A novel subjective bias detection method based on multi-information fusion

Lidan Zhao[a], Tong Li[a,*], Zhen Yang[a], Junrui Liu[a]
[a] Beijing University of Technology
*litong@bjut.edu.cn

***Abstract*—News and social media messages usually contain subjective opinions conflicting with the needs of readers who want to receive objective information through public channels. To this end, the detection of subjectively biased sentences has become an important research issue. However, existing subjective bias detection approaches lack considering the syntactic structure and topical context of biased descriptions. In this paper, we propose a Subjective bIas deTection mEthod (SITE) that comprehensively fuses multiple bias-relevant information. Specifically, we first investigate the modification and lexical features of biased sentences, based on which we formulate a set of rules to characterize biased sentences. Then, we extract the semantic features of sentences using the BERT model, based on which we further mine topic features by clustering semantically similar sentences. Finally, we comprehensively characterize biased sentences by fusing such features and train a classification model to detect biased sentences in social media. We conducted a series of experiments on a public dataset, the results of which show that SITE can detect biased sentences with 86.2% accuracy, outperforming baseline methods.**

***Index Terms*—Bias Detection, Dependency Structure, Topical Context**

I. Introduction

In recent years, there has been an increasing amount of research on bias in textual representations, including word bias detection and sentence bias detection. Word bias can be caused by people's stereotypes, such as gender and racial bias. The current research on sentence bias is mainly concerned with subjective bias. Subjective bias occurs when the language that should be neutral and fair is skewed by feeling, opinion, or taste (whether consciously or unconsciously) [1]. People may inadvertently add their subjective ideas when recording opinions, introducing subjective bias into the text and affecting readers' thinking. For example, "John McCain exposed as an unprincipled politician", which expresses the editor's negative view of the subject, would be more neutrally expressed as "John McCain described as an unprincipled politician". By studying bias in text, objective facts can be conveyed to readers more neutrally. In this way, the independent thinking ability of readers can be improved, and at the same time, readers can be more clear about their views on objective things.

Existing works believe that classifying subjective bias can better identify the words that trigger bias in a text. They detect biased sentences by identifying specific lexical cues [1], [2]. Some scholars focus on syntactic features to classify sentences [3]. However, extracting linguistic cues or certain syntactic features is insufficient to detect subjective bias because this information is present in most sentences. In addition, biased sentences have some other common features which help implement the bias detection task. Some scholars use deep learning models to mine the hidden semantic representation of sentences and combine sentence semantics and syntactic information as features for bias detection [4], [5]. However, these studies ignore other information, such as the topic information of sentences, bias category information, etc. The topic of the sentence can represent the background information associated with the semantics of the sentence. Sentences with similar topics have a close probability of bias, so topic features are also an important feature for detecting bias.

In this paper, we propose a Subjective bIas deTection mEthod (SITE) that comprehensively investigates and combines dependency, semantic, and topic features of sentences. Specifically, dependency features are defined by systematically investigating biased sentences and condensing biased features, resulting in a series of features of modification structure. Moreover, we mine the deep semantic features of sentences by adopting the Bidirectional Encoder Representations from Transformers (BERT) [6] model to complement the dependency features. On top of such semantic features, we further identify the topical context of sentences by clustering the sentence embeddings. Eventually, we fuse these three types of features and train a subjective bias detection model. The main contributions proposed in this paper are as follows:

- We investigate the modification structures that tend to trigger subjective bias and combine them with biased words to define dependency features.
- We consider the topical context of sentences by investigating and mining potential relationships among similar semantic sentences, based on which we propose a subjective bias detection method that combines dependency features, semantic features, and topic features.
- We conducted experiments on a public dataset, the results of which show that SITE can detect biased sentences with 86.2% accuracy.

The rest of the paper is structured as follows: Section II discusses related works. Section III describes the design details of the proposed approach. In section IV, we evaluate our approach on a public dataset and compare the performance with the existing approaches. Finally, we conclude in section V and look forward to future works.

DOI reference number: 10.18293/SEKE2023-103.

II. Related Work

Recent studies on bias focus on word bias detection [7]–[9] and sentence bias detection [3]–[5], [10].

