MEAN: Multi-head Entity Aware Attention Network for Political Perspective Detection in News Media

Chang Li
Department of Computer Science
Purdue University, West Lafayette, IN
li1873@purdue.edu

Dan Goldwasser
Department of Computer Science
Purdue University, West Lafayette, IN
dgoldwas@purdue.edu

Abstract

The way information is generated and disseminated has changed dramatically over the last decade. Identifying the political perspective shaping the way events are discussed in the media becomes more important due to the sharp increase in the number of news outlets and articles. Previous approaches usually only leverage linguistic information. However, news articles attempt to maintain credibility and seem impartial. Therefore, bias is introduced in subtle ways, usually by emphasizing different aspects of the story. In this paper, we propose a novel framework that considers entities mentioned in news articles and external knowledge about them, capturing the bias with respect to those entities. We explore different ways to inject entity information into the text model. Experiments show that our proposed framework achieves significant improvements over the standard text models, and is capable of identifying the difference in news narratives with different perspectives.

1 Introduction

The perspectives underlying the way information is conveyed to readers can prime them to take similar stances and shape their world view (Gentzkow and Shapiro, 2010, 2011). Given the highly polarized coverage of news events, recognizing these perspectives can help ensure that all point of view are represented by news aggregation services, and help avoid "information echo-chambers" in which only a single view point is represented.

Past work studying expression of bias in text has focused on lexical and syntactic representations of bias (Greene and Resnik, 2009; Recasens et al., 2013; Elfardy et al., 2015). Expressions of bias can include the use of the passive voice (e.g., *"mistakes were made"*), or references to known ideological talking points and framing decisions (Baumer et al., 2015; Budak et al., 2016; Card et al., 2016; Field et al., 2018; Morstatter et al., 2018) (e.g., *"pro-life"* vs. *"pro-choice"*). However, bias in news media is often more nuanced, expressed through informational choices (Fan et al., 2019), which highlight different aspects of the news story, depending on the *entity* or *relation* being discussed. For example, consider the following articles, discussing the same news story from different perspectives.

Adapted from **Huffington Post** (*Left*)

Rep. **Adam Schiff** (D-Calif.), one of the managers, pressed the case for additional witnesses, noting that **Trump** last month — in a video clip Schiff played senators — said he would "love" to have former administration officials testify in his Senate trial. "The Senate has an opportunity to take the president up on his offer to make his senior aides available",Schiff said. "But now the president is changing his tune."

Adapted from **Fox News** (*Right*)

House Intelligence Committee Chairman **Adam Schiff** of California, the leading House impeachment manager for Democrats, hurled the usual inflammatory accusations from his grab-bag of anti-**Trump** invectives ("corruption... cover-ups... misdeeds... lawlessness... guilt!") He tried to enliven his monotonous delivery with graphics, but the excessive words only added tedium to his largely laborious argument

Both stories describe the same set of events regarding the 2020 U.S Senate impeachment trial. In the top article, with a left leaning perspective, Rep. Schiff, leading the Democrats in the case, is quoted directly, while the bottom article, with a right leaning perspective, describes a negative reaction to his speech. Mapping the attitudes expressed in the text, to the appropriate right or left leaning perspective, requires extensive world knowledge about the identity of the people mentioned and their relationship, as well as the ability to associate relevant text with them. In the example above, recognizing that the negative sentiment words are associated with Rep. Schiff (rather than President Trump who is also mentioned in the article), and that he is associated with the left side of the political map, is the key to identifying the right leaning perspective of the article.

Proceedings of the 4th NLP4IF Workshop on NLP for Internet Freedom, pages 66–75

In this paper, we tackle this challenge and suggest an entity-centric approach to bias detection in news media. We follow the observation that expressions of bias often revolve around the main characters described in news stories, by associating them with different properties, highlighting their contribution to some events, while diminishing it, in others. To help account for the world knowledge needed to contextualize the actions and motives of entities mentioned in the news we train **entity** and **relation** (defined as a pair of entities in this paper) representations, incorporating information from external knowledge source and the news article dataset itself. We use the generalized term **aspect** to refer to either entity-specific or relation-specific view of the biased content in the article. We apply these representations in a **M**ulti-head **E**ntity **A**ware Attention **N**etwork (MEAN), which creates an entity-aware representation of the text.

