Bias in news article

Ankit Dahiya
MT19004
M.Tech CSE
ankit19004@iiitd.ac.in

Gowtham G
MT19011
M.Tech CSE
gowtham19011@iiitd.ac.in

Jain Piyush Pradeep
MT19122
M.Tech CSE
piyush19122@iiitd.ac.in

Abstract

Media plays an important role in influencing people and their views regarding some political parties. So, it important that they remain unbiased towards all political parties over a topic. This document contains the initial formulation and literature survey done on the selected problem related to finding biases in the news articles. It also talks about the baseline paper that is selected along with the methodologies that are used in it. The dataset that is being used is scraped from different news articles. Then, we propose the method that is used to find the bias in the news article and also find the particular part in which the bias is present. We proposed models based on SVM, Naive Bayes, CNN, and Bidirectional LSTM combined with CNN to find the biased news article, and the biased news sentences were tried using similarity between the sentences and LSTM.

1 Problem Definition

Finding the biases in the sentences of the news articles either towards Indian National Congress or Bhartiya Janta Party, two of the major parties of India. This finding can be used to categorize the media houses, which supports either of the party and identify the neutral articles. Also, the identification of the trueness of news will be easy.

2 Background

The increase in media houses increases the source of getting the news. The presence of social media makes it even easier for people to access these news articles. It is, therefore, very much convenient for users to get the information quickly, which can be the positive side. On the contrary, these news articles may be fake and may contain some bias towards a particular section of people. Because of the widespread reach of social media, biased news article amplifies, increasing the credibility of the news. However, there are some organizations which report unreliable news sources, but they get outdated very quickly, to identify the reliability of the news article quickly, automatic detection is necessary. On identifying these biases, it is easy to identify the media houses which supports a particular political party, because these sources are prone to stand towards one political party going against the other. Identifying these biases will also influence the factuality of the news articles because biased news is more prone to convey false news.

3 Baseline

Baseline model (Baly et al., 2018) has been implemented using SVM for predicting the bias. Dataset used in the baseline model had 243 input features since 141 input features were taken from body section as well as title section of the news medium and one of the features was the name of the news medium. Five-fold cross validation had been performed while applying SVM. GridSearchCV library had been used for finding the best values for kernel parameter, gamma parameter and C parameter of SVM. Non-linear kernel was found to perform better than linear kernel.

There were four metrics reported at the end – accuracy, macro-averaged F1 score, Mean Average Error(MAE) ,and variant of Mean Average Error(MAEm). MAEm gave a more accurate error value than MAE because MAEm could give good results in case of class imbalance in the dataset.

Score	Value
Macro-F1	55.91
Accuracy	57.48
MAE	0.41
MAEM	0.44

Table 1: Score for baseline.

4 Literary Review

In the reference paper, along with prediction of bias, trueness of the news (Factuality) was also predicted. Input features of target news medium were extracted from different sources like its Wikipedia page, its twitter account, news articles published by that news medium, its URL structure and its web traffic. There were totally 141 input features that were extracted from above mentioned sources of the target news medium. Bias prediction was done at news medium level rather than at the article level.

Some of the features that were extracted from the twitter account were hasAccount, isVerified, dateOfCreation, hasLocationInfo, numberOfFollowers etc. AlexaRank is used for finding about the web traffic of the target news medium. Fake news medium was found to use a lot of special characters in its URL. SVM was used to train the model with the collected dataset. Ablation studies was performed to find out the importance of different sources of target news medium. Articles related to news medium and Wikipedia page was found more important in the results of ablation study.

Along with this, we have also studied how we can apply deep learning concepts in the field of natural language processing. The major focus was on the use of CNN(Kim, 2014) and LSTM for the task of sentence classification.

5 Dataset Used

Our task is to find the bias in the news article related to one topic (here we have gone with the topic of CAA) for this, we have scraped 2876 articles collected from the different news channels. Url for the news was extracted using news API which is a standard API for extracting the news data. After we have the URL, news articles were extracted using the URL and newspaper library of python. Newspaper library is used to seperate HTML tags from the main content of the article. Inference from all the news article are present in the table. Refer table 2.