A. Word bias detection

Researchers use models such as Word2vec [11] and GloVe [12] to embed the words as vectors, comprehensively use the relationships between words, and apply contextual semantics to detect biased words. Bolukbasi et al. [7] first identify the bias subspace and determine the direction in which the embedding vector captures the bias. They then calculate the cosine distance to judge the similarity between the embedding vectors and the bias vectors, thereby detecting the presence of bias. Kumar et al. [8] define an indirect bias based on this for studying gender bias, which considers not only the relationship between each word in the text and the bias vector but also whether the two-word vectors are strongly related. Manzini et al. [9] determine the presence of bias by calculating the similarity between the detected words and each bias vector. These studies detect word-level bias by comparing the biased word vector direction with the word vector direction.

In their work, they consider the vector relationship between words and biased words to implement the word bias detection task. When we perform sentence bias detection, similar to this, we use hidden vectors to represent various features of sentences and then detect whether bias exists.

B. Sentence bias detection

Researchers detect bias by extracting lexical, syntactic, semantic, and other features of sentences. May et al. [10] use the Sentence Encoder Association Test (SEAT) method to map sentences into fixed-size vectors and determine whether a sentence is biased or not by calculating the salience of the association and the size of the association in different vector spaces. Liang et al. [13] refer to the bias between words in a sentence as fine-grained local bias and the bias between semantics in a sentence as high-level global bias. Sentence-level bias detection is performed in two parts. The first part is calculating the contextual probability between the word vector and bias vector to identify local bias. The second part identifies global bias through sentence sentiment scores and regard scores.

Hube et al. [3] focus on cases of sentence-level linguistic bias in Wikipedia and propose DMSW. DMSW is a supervised classification method that relies on automatically creating biased words, as well as other syntactic and semantic features of biased statements. They analyze the proportion of words with bias in the sentence and their context, LIWC [14] features, and the framing bias and epistemic bias of the words in context for bias detection. Hube et al. [5] focus on the specific case of phrasing bias, which may be introduced through specific inflammatory words or phrases in a statement. They propose an RNN-based classification model for biased statements, extracting sentence hidden semantic representations to capture the inter-dependencies between words in phrases that introduce bias while incorporating LIWC features for text classification. Pant et al. [4] explore various BERT-based models, including BERT, RoBERTa, and ALBERT, with their base and large specifications along with their native classifiers. To extract the semantic information of sentences, an integrated BERT-based model is proposed for detecting subjective bias in Wikipedia.

III. Method

Our method consists of three modules: dependency feature extraction module, semantic feature extraction module, and topic feature extraction module. The overview of SITE architecture is shown in Fig. 1.

The task of the first module is to extract the dependency features of sentences. According to the dependency analysis results, SITE determines and fuses the modification structure. At the same time, SITE determines the biased words and calculates the proportion of the biased words to extract the dependency vector of a sentence. In the second module, we use the BERT model to consider the correlation between tokens to generate global information about the sentence, that is, the semantic vector of the sentence. The third module extracts the topic features of sentences. SITE trains a topic model by clustering the embedding vectors of sentences and extracting topic words for each cluster using a class-based variant of TF-IDF. Finally, the semantic vector of the sentence is used as the input of the model to obtain the topic vector. The three feature vectors are concatenated as the feature vector of the sentence. Based on these features, the classifier identifies whether there is a subjective bias in the sentence.

A. Dependency feature extraction

By investigating the common features of biased sentences, we find that some sentence structures or words are less likely to appear in neutral sentences. SITE defines these modification structures and lexical features in these sentences as dependency features.

1) Modification features: The longer a sentence is, the more complex its syntactic structure is, and the richer its meaning is. Biased sentences often convey subjective ideas through the use of nested modification structures. Such modification structures are constructed based on the words in the sentence. Different relationships between words will be biased to varying degrees. In biased sentences, the relationship that has a large influence on subjective bias appears more likely. Therefore, we perform dependency analysis on the sentences and then define the modification structures associated with bias.

Dependency parsing shows the subordination and modification relationship between words. The dependencies of a sentence can be represented by a diagram.