We conducted our experiments over two datasets, Allsides (Li and Goldwasser, 2019) and SemEval Hyperpartisan news detection (Kiesel et al., 2019). We compared our approach to several competitive text classification models, and conducted a careful ablation study designed to evaluate the individual contribution of representing world knowledge using entity embedding, and creating the entity-aware text representation using multi-head attention. Our results demonstrate the importance of both aspects, each contributing to the model's performance.

2 Related Work

The problem of perspective identification is originally studied as a text classification task (Lin et al., 2006; Greene and Resnik, 2009; Iyyer et al., 2014), in which a classifier is trained to differentiate between specific perspectives. Other works use linguistic indicators of bias and expressions of implicit sentiment (Recasens et al., 2013; Baumer et al., 2015; Field et al., 2018).

Recent work by Fan et al., 2019 aims to characterize content relevant for bias detection. Unlike their work which relies on annotated spans of text, we aim to characterize this content without explicit supervision.

In the recent SemEval-2019, a hyperpartisan news article detection task was suggested[1]. Many works attempt to solve this problem with deep learning models (Jiang et al., 2019; Hanawa et al., 2019). We build on these works to help shape our text representation approach.

Several recent works also started to make use of concepts or entities appearing in text to get a better representation. Wang et al., 2017 treats the extracted concepts as pseudo words and appending them to the original word sequence which is then fed to a CNN. The KCNN (Wang et al., 2018) model, used for news recommendation, concatenates entity embeddings with the respective word embeddings at each word position to enhance the input. We take a different approach, and instead learn a document representation with respect to each entity in the article.

Using auxiliary information to improve text model was studied recently. Tang et al. proposes user-word composition vector model that modifies word embeddings given author representations in order to capture user-specific modification to word meanings. Other works incorporate user and product information to compute attentions over different semantic levels in the context of sentiment classification of online review (Chen et al., 2016; Wu et al., 2018). In this work, we learn the entity embedding based on external knowledge source (i.e. Wikipedia) or text, instead of including them in the training of bias prediction task. Therefore, we are able to capture rich knowledge about entities from various sources.

Another series of work that is closely related to ours is aspect based sentiment analysis. It aims at determining the sentiment polarity of a text span in a specific aspect or toward a target in the text. Many neural network based approaches have been proposed (Wang et al., 2016; Chen et al., 2017; Fan et al., 2018) to incorporate the aspect term into the text model. Recently, several works (Zeng et al., 2019; Song et al., 2019) designed their model based on BERT (Devlin et al., 2019). Unlike these works, we are not trying to determine the sentiment toward each entity mentioned in text. Instead, we are interested in identifying the underlying political perspective through the angles of these entities.

3 Model

The problem of political perspective detection in news media can be formalised as follows. Given a news article d, where d consists of sentences s_i, $i \in [1, L]$, and each sentence s_i consists of words w_{it}, $t \in [1, T]$. L and T are the number of sentences in d and number of words in s_i respectively. The goal of this task is to predict the political perspective

[1] https://pan.webis.de/semeval19/semeval19-web/

y of the document. Given different datasets, this can either be a binary classification task, where $y \in \{0, 1\}$ (hyperpartisan or not), or a multi-class classification problem, where $y \in \{0, 1, 2\}$ (left, center, right).

To inject knowledge about entities and relations, which would help solve the above classification problem, we first extract entities from the data corpus, and then learn knowledge representations for them using both external knowledge and the text corpus itself. In the second part, we describe how the learned aspect representations can be used in our Multi-head Entity Aware Attention Network. The overall architecture of our model is shown in Figure 1. It includes two sequence encoders, one for word level and another for sentence level. Our model learns a document representation with respect to each entity or relation in the document. The hidden states from an encoder are combined through a multi head entity-aware attention mechanism such that the generated sentence and document vectors will consider not only the context within the text but also the knowledge about the target entity (e.g. their political affiliation, or stance on controversial issues) or relation. We explain the acquisition of entity and relation knowledge representation and the structure of MEAN in details below.

3.1 Entity and Relation Knowledge Representation

We utilize the entity linking system DBpedia Spotlight (Daiber et al., 2013) to recognize and disambiguate the entities in news articles. We use the default configuration of DBpedia Spotlight, including the confidence threshold of 0.35, which helps to exclude uncertain or wrong entity annotations. We keep only entities with Person or Organization types that appear in the corpus. For each news article, we extract the top 5 entities (relations) based on number of mentions in the article as anchor aspects and learn a document representation with respect to each of them. The intuition is that the anchor aspects are the major figures and interactions discussed in a news article. By examining how each anchor aspect is discussed, our model can make better overall bias prediction. In this section, we introduce our pre-training models for learning entity and relation representations.