Our task is to find bias in the news article which are biased towards certain political party. So for finding the bias in news articles, we need annotated ground truth articles that are having biased in the article. So the first task we require is to annotate the articles in our dataset. Based on this the article is marked as either left bias, right bias, or neutral. Annotating the articles is a hard task because it requires the annotator to have thorough knowledge about the topic and another thing is that the annotator should not be biased himself and the annotations are based on the person's point of view. For this, from all 2876 articles, 905 articles were randomly selected which were manually annotated as either 1 (biased toward BJP), 0 (not biased), and -1 (biased against BJP or favors congress). Details about the same are present in the table 3.

Information	Value
No of different sources	139
Date Range	19/2/2020 - 17/2/2020
Most Frequent Sources	the Hindu, First Post, Indian Express,Times of India, Money control
Most Frequent Words	CAA, Delhi, Government, People Citizenship,
Average word in article	553

Table 2: Information related to data Set.

Meaning	Tag	Number of Articles
Biased towards BJP	1	192
Neutral	0	372
Biased against BJP	-1	335

Table 3: Information related to Annotation.

6 Proposed Solution

General architecture of the system is shown in the figure 1.

6.1 Preprocessing Steps

Before submitting the dataset for the classification task data needs to be preprocessed by applying the following steps:

- Extract email id and website : Email ids and website names present in the articles gives significant information about the article and its source. So they are extracted from the text before submitting it to further preprocessing steps.

- Word expansion : Some words like I'd, ain't, etc. are significantly present in the english text but when we submit this to our model it doesn't make any significant contribution to the classification task. So these words need to

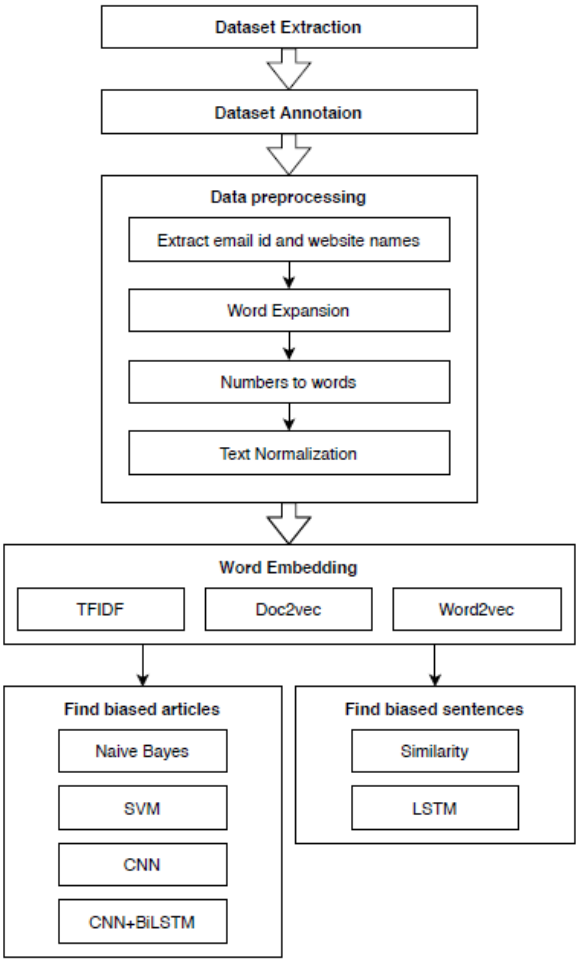

Figure 1: System Architecture

expanded to there corresponding forms like I would, am not.

- Numbers to words : Numbers present in the text is converted in to there words form as this will help classifier to understand the meaning properly. This was done using num2words library of python.

- Normalization : After that articles are converted into there normalized form by apply tokenization, lemmatization and changing the text into lower case. Along with this punctuation and stopwords were also removed.

6.2 Word embedding

Word embedding means to learn the vector representation of text such that similar words have similar embedding. Word embedding is used to generate the vector representation for words as well as the documents.

- TF-IDF : TF-IDF vector representation gives the numeric embedding to each sentence in the corpus, and each sentence is converted into a fixed dimension vector. As the numbers of words in the vocabulary may be large we are only considering the top 1000 features.

- Word2vec : Word2vec is used to create embedding for words based on the similarity of words. It is based on the two-layered neural network which is used to create embedding for the words without losing the context of words.