For example: “the British Broadcasting Corporation or BBC is the most widely respected broadcasting organisation in the world”. The analysis of the dependencies is shown in Fig. 2. There is an arrow pointing from “organisation” to “broadcasting”, indicating that “broadcasting” modifies “organisation”. The relationship on the arrow is “compound”, indicating that “broadcasting” is a noun compound of “organisation”, which is used to modify “organisation”. Similarly, “respected”

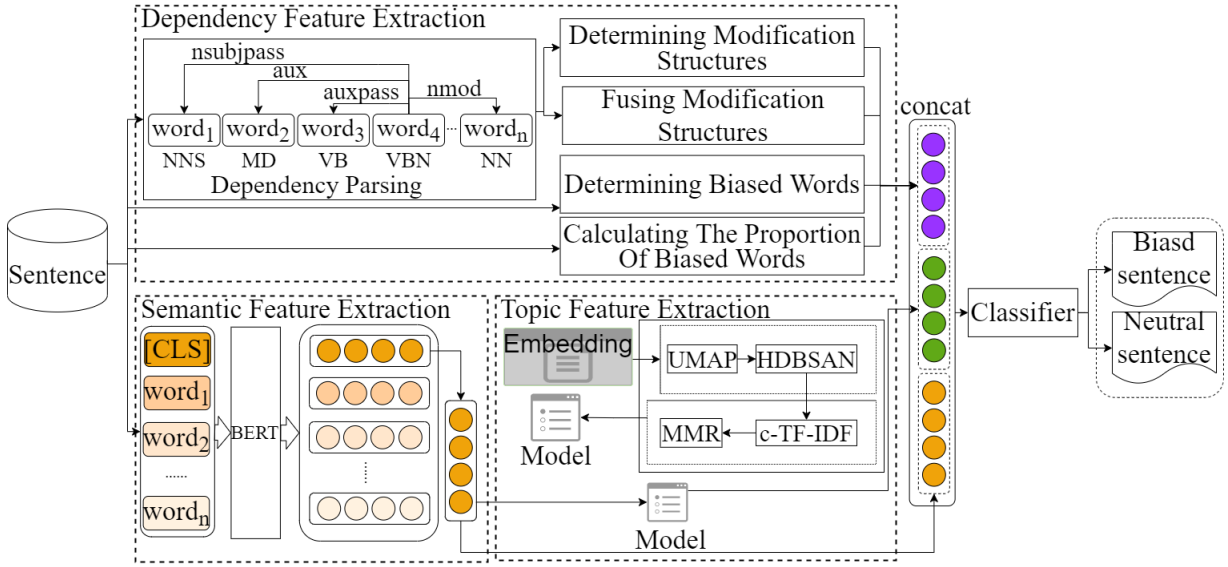


Fig. 1. Overview of SITE

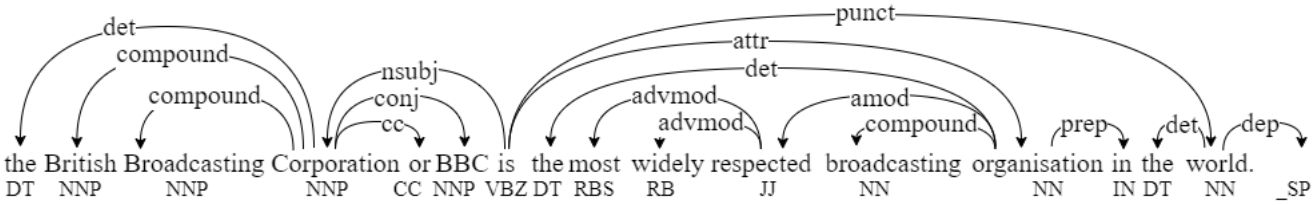


Fig. 2. The analysis of the dependencies

depends on "organisation", which is an adjective modifier of "organisation". Furthermore, "det" is the relation between the head of an NP and its determiner. "nsubj" is a nominal which is the syntactic subject and the proto-agent of a clause. "cc" is the relation between an element of a conjunct and the coordinating conjunction word of the conjunct. "advmod" is a (non-clausal) adverb or adverbial phrase (ADVP) that serves to modify the meaning of the word. In addition to these relationships, the other relationships represent the modification relationship between the words at both ends of the arrow. For specific meanings, please refer to the Universal Dependencies website[1] for more information. In Fig. 2, the tags below the words are part-of-speech (POS) tags of the words. "DT" stands for determiner. "NNP" stands for proper noun. "CC" stands for coordinating conjunction. "VBZ" stands for 3rd person singular. "RBS" stands for the superlative. In addition to these POS tags, more detailed explanations of the labels can be found on the Universal Dependencies website[2].