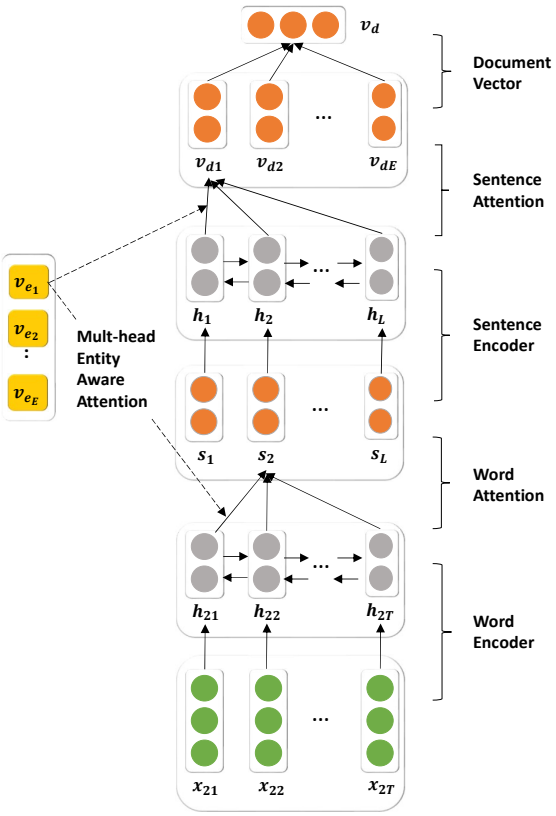

Figure 1: Overall Architecture of MEAN Model

3.1.1 Wikipedia Based Entity Representation

Wikipedia2Vec (Yamada et al., 2018) is a model that learns entity embeddings from Wikipedia. It learns embeddings of words and entities by iterating over the entire Wikipedia pages and maps similar words and entities close to one another in a continuous vector space. It jointly optimizes the representations by modeling entity-entity, word-word and entity-word relationships. We use entity representation from Wikipedia2Vec to initialize our entity embedding model in 3.1.2 which enables us to use the background knowledge of entities without training on a very large corpus.

3.1.2 Text Based Entity Representation

Inspired by the masked language modeling objective used in BERT (Devlin et al., 2019), we propose an entity-level masking task for learning meaningful representations of entities based on the news articles in which they are mentioned. The objective is to predict the masked entity based on the context provided by the other words in a sentence. Specifically, the entity mentions (regardless of number of tokens in text) are replaced with a special token "[MASK]" during preprocessing. We use a bidirectional LSTM to encode the sentence, and

the hidden state of the mask token will be used for prediction. It has the same structure as the sentence level encoder we describe in 3.2.2. We use negative sampling to randomly generate negative entity candidates from all possible entities uniformly. The learned entity representations can directly capture the context in the news articles they appear in.

3.1.3 Text Based Relation Representation

Similarly, we learn representation for an entity pair to encode the relationship between them. Given a sentence with two entity mentions masked, our model tries to predict the pair of entities. Again, a bidirectional LSTM with self attention is adopted to encode the sentence and the sentence representation are then used for prediction. We generate negative relation candidates from all possible relations uniformly.

3.2 Multi-Head Entity-Aware Attention Network

The basic component of our model is the Hierarchical LSTM model (Yang et al., 2016). The goal of our model is to learn document representation d_e with respect to an aspect e mentioned in it for bias prediction. In order to incorporate knowledge of aspects to better capture the nuance between news articles with different bias, we use aspect embeddings obtained in Section 3.1 to adjust the attention weight given to each sentence and word. It consists of several parts: a word sequence encoder, a word-level attention layer, a sentence sequence encoder and a sentence-level attention layer. The following sections describe the details of these components.

3.2.1 LSTM Networks

Long Short Term Memory networks (LSTMs) (Hochreiter and Schmidhuber, 1997) are a special kind of RNN, capable of learning long-term dependencies. Many recent works have demonstrated their ability to generate meaningful text representations. To capture the context in both directions, we use bidirectional LSTM in this work. For each element in the input sequence, the hidden state h is a concatenation of the forward hidden state $\overrightarrow{h}$ and backward hidden state $\overleftarrow{h}$ computed by the respective LSTM cells.