- Doc2vec : Doc2vec is based on the word2vec method described earlier. It is used to create vector embedding for the documents regardless of the length of the document. Along with the words, more information for the articles can also be included.

6.3 Models for finding the biased news articles

Here we have described various methods that we have applied to find the label for the document that is whether the documents will be marked as 1,0,-1. The models are shown in the increasing order of there accuracy.

6.3.1 Naive Bayes

Naive Bayes is one of the simplest models which can be used for the classification task. It tries to predict the class on the basis of the input, and then as per the maximum likelihood, out of much-classified output one is chosen. Here we have first converted the data into the TF-IDF vector form of max length 1000. Now submit this vector for the prediction task with encoded vector and encoded label for 1,0,-1. Prediction accuracy was lowest among all the model as it was not able to catch the context properly as it doesn't take the neighboring words into consideration.

6.3.2 Support vector machine (SVM)

As suggested in the baseline model we have applied SVM for our dataset with some hyperparameter tuning. Among all the available kernel, a polynomial kernel with degree 3 performs best for our dataset. Following parameter setting was used for SVM -

- Kernel : Polynomial, The reason for choosing this is shown in figure 2

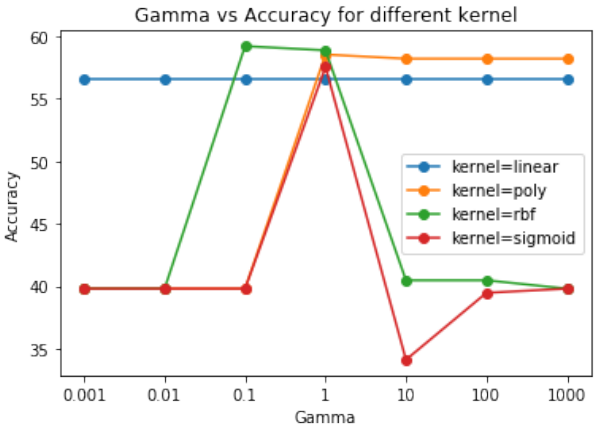

Figure 2: SVM Kernel Vs Accuracy

- Degree : 3, Degree of the polynomial hyper plane more value of degree may leads to over fitting.
- Parameter C : 0.01, small value of C ensures large margin in the training set
- Gamma : 10, High value of gamma makes model biased toward training data.

Even though SVM has an increased accuracy over Naive Bayes but still it is not able to capture the meaning of the words which is the primary requirement for our task.

6.3.3 Convolutional neural network (CNN)

Nowadays, CNN is frequently used for the task of text classification because CNN can capture the context in a better way as compared to normal neural networks or SVM.CNN's are useful in learning the patterns across the text, thus the context of the words is captured in a better way,and the meaning of the whole document can be captured.

Following steps were applied to use CNN for our task.(Pictorial representation is shown in figure 3) -

- Load the annotation from the dataset and convert it into categorical label. As we have 3 classes we will have length 3 vector for each annotation.
- After that convert each document to the vector form in which each word is represented by a number. Use padding (at the end) such that all the documents vectors are of the same length (1000 x 1).
- Now, pass this vector through the embedding layer which it's based on the word2vec concept and this will generate a vector of same size (1000 x 500). This embedding is passed through a dropout layer to avoid over-fitting.
- The output from the previous layer is passed through five different convolution layers with different filter sizes (2,3,4,5,6). Different filter size helps to capture the context of words from various windows sizes. The output from each layer is branch normalized to avoid over-fitting. The output from this layer is passed through a relu activation function layer.
- Now apply max pooling on each of this layer and concatenate them together. The output from this is passed through a dropout layer to avoid over-fitting.
- Pass this layer through a dense layer followed by another dense layer which generates the final output.
- Categorical cross-entropy is used as the loss measure and the neural net is optimized using the Adam optimizer.

6.3.4 CNN + Bidirectional Long Short Term Memory networks (BiLSTM)

Convolutional Neural Network (CNN) is good at extracting the features from the text. Bidirectional – Long Short-Term Memory (BiLSTM) is used for maintaining the chronological ordering of data. So, CNN+BiLSTM model has been used for finding the party to which the given news article is biased towards. Following are the steps performed for solving the above-mentioned problem,

- Pre-processing steps – tokenization, removing punctuation marks, removing stop words, lemmatization and converting number to words have been performed.
- Doc2vec is used for creating representations for different documents in the dataset. Two techniques available in Doc2vec – DBOW (Similar to CBOW from Word2Vec) and DM (Similar to Skip-gram) have been used. Representations obtained from both DBOW and DM are concatenated together and used for representing documents.

