasible in most cases. For this reason, methods to support an automatic bias detection process have been explored in NLP. They are the focus of this chapter. However, to date, there is no standardized data set for politically biased text, and hence the considered papers in our review all rely on individually constructed corpora, limiting the direct comparability of these studies.

#### Recursive Neural Networks

Iyyer et al. (2014) created a balanced data set by subsampling to account for label imbalances. They used a filtered subset of texts from two sources: the U.S. Congressional floor debate (Convote) data set (Thomas et al., 2006) and a manually labeled subset of the Ideological Books Corpus (IBC) (Gross et al., 2013). The former includes transcripts of debates from the U.S. Congress in 2005 labeled with the speaker’s parties (Democrat, Republican, or Independent), which was taken as a proxy for the text’s political bias. The modified IBC data set included texts written by authors with well-known political leanings. Iyyer et al. (2014) subsequently hired crowdsourcers to obtain annotations on a 3-point scale (left, neutral, right) for these texts on the sentence and phrase level. While they only included sentences on which at
least two labelers agreed, this approach introduces
an uncertainty since the crowd workers were not
specialized in political science and hence might
have either propagated their views in the labeling
process or misjudged the presence or direction of
political bias in the data. Post labeling, the authors
trained and tested their recursive neural network
(RNN) architectures on the two different data sets
and found better results for the one with shorter
sentences and more training data. They suggested
that that is likely the case because a) more training
data brings significant improvements for RNN and
b) information is lost at every propagation step, i.e.,
the meaning of shorter sentences is captured easier
than that of longer ones. Their best performing
RNN reached an accuracy of 70.2 %.

Chen et al. (2020), constructed a binary-labeled
corpus at the document-level from allsides.com and
adfontesmedia.com, two platforms that provide
assessments of news articles’ topics, political bias,
and unfairness. Compared to the work by Iyyer
et al. (2014), the authors hence only evaluated
whether political bias is present, but not whether
the text shows a left or right tendency. On the
one hand, this makes the assessment of bias more
reliable since it is easier to determine whether
something is politically biased than the task of
addingly identifying the direction of the bias. At
the same time, however, this causes information
about the political tendency of the text to be
omitted. The authors subsequently approached the
exposure of imbalanced news coverage with an
RNN. Their choice was motivated by the fact that
such networks can capture syntactic and semantic
composition when provided with textual input that
keeps the word order. This capturing is possible
by considering the hierarchical nature of language:
Each word in a sentence is represented as a vector,
and so are higher-level linguistic constructs like
phrases and sentences with the exact vector dimen-
sions as the words they are built on. That way,
the underlying vector representations are trained
to retain the meaning of a sentence (Iyyer et al.,
2014). For example, if a vector represents a liberal
linguistic construct, i.e., a phrase or a sentence, it
should significantly differ from the corresponding
vector representation of a right-wing sentence.
This property of RNN is especially relevant when
identifying more advanced social constructs like
political bias, which are only identifiable at higher
levels of sentence structures rather than at the word
level.

To avoid the learning of media-outlet-specific
features, Chen et al. (2020) removed portal-
identifying information from the text in their
study. The authors achieved an accuracy of 75.42
% for political non-objectivity detection. When
considering individual results, it can be noted
that the prediction of objective articles in this
research tended to be more accurate than the
prediction of non-objective articles, presumably
due to the uneven distribution of biased and
non-biased articles in the training data. Compared
to the previous study by Iyyer et al. (2014), Chen
et al. (2020) did not create a balanced data set to
account for label imbalances, which might be an
explanation for this outcome.

An advantage of RNNs, in general, is that semantic
information of close-by words, but also of con-
structs that are further apart, are detected. However,
this mechanic only works with sufficient training
data, as was suggested by the finding of Iyyer et al.
(2014) that better results were obtained with the
more extensive data set. For example, the construct
’should not be used as an instrument to achieve char-
table or social ends’ got misinterpreted by their
network as non-biased instead of being liberally
oriented because formulations with ‘should not’
did not appear often enough in the training data
for the RNN to pick up on it. Another issue that
needs to be taken into account is that the semantic
information that the network captures depends on
the text’s overall context. Sarcasm and idioms will
most likely not be correctly detected by the RNN
architectures in the two studies considered.