3.2.2 Hierarchical Aspect Attention

Word Sequence Encoder Given a sentence with words w_{it}, $t \in [1, T]$, each word is first converted to its embedding vector x_{it}. We can adopt pre-trained Glove (Pennington et al., 2014) word embeddings or deep contextualized word representation ELMo (Gardner et al., 2017) for this step. The word vectors are then fed into a word level bidirectional LSTM network to incorporate contextual information within the sentence. The hidden states h_{it} from the bidirectional LSTM network, are passed to the next layer.

Word Level Attention In (Yang et al., 2016), a self attention mechanism is introduced to identify words that are important to the meaning of the sentence, and therefore higher weights are given to them when forming the aggregated sentence vector. Actually the same words can also convey different meanings on distinct entities or relations. Following this intuition, we extend the idea by taking the aspect knowledge into account.

$$p_{itw} = tanh(W_w h_{it} + U_w v_e + b_w) \quad (1)$$

$$\alpha_{itw} = \frac{exp(p_{itw}^T p_w)}{\sum_t exp(p_{itw}^T p_w)} \quad (2)$$

$$s_{iw} = \sum_t \alpha_{itw} h_{it} \quad (3)$$

In addition to using the hidden states h_{it} alone to compute attention weight, we add the vector v_e for the anchor aspect e as another source of information. As a result, p_{itw} encode the importance of a specific word not only according to its context, but also the aspect of interest. p_{itw} is compared with the word level preference vector p_w to compute a similarity score, which is then normalized to get the attention weight α_{itw} through a softmax function. A weighted sum of the word hidden states are computed based on the attention weight as the sentence vector s_{iw}

Inspired by the multi-head attention scheme in (Vaswani et al., 2017), we propose a multi-head attention in our model to extend its ability to jointly attend to information at different positions. The sentence vector s_i is computed as an average of s_{iw} obtained from different attention heads. Note that we learn a separate copy of the parameters W_w, U_w, b_w and p_w for each attention head.

$$s_i = \frac{\sum_w s_{iw}}{NH_W} \quad (4)$$

NH_W is the number of word-level attention head.

Sentence Sequence Encoder and Sentence Level Attention Given the sentence vectors s_i, $i \in [1, L]$, we can generate the document vector in a similar way. Hidden states h_i together with the aspect embedding v_e are used to compute the attention weight for each sentence. After that, the document vector v_{des} is obtained as a weighted average of hidden states h_i. v_{des} obtained from different attention heads are averaged to generate aspect oriented document representation v_{de}.

$$v_{de} = \frac{\sum_s v_{des}}{NH_S} \tag{5}$$

where NH_S is the number of attention heads at sentence level.

Document Classification The document representations v_{de} with respect to aspect e captures the bias related information in news article d from the angle of aspect e. They can be used as features for predicting the document bias label.

$$p_{de} = \text{softmax}(W_c v_{de} + b_c) \tag{6}$$

We use the negative log likelihood of the correct labels as classification training loss:

$$L = -\sum_d \sum_{e \in E_d} \log p_{dej} \tag{7}$$

where E_d is the set of aspects mentioned in news article d, and j is the bias label of d.

Note that we use the bias label for the entire news article d as label for each aspect oriented document representation v_{de} during training. This is not ideal as the narratives about some aspects in the article may not be consistent with the overall political perspective. But it is a reasonable approximation given the labels for aspect oriented document representations are expensive to obtain. At test time, we use average pooling to get the aggregated document representation v_d which combine the political perspective targeting each aspect of interest.

$$v_d = \frac{\sum_e v_{de}}{|E_d|} \tag{8}$$

Given the entity and relation representations are not in the same space, we use them to train separate models. We regard the MEAN model using entity embedding and relation embedding for attention as **MEAN_ENT** and **MEAN_REL** respectively. We also explore a simple ensemble **MEAN_Ensemble**, which makes prediction based on the sum of probability scores p_{de} from the above two models at test time. Note that this does not require retraining.

4 Experiments

4.1 Datasets and Evaluation

We run experiments on two news article datasets: Allsides and SemEval. The statistics of both datasets is shown in Table 1.

Allsides This dataset (Li and Goldwasser, 2019) is collected from two news aggregation websites[2] on 2020 different events discussing 94 event types. The websites provide news coverage from multiple perspectives, indicating the bias of each article using crowdsourced and editorial reviewed approaches. Each article has a bias label left, center or right. We used the same randomly separated splits for evaluation in this paper so that our results are directly comparable with previous ones.