Based on the dependency parsing of biased sentences, we find some common modification structures in biased sentences that introduce optimistic or pessimistic views of objective facts. The modification structures we define include the same noun or proper noun modified by multiple adjectives, and a noun or proper noun modified by both an adjective and a noun. For example, in Fig. 2, "the most widely respected broadcasting organisation" is an explanation of the BBC. In the sentence, "organisation" is modified two times to express a stronger admiration, which leads to a biased sentence. In such a case, the likelihood of adding subjective emotions is greatly increased. In addition to this, we extract several other structures (Table I). We define such structures as atomic features (AF), modification structures that are more likely to occur in biased sentences.

Complex modification structures help editors express their subjective emotions, so sentences with multiple modification features are more likely to develop a subjective bias (Table I). For example, the structure AF1 and the structure AF4 are more likely to coexist in biased sentences. Therefore, we determine whether both structure AF1 and structure AF4 are present in the sentence. In addition, we combined structures AF2 and AF5, and structures AF1, AF3, and AF4. The details of the combined features are shown in Table II.

[1]https://universaldependencies.org/en/dep/
[2]https://universaldependencies.org/u/pos/

TABLE I
DESCRIPTION OF THE MODIFICATION STRUCTURE.

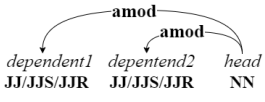
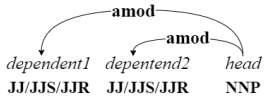
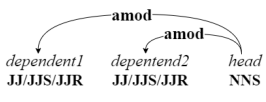
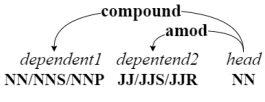
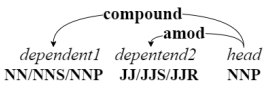
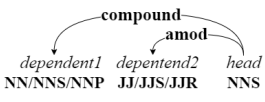
No.	Structure	Example
AF1		"the Bugatti Veyron 16.4 is the *fastest (dependent1) street-legal (dependent2)* production *car (head)* in the world."
AF2		"*brilliant (dependent1)* but *pompous (dependent2)* entrepreneur Mark *(head)* is the envy of all his colleagues."
AF3		"*The Great Global Warming Swindle* argues against *prominent (dependent1) scientific (dependent2) views (head)* on global warming."
AF4		"African poverty is due to the *rampant (dependent1) government (dependent2) corruption (head)* on that continent."
AF5		"*famous (dependent1)* Indian *singer (dependent2) Sonu Nigam (head)* sang many songs of Akhlaq Ahmed."
AF6		"his novels were *real (dependent1) page-turners (dependent2-head)*, but grounded on meticulous historical research."

TABLE II
FUSING THE MODIFICATION STRUCTURE.

No.	Structure	Example
CF1	AF1&AF4	"Missouri governor Lilburn Boggs issued the *ominous sounding extermination order* (AF1&AF4)."
CF2	AF2&AF5	"Kristin Shepard is a fictional character on the *popular American television series* (AF2&AF5)."
CF3	AF1&AF3&AF4	"the most *famous German "Panzer Ace"* (AF1&AF4) is credited by Kurowski as having destroyed 60 tanks and nearly as *many anti-tank guns* (AF3)."

2) Subjective bias lexical features: Dependency parsing is a more detailed analysis of the syntax in a sentence. It analyses the relationship between the constituents of a sentence, with less analysis of the semantic aspects. For example, in the analysis of "the most widely respected broadcasting organisation" and "the most widely regarded broadcasting organisation", the dependency parsing can be interpreted as a modification of the organisation. However, in the actual context, the former expresses a positive sentiment towards the organisation. This shows that the identification of the modification features in the sentence alone does not reveal the features that trigger the bias. Therefore, it is necessary to pay attention to the biased words in the sentences, especially in the modification structures.

Subjective bias lexical features mainly refer to biased words. Biased words mainly capture the subjective sentiment expressed in the sentence, including factive verbs, assertive verbs, implicatives, and other entailments, hedges, and subjective intensifiers [2]. For example, "say" and "state" are usually neutral, "pointing out" and "claim" cast doubt on the certainty of the proposition [2].