Opinion-aware Knowledge Graph

A different approach was taken by Chen et al. (2017), who created an opinion-aware knowledge graph. Specifically, they used a background knowl-
edge graph (Bizer et al., 2009) containing enti-
ties and semantic relations and infused it with
ideology-specific training data to estimate opinions
expressed towards entities in the graph as senti-
tment distributions over two ideological categories
(conservative vs. liberal). In the next step, the
opinion distributions were propagated based on the
semantic relations between the entities in the graph.
The final opinion-aware knowledge graph was then

...
used to detect the political ideology of the test data set by matching the test entities with the entities in the constructed graph and inferring the respective political orientation from these entities. The authors’ graph was built on three different data sets: In addition to the Convote data set (Thomas et al., 2006) and the IBC (Iyyer et al., 2014) that were also used by Iyyer et al. (2014) (see Section 3 – Recursive Neural Networks), they added Tweets annotated for political bias. For that, they took a list by Bakshy et al. (2015) that contained media outlets with their respective ideological leanings. They subsequently found the corresponding Twitter accounts of these organizations and labeled Tweets from these accounts with the ideology of the respective source.

Such a knowledge graph has the advantage that factual and subjective information can be used for a joint inference based on texts and knowledge bases to detect the political bias of a sentence or document. This is supported by the results that Chen et al. (2017) achieved: The accuracies for their best-performing RNN and support vector machines (SVM) on the data were 70% and 76% respectively, while their knowledge graph achieved an accuracy of 81%.

ELMo Sentence Representation Convolutional Neural Network

Compared to the previously discussed approaches, Jiang et al. (2019) introduced an Embeddings from Language Model (ELMo) sentence representation convolutional neural network (CNN) to identify left- or right-wing hyperpartisanship. They first calculated sentence-level embeddings as the mean of ELMo word embeddings to represent documents as sequences of these sentence-level embeddings, which were subsequently used in a CNN to predict the political orientation. As part of the SemEval-2019 Task 4: Hyperpartisan News Detection competition (Kiesel et al., 2019), the authors were given two different data sets: Firstly, one that encompassed 750k articles that were classified by the political bias of the respective news source they were collected from. To obtain the source’s bias, the organizers of the competition cross-checked two public media bias lists from BuzzFeed and Media Bias Fact Check (Kiesel et al., 2019). In addition, the second data set they provided included 645 manually labeled articles. For these articles, the bias was rated on a 5-point Likert scale by three annotators (Vincent and Mestre, 2018).

Notably in this study is that Jiang et al. (2019) found that their best-performing model was only trained on the manually labeled articles while including the by-publisher data set worsened the accuracy on the test set. The authors achieved an accuracy of 82 % on the held-out test data in the competition (only using the manually labeled set) vs. an accuracy of 64 % for the model trained on the articles classified by publisher. This result indicates that the bias of a news outlet isn’t necessarily propagated to the articles by the respective publishers – which amplifies the need for a data set labeled on the article level.

LIWC and Reverse Feature Analysis

Based on their previously described RNN model, Chen et al. (2020) further performed a reverse feature analysis to investigate how political bias is revealed on the word- and sentence-level as well as in the overall article structure. In each iteration, they removed text parts and re-calculated the bias associated with the text. The estimated bias of the removed segment was derived by subtracting the estimated bias of the new text from the estimated bias of the old text. This approach can be viewed as an attention-based model that outputs weights indicating feature importance (Bahdanau et al., 2014).

On a word level, Chen et al. (2020) correlated the most biased sentences with Linguistic Inquiry and Word Count (LIWC) categories (Pennebaker et al., 2001). They found that especially the classes ‘negative emotions’, ‘focus present’ and ‘percept’ were negatively correlated with political objectivity. This result means that authors of politically biased articles tended to use opinionated and feelings-related words such as ‘angry’ and ‘disappoint’. Chen et al. (2020) also found that unfair articles, i.e., those that only report selected facts, tended to include more words from the category ‘focus present’, for example, ‘admit’ and ‘determine’.

The investigation of higher-level linguistic structures, on the other hand, yielded that the bias strength in the first and second quarters of articles tended to be comparable for objective and non-objective articles. This outcome can be explained by the fact that most articles start with
a high-level summary, followed by background
information (Pöttker and Starck, 2003; Chen et al.,
2020), which shows a tendency to be written in a
neutral tone. The biased nature of articles typically
shows in later parts of the text, especially in the
last quarter. Chen et al. (2020) further found that
it was easiest to detect unfair articles, i.e., those
in which selected facts were reported in favor of
a party. According to the authors, this could be
the case because the word usage in such articles
tended to be more emotional, both with regards to
positive and negative feelings. This finding made
it easier to recognize the underlying bias.

An advantage of the unsupervised reverse feature
analysis proposed by the authors is that the unit to
be analyzed does not have to be defined before
the training of the model, as it is the case with
most other attention models (Chen et al., 2020). In
addition, the knowledge of how biased articles are
structured can make the detection of political bias
in these texts more efficient since future models
could be trained to focus on the last one or two
quarters of a text, for example. Looking at the
proposed word-level analysis, however, it can be
observed that most LIWC categories did not exhibit
a strong correlation. In general, words captured by
LIWC are limited since it is a human-made lexicon
that might not capture the most revealing terms for
political bias. Furthermore, the categories are only
used at the word level. However, more complex
constructs like political bias tend to show at higher-
level text granularity (Chen et al., 2018; Iyyer et al.,
2014), e.g., in phrases or sentences, and hence a
standalone LIWC analysis is not as meaningful in
the detection of political bias.