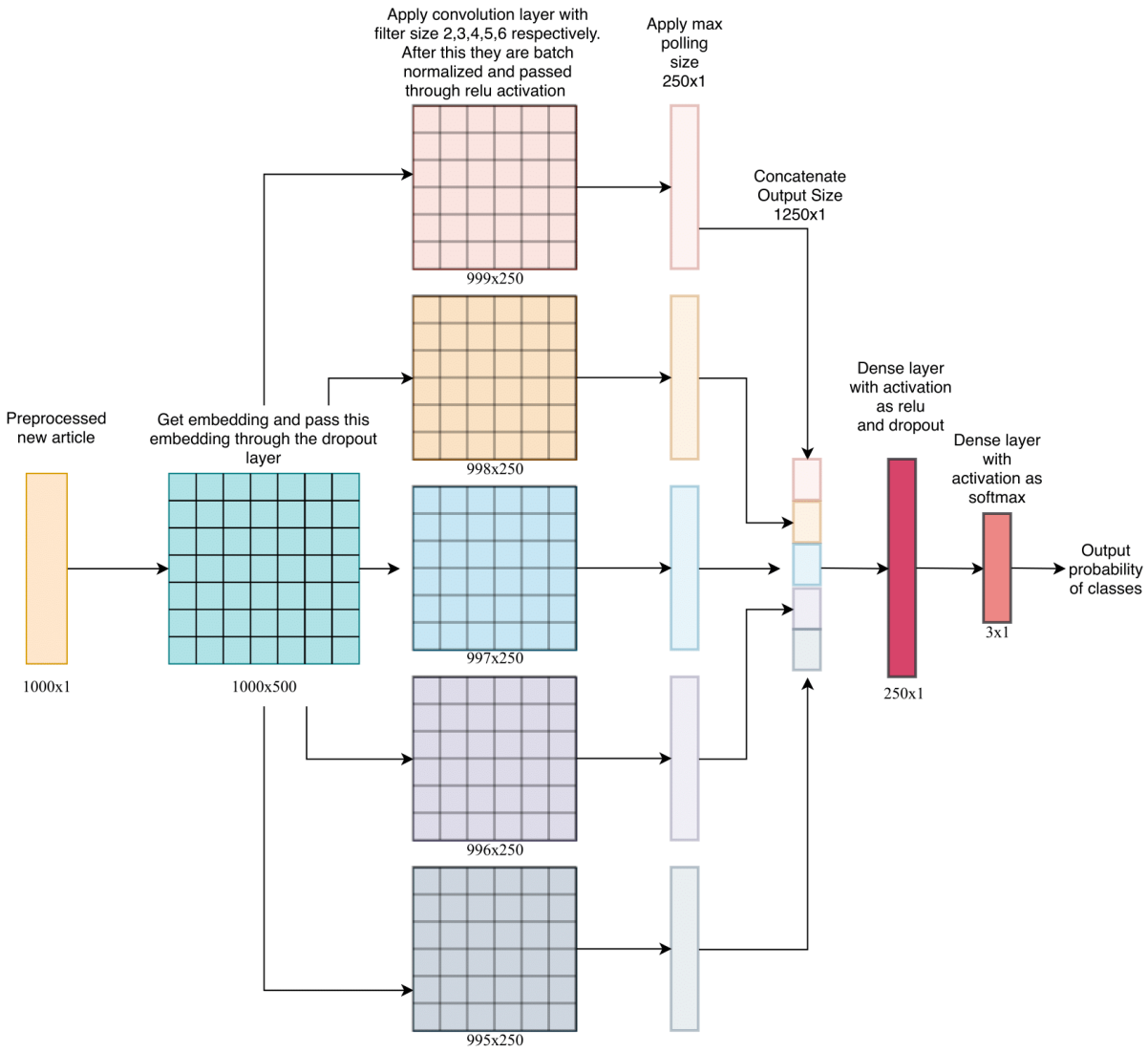

Figure 3: Model based on this Convolution Neural Network Architecture

- Convolutional layers with different filter sizes (2,3,4,5) are applied on document representation obtained using Doc2vec. After every convolutional layer, max-pooling layer with filter size – 2 is applied.
- Output of all max-pooling layers are concatenated together and passed as input to bi-directional LSTM layer. Bi-directional LSTM layer uses dropout activation function.
- Then, fully connected layer is used and finally softmax layer is connected which outputs the probability values indicating the chances of different classes being the answer for given document.
- Categorical cross-entropy loss function calculates the loss values which will be backpropagated into the neural network.
- Adam Optimizer is used as the optimization algorithm to update the parameter values during training.

Among all the approaches tried, CNN+BiLSTM approach achieve the best test accuracy because it acts as a hybrid approach using the advantages of both CNN and LSTM.

6.4 Models for finding the biased news sentences

This section talks about two approaches to find bias in the sentences. RNN was also trained but didn't provide the expected result. Finally result metrics for the better model is shown with its training loss.

6.4.1 Similarity based

The similarity between the sentences can be a good measure to see how closely one sentence is related to other. Taking this into consideration, the similarity between all the annotated sentences can be taken into account with the test sentence. To achieve this following steps were taken.

- Define a training dataset for which ground truth is available.
- Preprocess the data (Mentioned in the preprocessed steps section).