Suppose the modification structure in Table I, II exists in a sentence, and the subordinate words in the structure are words with subjective sentiment. In this case, the sentence is more likely to have subjective bias. In this paper, we use the biased words table provided by Pryzant et al. [1] to identify bias modifiers in modification structures.

A biased word is the smallest unit of biased expression in a sentence, and it is a way to express subjective ideas. Therefore, in addition to incorporating it into the modification structure, we consider the number of biased words in the sentence. In a sentence, if there are more biased words, there is a greater probability that the sentence has a subjective bias. Therefore, it makes sense to use the proportion of biased words in a sentence as auxiliary information for the bias detection task.

Our method extracts the features defined above as dependency vectors of sentences, which are used in the bias detection task.

B. Semantic feature extraction

The representation of subjective bias in sentences is subtle, and merely extracting dependency features is insufficient to detect bias. In addition to this, semantics is crucial information for sentence classification tasks. BERT [6] is an Autoencoder language model that works well with contextual information to generate sentence semantic vectors. It uses the Masked Language Model to pre-train a bidirectional Transformer to generate language representations capable of incorporating contextual information. It is trained on a large corpus, and then the model is fine-tuned for our tasks by adding some extra layers at the end, which can be classification, question answering.

In this paper, we fine-tune the BERT model [6] to extract hidden vector representations of sentences. The input of BERT is the representation of each token in the sentence. To complete the classification task, in addition to the token, a specific classification token ([CLS]) needs to be inserted at the beginning of the sequence. BERT focuses on information from different representation subspaces at different positions by using the multi-head attention mechanism of the encoder in the Transformer and weighs the correlation between words. Specifically, the input is represented by $(x_1, x_2, ..., x_n)$, and the corresponding embedding vector $(a_1, a_2, ..., a_n)$ is generated through the BERT embedding layer. Multiple heads are generated by computing the embedding vector with multiple sets of Q, K, and V using an attention mechanism. Then, it merges the heads and dot-multiplies W^O for a linear

transformation to generate an output corresponding to each token. The specific calculation is as (1):

$$\begin{aligned} MultiHead(Q,K,V) &= Concat(head_1,...,head_h)W^O \\ \text{where} \quad head_i &= Attention(QW_i^Q, KW_i^K, VW_i^V) \end{aligned} \tag{1}$$

where the parameter matrices $W_i^Q \in R^{d_{\text{model}} \times d_k}$, $W_i^K \in R^{d_{\text{model}} \times d_k}$, $W_i^V \in R^{d_{\text{model}} \times d_v}$ and $W^O \in R^{hd_v \times d_{\text{model}}}$. Q, K, and V represent query, key, and value vectors, respectively. d represents the dimension.

Therefore, after the 12 layers of encoder, the output of last layer corresponding to token is used to aggregate the representation information of the entire sequence. Because each word token has its meaning, its semantics account for a large proportion of the final semantic vector. However, [CLS] has no semantics. The output of the last layer corresponding to it can more fairly express the semantic information of the entire sentence. Therefore, by training the BERT model, we extract the output of the last transformer layer corresponding to [CLS] as the sentence semantic vector. We use this vector as part of the feature vector of the classifier for the subjective bias detection task.

C. Topic feature extraction

Subjective bias may come from differences in social background, cultural background, or other factors. The likelihood of a subjective bias occurring in a sentence varies under different topics. For example, subjective bias is more likely to occur under topics such as religion, society, competition, and politics, while it is less likely to occur under topics such as philosophy and music. Understanding the topic context can help us better understand the text and detect subjective bias in the text. For example, suppose we are analyzing a text about politics. In that case, we can better understand the ideas and messages in the text if we understand the subject context of the political position to which the text belongs, the history of that political position, and the current political environment to detect subjective bias more accurately. Therefore, topic features of texts can help us better understand texts and play a crucial role in bias detection. The sentence topic is the analysis of the underlying semantics of a sentence. We improve the overall performance of the model by extracting topic features of sentences and mining biased correlations under the same topic.