Multi-Task Ordinal Regression

Baly et al. (2020) proposed a multi-task ordinal
regression to detect the bias of entire news outlets
in combination with a trustworthiness estimation.
Specifically, the authors modeled the left-right bias
on a 7-point scale (extreme-left, left, center-left,
center, center-right, right, extreme-right) and
factuality, which has been used as a substitute
for trustworthiness by the authors, on a 3-point
scale (low, mixed, high). Their approach was
motivated by the observation that center media
tends to be more impartial than hyperpartisanship
media, which tends to be more emotional, i.e., less
factual reporting. Compared to previous studies
that looked at the detection of trustworthiness and
political bias independently, Baly et al. (2020)
reported significant performance improvements
for the joint model. They collected multiple
articles from the target medium and derived
part-of-speech tags, linguistic features as well as
word embeddings. The authors used a model to
approximate the learning of the joint probability
density function between political bias and
factuality. They found a joint model in which
political bias is considered on 3- and 5-point scales
as auxiliary tasks yielded the best performance
at a mean absolute error of 1.475. An accuracy
score to compare this approach to the ones pre-
sented in the previous subsections was not reported.

One limitation of the study by Baly et al. (2020)
is the fact that they evaluated the political bias of
entire news outlets. While their study is based on
sample articles from each outlet, their final results
refer to the outlet itself. However, the evaluated
bias of the outlet does not necessarily reflect the
bias of future articles. Furthermore, the 7-point
Likert scale used for classifying political bias goes
beyond the universal left-right classification and
can exhibit more regional idiosyncrasies (Tavits
and Letki, 2009), which could reduce the validity
of the results.

4 Political Bias Mitigation

The mitigation of political bias is a new field with
little published research up to date. The most
promising study to decrease political bias was pub-
lished by Liu et al. (2021), who proposed an ap-
proach called ‘Reinforced Calibration’ for autom-
atically generated text. This method is also the only
work on political bias mitigation we are aware of
to date.

Reinforced Calibration

When generating text based on language models,
text prompts like ‘I think about marijuana because’
are used. Liu et al. (2021) found that attributes
such as gender, location, or topic have a significant
influence on the political bias of the subsequently
generated text. For example, for the sample
sentence above, a GPT-2 language model generates
the liberally-biased supplement ‘I believe it should
be legal and not regulated.’. A noteworthy aspect
that the authors found was that even conservative
prompts were completed with liberal output by

GPT-2.

To approach the task of mitigating such political bias in generated text, Liu et al. (2021) kept the main GPT-2 architecture but added a debiasing stage, with which the original text generation was re-calibrated to produce unbiased output based on multiple steps of "reinforced optimization" (Liu et al., 2021). They defined a state at step $t$ as all previously generated tokens and an action as the next output token. The policy in this context was the softmax output of the last hidden state, as this could be taken as the probability to choose a specific token, i.e., an action in this reinforcement learning setting, according to Dathathri et al. (2019). The authors further used a debias reward to guide the reinforced optimization. In this context, they employed two different rewards: a word-embedding-guided and a classifier-guided debias reward.

A word-embedding-guided debias reward was used in previous studies to force what are considered neutral words to be equally apart from topic-sensitive words in the embedding space, e.g., gender (Zhao et al., 2018; Park et al., 2018; Bolukbasi et al., 2016). Liu et al. (2021) used this approach to pick the next unbiased token at each time step. However, an issue with this approach is that political bias tends to occur at higher granularity levels (see Section 3 for more details) instead of at the word level. Furthermore, this approach is dependent on the quality of previously defined political bias words, which can have a significant impact on the final results (Zhou et al., 2019; Liu et al., 2021).

The classifier-guided debias that the authors additionally employed helped alleviate these issues. It was based on two different auxiliary tasks: Firstly, a political bias classifier was used to evaluate whether the text at hand was objective or not. Secondly, a constraint was introduced in the form of the Kullback-Leibler divergence between the original and the newly debiased policy to regulate the shift away from the vanilla softmax output, which might cause limited semantic coherence. Both components were balanced in the process of the reinforced optimization.

Liu et al. (2021) found that with regards to the considered attributes in the prompts, Reinforced Calibration was able to reduce the political bias in the generated text while maintaining readability. Comparing their debiasing results, they further found that the word-embedding debias reward led to worse performance than the classifier-guided debias reward.