- For each test sample, i.e., each sample in the test data, compare its cosine similarity with all the preprocessed data.

$$Similarity(A, B) = \frac{A.B}{||A||.||B||}$$

- Give the label of train sentence for which this test sample gives the highest similarity score to the test sentence in question.

6.4.2 LSTM Based

This approach uses LSTM to predict the biasness of the sentence. The architecure of the model is fairly simple. It uses one LSTM layer of size 512 which takes the input vector of size 150. It has a dropout of 0.2. The output of this layer then goes to the dense layer with input size 3. It because we have 3 classes to predict from. The dense layer uses softmax function to predict the label for the vector.

Parameters:

- Loss: Categorical Cross Entropy
- Learning Rate: 0.001
- Optimizer: Adam
- Epochs: 100

Parameter tuning is done to choose these parameters. To generate the vector form of the sentences, Doc2Vec has been used. Vector size is defined to be 150. These vectors are inferred after performing the pre-processing steps on train set sentences. After pre-processing the train set sentences, Doc2Vec model has been trained. To get the vector representation of test sentences, the same model is used.

A graph has been plotted (7) showing the train loss for the LSTM model for 100 epochs.

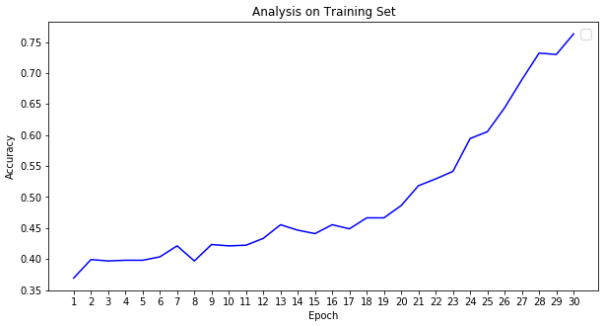

Figure 4: Training Accuracy of CNN-BiLSTM

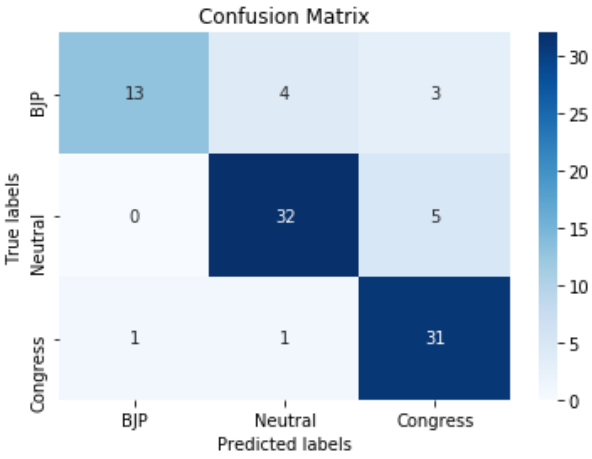

Figure 5: Confusion Matrix of CNN-BiLSTM

7 Results

7.1 Model for finding the biased news article

Following table shows the results obtained using different approaches. F1 score, precision and recall values reported are based on macro type. While micro type versions of F1 score, precision and recall are same as Accuracy values.