Techniques for topic modeling fall into two main categories: one is bag-of-words-based models. Such as LDA (Latent Dirichlet Allocation) [15], NMF (Nonnegative Matrix Factorization) [16], etc. The other category is clustering methods based on pre-trained word embeddings. For example, CTM (Correlated Topic Model) [17], BERTopic [18]. Bag-of-words-based methods do not fully consider the contextual semantics of each word. Therefore, we prefer to use topic models based on pre-trained word embeddings. General topic models use density-based methods for clustering but centroid-based methods for sampling topic words. The inconsistency of these two methods leads to sampling topic words from other clusters, which makes the topic representation inaccurate. Our method uses the BERTopic model. The model uses a hierarchical and density-based approach to clustering, using a class-based TF-IDF variant to sample each cluster's words. The specific calculation is as (2):

$$W_{x,c} = tf_{x,c} \cdot log(1 + \frac{A}{f_x}) \tag{2}$$

where c refers to the cluster we created before, $tf_{x,c}$ represents the frequency of word x in class c, f_x represents the frequency of word x across all classes, A represents average number of words per class. The importance score of each word x in each class is obtained by calculating the above formula. This way, sampled topic words are constrained to corresponding clusters, and the inconsistency between clustering and sampling words is solved.

In this paper, we use the sentence vectors generated by the BERT model as sentence embeddings to train BERTopic. To reduce computational complexity and memory while speeding up computation, BERTopic uses UMAP (Uniform Manifold Approximation and Projection) [19] for dimensionality reduction of sentence embeddings. UMAP can preserve some high-quality embeddings and can discover more local semantic features. Additionally, text clustering is performed using HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise) [20]. BERTopic collects semantically similar sentences in a cluster and uses the class-based TF-IDF variant to extract the topic words of each cluster, thereby generating individual topic vectors for the entire text. By calculating the similarity between each sentence and the topic, we can get the probability of the sentence under each topic and use it as the topic feature.

D. Subjective Bias Classification

After extracting dependency, semantic, and topic features, we concatenate them to represent the sentence's feature vector. In addition, each sentence includes a label indicating whether it contains bias. We use an SVM (Support Vector Machine) [21] classification model to perform subjective bias detection.

The feature vectors we extracted have the characteristics of high dimensionality and nonlinearity separability. SVM represents the training data as points in space and constructs hyperplanes in high-dimensional or infinite-dimensional space to separate these points. This method can effectively handle high-dimensional and nonlinear data and has good interpretability. Therefore, we use SVM for classification.

SVM can use different kernel functions to classify different types of text data. By finding an optimal separating hyperplane to divide the data into two categories, SVM maximizes the model's predictive accuracy without overfitting the training data. When predicting whether a new sentence contains bias, its feature vector is mapped to the same space, and the category to which it belongs is predicted based on which side of the margin it falls on.

IV. Experiment

In this section, we first describe our dataset and further describe the experimental setup and results.

A. Dataset

There is only one open-source dataset for subjective bias detection work. The dataset we used is the Wiki Neutrality Corpus (WNC) dataset open-sourced by Reid Pryzant et al. [1]. The data is derived from the editing history of Wikipedia. Wikipedia follows three main principles when verifying entries: neutral point of view [3] (NPOV), available for verification, and non-original research. NPOV refers to editors presenting facts in a neutral way and recording opinions without taking a position. This dataset is widely used in work on subjective bias detection, and the NPOV principle in it fits well with our point of view. Therefore, we conducted experiments with the dataset [4] provided by Reid Pryzant et al. [1].

The dataset is an edited data crawl on Wikipedia. It contains about 180,000 biased sentences and 360,000 neutral sentences. We randomly shuffle these sentences and split the dataset into two parts: training and test sets. The training data is then used to train the classification model and the retained test data is executed for evaluation.

B. Experimental Settings

We train a classifier on the feature set mentioned in section III. We first extract the dependency vectors of sentences according to the defined dependency features. Then we train the BERT model with the sentences and their labels as input and extract the last layer vector corresponding to [CLS] as semantic vectors. For the BERT, we use a learning rate of $2 * 10^{-5}$, a maximum sequence length of 128, a batch size of 32, and training epochs of 3 while fine-tuning the model. Finally, we use the trained BERT model to generate sentence embedding vectors to train the BERTopic model and extract the topic probabilities of sentences as topic vectors. For BERTopic, we train the model with the following hyperparameters: top_n_words of 10, n_gram_range of (1, 1), and min_topic_size of 10. The above three feature vectors are concatenated to generate the final feature vectors. We train the Support Vector Machine (SVM) model with this vector as the input to the classifier. SVM maps the feature vectors to some points in embedding space and finds a hyperplane to segment the samples to complete the classification task.