An advantage of this approach is that the underlying language model does not have to be accessed or retrained; instead, an additional debiasing layer can be added, significantly reducing the necessary computing power and time. While the authors used the GPT-2 architecture in the paper as a base, the idea is also easily expandable to other language models through the addition of the debias stage. However, a drawback is that in this study, the focus was only on binary outputs, i.e., left or right ideologies, and an extension to more fine-grained political bias distinctions is non-trivial. Another aspect to consider regarding the mitigation of political bias is that through the additional layer of the Reinforced Calibration approach, additional noise is introduced, which might cause the overall performance of the respective NLP model to decline.

5 Future Research Directions

This paper highlighted ethical implications of political bias in text and summarized contemporary studies that focus on the detection and mitigation of political bias in NLP. We further analyzed the advantages as well as drawbacks of the individual methods.

So far, the limited existing approaches have not been evaluated in a unified framework. This paper addressed this gap to allow for a more exhaustive discourse of the topic at hand. We also found that while multiple authors have addressed political bias detection, the mitigation of such bias remains understudied. In this final section, we present a non-exhaustive overview of how to address the most severe shortcomings in the research, which we identified in our review to foster research in the area of political bias in NLP.

Standardized Definitions, Benchmarks, and Data

Due to the relative recentness of the subject, standardized definitions, evaluation metrics, and benchmarks are missing to measure political bias in
text. While we recognize that different applications might require different standards, this area should be addressed in future research. Especially the usage of different data sets and labeling instructions for politically biased text limits the comparability of contemporary studies. This issue is aggravated because political bias is evaluated on different levels: Some authors consider political bias on a news outlet, some on an article, and some on a sentence level. This divergence ties in with the lack of a standardized gold-standard political bias data set at the sentence level, limiting the progression of research in the field and should therefore be addressed.

Non-binary Political Bias

In all reviewed studies, the political spectrum considered was limited. Most studies focused on a binary left-right classification of political partisanship. More nuanced political ideologies were being disregarded. Future work could follow two directions regarding this issue: In supervised approaches, more nuanced political ideologies could be taken into account. On the other hand, unsupervised approaches could help discover the variety of political ideologies present and prevent limitations through pre-defined political affiliations.

Application of Bias Mitigation Techniques from Other Bias Types

Methods from other NLP bias analyses could be considered to mitigate political bias in NLP tasks. For example, data augmentation methods could be used to decrease political bias in generated text. This approach could be successful if disproportionate class distributions in the data cause political bias in NLP applications. Data augmentation was previously implemented for gender and race bias (Zmigrod et al., 2019; Yucer et al., 2020). In the case of gender bias, (Zhang et al., 2020) augmented the training data such that the gender in sentences was swapped and the algorithm was trained on the combination of the old and the augmented data. Kusner et al. (2017), on the other hand, used an approach in which data samples were treated equally in actual and counterfactual demographic groups, which could be extended to political partisanship, too.

Another approach to consider would be embedding manipulations. Garg et al. (2018) found that societal biases are reflected in word embeddings, which is likely valid for political bias as well. With regards to gender bias, this was studied, for example, by Bolukbasi et al. (2016). The authors ensured that gender-neutral word embeddings were orthogonal to a gender direction defined by gender-bias words selected through a classifier. Zhang et al. (2020) built on this method and tried to force neutral words to have an equal distance to pre-defined groups of sensitive words to obtain a gender-neutral embedding space. In addition, Zhou et al. (2019) retrained language models with a fairness loss to ensure unbiased text generation.

These approaches rely on the retraining of the underlying language model, which is often not available (such as in the case of GPT-3) or computationally costly. Nevertheless, a comparison between complete language model retraining approaches and Reinforced Calibration that focused on adding a debiasing layer should be conducted to evaluate the performance in both settings and assess which one is more effective in mitigating political bias.

Mitigating and Detecting Political Bias in Languages Other Than English

The considered studies only focused on English text. In future work, existing techniques could also be applied to political bias in other languages. However, this is non-trivial for two reasons: Firstly, especially in countries other than the U.S., the party landscape is often more diverse, and the differentiation between political camps is more nuanced, which might be harder to be picked up by NLP models. Secondly, most politically-oriented corpora are English, and hence there would be a need to create complementary training data. With regards to both detection and mitigation approaches, an extensive training set is salient and needs to be created before considering the transfer of existing approaches to other languages.

6 Conclusion

Political bias detection and mitigation in NLP is an emerging field. Due to the increased usage of NLP and its potential to propagate societal biases, it is vital to address such problems early to unify efforts within the research community. To the best of our knowledge, this is the first review paper to address the state of the research in this area. We further suggested research opportunities to advance the detection and mitigation of political bias in NLP methods.
References


Johannes Kiesel, Maria Mestre, Rishabh Shukla, Emmanuel Vincent, Payam Adineh, David Corney,


Matt Thomas, Bo Pang, and Lillian Lee. 2006. Get out the vote: Determining support or opposition from congressional floor-debate transcripts. arXiv preprint cs/0607062.