CNN performs better than Naive bayes and SVM models. Then in CNN+BiLSTM model, advantages of CNN and BiLSTM are both utilised which helped in achieving in good accuracy on test set. Training accuracy in different epochs and Confusion matrix of CNN-BiLSTM model are shown in Figure 4 and Figure 5.

7.2 Model for finding the biased news sentences

Similarity based model for finding sentence bias achieved an accuracy of 42.54% while LSTM model performs better with 58%. Table 5 shows the result obtained with LSTM. Precision, Recall and F1 Score have been shown. Along with this confusion matrix 6 is drawn to get a better idea

Model	Accuracy	F1 Score	Precision	Recall
Naive Bayes	53.84	42.34	49.21	49.36
SVM	57.12	52.75	56.47	53.00
CNN	64.67	61.71	61.65	64.88
CNN+BiLSTM	84.44	83.02	86.27	81.81

Table 4: Result for model used for finding the biased news article

Model	Precision	Recall	F1 score
LSTM	64	52	52

Table 5: Result for LSTM Model.

of how good the classifier is. From the matrix it is clear that the model has not been able to learn class 1 (Biased against Congress), this may be due to less number of samples for it. It is also clear that since most of the sentences are not biased, the model gives a good score for that.

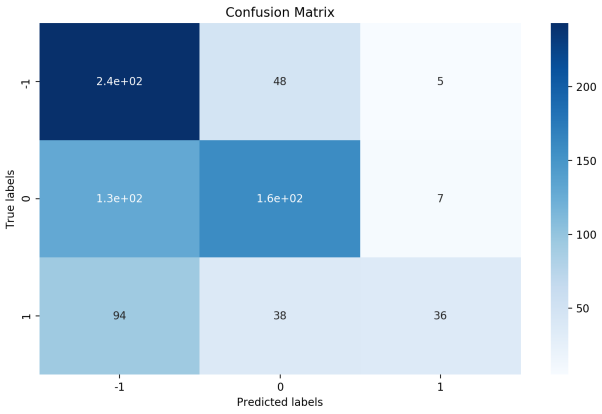

Figure 6: Confusion Matrix of LSTM (Sentence Bias)

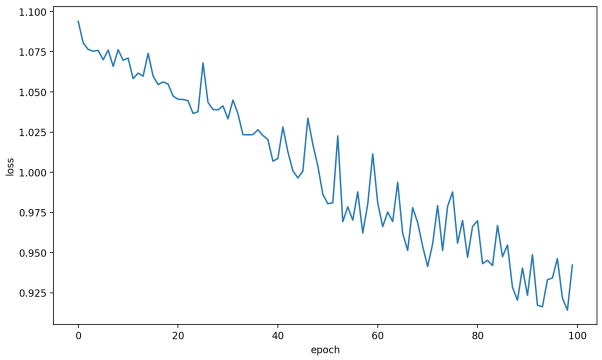

Figure 7: Training Loss for LSTM

8 Conclusion and Future work

CNN-BiLSTM model produced good results because it uses CNN's ability to extract the features and BiLSTM's ability to learn long range bi-directional dependencies in the text. Doc2vec based embedding vector representation is more suitable for representing long documents because it can represent the text in sentence/paragraph level.

In future work, same model can be tried with different languages, attention mechanism can be explored and also other word embeddings such as Glove and Fasttext can be tried.

To get better result for sentence bias part we can increase the number of annotations and have a balanced dataset for all the three classes.

References

Ramy Baly, Georgi Karadzhov, Dimitar Alexandrov, James R. Glass, and Preslav Nakov. 2018. Predicting factuality of reporting and bias of news media sources. *CoRR*, abs/1810.01765.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*.

Maryem Rhanoui, Mounia Mikram, Siham Yousfi, and Soukaina Barzali. 2019. A cnn-bilstm model for document-level sentiment analysis. *Machine Learning and Knowledge Extraction*, 1:832–847.