We use precision (P), recall (R), F1 score, and accuracy (ACC) as evaluation metrics to assess the results of the experiments. Precision measures the percentage of sentences predicted as having a subjective bias that has bias. Recall measures the percentage of all biased sentences in which the bias was correctly identified. Accuracy measures the percentage of the number of predicted labels that are correct. The F1-score is the summed average of the precision and recall. Thus, better classification models have higher P, R, F1, and ACC.

[3]https://en.wikipedia.org/wiki/Wikipedia:Neutral_point_of_view

[4]http://bit.ly/bias-corpus

We compare our method to four existing baselines that focus on the same task as ours and have been more effective over the past five years.

- Recasens et al. [2]: the method is to detect subjective bias in sentences using common linguistic cues from different bias categories.
- DBWS [3]: a supervised text classifier based on biased vocabulary and other features. This method uses a random forest classifier for the bias detection task.
- Hube et al. [5]: an approach that relies on recurrent neural networks. The method mines the inter-dependencies between words in phrases that introduce biases.
- Pant et al. [4]: BERT-based model conducts comprehensive experiments to detect subjective bias, with the same purpose of experiments in this paper.

C. Experimental Results

Table III shows the experimental results of SITE compared to baseline. The best results and the second-best results are highlighted in bold and underlined, respectively. SITE outperformed other baselines, achieving a precision of 0.829, recall of 0.713, F1 score of 0.767, and accuracy of 0.862, all of which are higher than other methods.

TABLE III
Comparison of experimental results.

	P	R	F1	ACC
Recasens et al.	0.758	0.206	0.324	0.712
DBWS	<u>0.778</u>	0.276	0.408	<u>0.732</u>
Hube et al.	0.531	0.548	0.540	0.710
Pant et al.	0.733	<u>0.677</u>	<u>0.704</u>	0.716
SITE	**0.829**	**0.713**	**0.767**	**0.862**

The first baseline [2] and the second baseline [3] mainly analyze the lexical and syntax of sentences. From the experimental results, it can be seen that by analyzing the lexical and syntactic features, only a small part of biased sentences can be identified. So the recall is lower, 0.206 and 0.276, respectively.

Hube et al. [5] mines the hidden semantic representation and LIWC features of sentences, etc. This method uses RNN for semantic representation. RNN cannot support long-term sequences and is not suitable for processing long sentences. But sentences with subjective bias generally have complex syntactic features and are relatively long, so the classification effect is relatively poor. Compared to Hube et al. [5], our method improved P by 29.8%, R by 16.5%, F1 by 22.7%, and ACC by 15.2%.

Pant et al. [4] using the BERT model to mine sentence information, its context can be taken into account, and the classification effect is relatively good. The overall performance of this method is second only to our method. However, the sentence information extracted by this method is single. On this basis, this paper extracts topic features, mines sentence modification features, biased words, etc., and achieves better results. Compared to the Pant et al. [4], our method improved P by 9.6%, R by 3.6%, F1 by 6.3%, and ACC by 14.6%.

This experimental result proves that considering sentence dependency, semantic, and topic features, it is possible to fully mine sentence information and improve the overall performance of the classification model.

V. Conclusion

In this paper, we propose a subjective bias detection model that fuses multiple kinds of information. The model fuses dependency features, semantic features, and topic features to classify sentences.

Dependency features are more inclined to judge simple sentences and sentences with explicit expressions. This is because dependency features represent modification structures in sentences. When the biased sentence is simple, modifiers to the main body of the sentence are often added to the sentence to introduce bias. When a sentence is clearly expressed, there will be a significant number of modification structures. In this case, the dependency feature of the sentence plays an important role in detecting bias, and the accuracy of the result is high. Topic features are more inclined to judge sentences with rich context. Because extracting the topic features of a sentence is equivalent to extracting various information such as background and history under a certain topic to which the sentence belongs. When there are more sentences under the same topic, the more information it contains, and the more accurate the subjective bias detection under the topic is. Semantic features are the key information for sentence feature extraction, which can handle the context information of sentences well and are applicable to any sentence. The results show that considering these three types of features can better highlight the biased features of sentences, which helps improve model performance. The completion of this classification task dramatically helps readers to read texts and also helps to detect biased texts on Wikipedia.

Subjective bias does not exist explicitly in the text, and we need to mine the hidden information. We need to interpret and extract sentence features to detect bias further. However, a complete elaboration of sentence features is complex and requires continuous research to detect bias further. It makes sense to automate the removal of subjective biases from text and generate more objective expressions. In future studies, we will conduct bias neutralization studies.

Acknowledgement

This work is partially supported by the National Key R&D Program of China (No.2022YFB3103100), the Major Research Plan of the National Natural Science Foundation of China (92167102), the Project of Beijing Municipal Education Commission (No.KM202110005025), the Importation and Development of High-Caliber Talents Project of Beijing Municipal Institutions (CIT&TCD20190308), and Engineering Research Center of Intelligent Perception and Autonomous Control, Ministry of Education.

References

[1] R. Pryzant, R. D. Martinez, N. Dass, S. Kurohashi, D. Jurafsky, and D. Yang, "Automatically neutralizing subjective bias in text," in *Proceedings of the aaai conference on artificial intelligence*, vol. 34, 2020, pp. 480–489.

[2] M. Recasens, C. Danescu-Niculescu-Mizil, and D. Jurafsky, "Linguistic models for analyzing and detecting biased language," in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2013, pp. 1650–1659.

[3] C. Hube and B. Fetahu, "Detecting biased statements in wikipedia," in *Companion proceedings of the the web conference 2018*, 2018, pp. 1779–1786.

[4] K. Pant, T. Dadu, and R. Mamidi, "Towards detection of subjective bias using contextualized word embeddings," in *Companion Proceedings of the Web Conference 2020*, 2020, pp. 75–76.

[5] C. Hube and B. Fetahu, "Neural based statement classification for biased language," in *Proceedings of the twelfth ACM international conference on web search and data mining*, 2019, pp. 195–203.

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[7] T. Bolukbasi, K.-W. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai, "Man is to computer programmer as woman is to homemaker? debiasing word embeddings," *Advances in neural information processing systems*, vol. 29, 2016.

[8] V. Kumar, T. S. Bhotia, and T. Chakraborty, "Nurse is closer to woman than surgeon? mitigating gender-biased proximities in word embeddings," *Transactions of the Association for Computational Linguistics*, vol. 8, pp. 486–503, 2020.

[9] T. Manzini, Y. C. Lim, A. W. Black, and Y. Tsvetkov, "Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings," in *NAACL-HLT (1)*, 2019.

[10] C. May, A. Wang, S. Bordia, S. R. Bowman, and R. Rudinger, "On measuring social biases in sentence encoders," in *NAACL-HLT (1)*, 2019.

[11] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, Y. Bengio and Y. LeCun, Eds., 2013.

[12] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[13] P. P. Liang, C. Wu, L.-P. Morency, and R. Salakhutdinov, "Towards understanding and mitigating social biases in language models," in *International Conference on Machine Learning*. PMLR, 2021, pp. 6565–6576.

[14] J. W. Pennebaker, M. E. Francis, and R. J. Booth, "Linguistic inquiry and word count: Liwc 2001," *Mahway: Lawrence Erlbaum Associates*, vol. 71, no. 2001, p. 2001, 2001.

[15] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[16] D. Lee and H. S. Seung, "Algorithms for non-negative matrix factorization," *Advances in neural information processing systems*, vol. 13, 2000.

[17] G. Xun, Y. Li, W. X. Zhao, J. Gao, and A. Zhang, "A correlated topic model using word embeddings." in *IJCAI*, vol. 17, 2017, pp. 4207–4213.

[18] M. Grootendorst, "Bertopic: Neural topic modeling with a class-based tf-idf procedure," *arXiv preprint arXiv:2203.05794*, 2022.

[19] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," *stat*, vol. 1050, p. 18, 2020.

[20] L. McInnes, J. Healy, and S. Astels, "hdbscan: Hierarchical density based clustering," *J. Open Source Softw.*, vol. 2, no. 11, p. 205, 2017.

[21] C.-W. Hsu, C.-C. Chang, C.-J. Lin *et al.*, "A practical guide to support vector classification," 2003.