SemEval This is the official training dataset from SemEval 2019 Task 4: Hyperpartisan News Detection (Kiesel et al., 2019). The task is to decide whether a given news article follows a hyperpartisan argumentation. There are 645 articles in this dataset and each is labelled manually with a binary label to indicate whether it is hyperpartisan or not. Since the test set is not available at this time, we conducted 10-fold cross validation on the training set with exactly the same splits as in (Jiang et al., 2019) so that we can compare with the system that ranked in the first place.

Dataset	Center	Left	Right	Avg # Sent.	Avg # Words
Allsides	4164	3931	2290	49.96	1040.05
		Hyperpartisan			
SemEval	407	238		27.11	494.29

Table 1: Datasets Statistics

4.2 Baselines

We compare our model with several baseline methods, including traditional approaches that utilize textual information alone, and other strategies to utilize knowledge of entities.

[2] Allsides.com and Memeorandum.com

4.2.1 Methods using only textual information

SkipThought regard each document as a long sentence, and map it to a 4800-dimension vector with the sentence level encoder Skip-Thought (Kiros et al., 2015).

HLSTM first tokenize a document into sentences, then each sentence was tokenized into words. A word-level and a sentence-level bidirectional LSTM are used to construct a vector representation for each sentence and then the document. Self attention is used to aggregate hidden states at both word and sentence levels.

BERT is a language representation model based on deep bidirectional Transformer architectures (Vaswani et al., 2017). It was pre-trained with masked language model and next sentence prediction tasks on huge corpus. As a result, it can achieve state-of-the-art results on a wide range of tasks by fine-tuning with just one additional output layer.

CNN_Glove (CNN_ELMo) is the model from the team that ranked first in hyperpartisan news detection task in SemEval 2019 (Jiang et al., 2019). It uses the pre-trained Glove (ELMo) word vectors, which is then averaged as sentence vectors. The sentences vectors are fed into 5 convolutional layers of different kernal sizes. The outputs for all convolution layers are concatenated for prediction.

4.2.2 Methods using entity information

Models listed below have the same architecture with MEAN, including multi-head self attention. The only difference is how and where entity information is used.

HLSTM_Embed concatenate the entity embedding with word embedding at each position such that the new input to word level LSTM $x'_{it} = [x_{it}; v_e]$ where ; is the concatenation operator. This model has the potential to bias the political preference of a word. This is because a word can be associated with bias when describing one entity while neutral when describing others.

HLSTM_Output concatenate the entity embedding with the document vector v_d generated by HLSTM such that $v'_d = [v_d; v_e]$. This means that we bias the probability distribution of political bias based on the final document encoding. If an entity is usually associated with one bias in certain topics, then this model would be able to capture that.

4.3 Implementation Details

We use the spaCy toolkit for preprocessing the documents. All models are implemented with PyTorch (Paszke et al., 2017)[3]. The 300d Glove word vectors (Pennington et al., 2014) trained on 6 billion tokens are used to convert words to word embeddings. They are not updated during training. The sizes of LSTM hidden states for both word level and sentence level are 300 for both Allsides and SemEval dataset. The number of attention head at both word and sentence levels are set to 4 for Allsides, while only one head is used for SemEval due to the limited data size. For the training of the neural network, we used the Adam optimizer (Kingma and Ba, 2014) to update the parameters. On Allsides dataset, 5% of the training data is used as the validation set. We perform early stopping using the validation set. However, same as (Jiang et al., 2019), we use the evaluation part of each fold for early stopping and model selection. The learning rate lr is set to 0.001 for all models except BERT for which $2e-5$ is used. The mini-batch size is $b = 10$ for all models except for relation attention models which can only set $b = 8$ due to the size of GPU memory.

4.4 Results

4.4.1 Results on Allsides

We report the micro F1 and macro F1 scores on test set for Allsides dataset in Table 4. The results are divided into two groups based on whether contextualized word representations are used. In the first group, we have the results of models using only textual information, which are reported in (Li and Goldwasser, 2019). Although baseline models using entity information significantly outperform the HLSTM baseline, they are no better than our MEAN model, indicating these two strategies of using entity embedding is not optimal. Our MEAN model achieves the best result in terms of both micro and macro F1 scores no matter whether contextualized word embeddings are used or not. This demonstrates our model can use knowledge encoded in entity embedding as additional context to identify bias expressed in more subtle ways. Therefore it generates high-quality document representation for political perspective prediction. The gaps between our model and baselines decrease when contextualized word representations are used

[3]Please refer to https://github.com/BillMcGrady/MEAN for data and source code

since local context is better captured in this setting. We also observe our MEAN_REL models is not as good as MEAN_ENT models. This is expected since we do not have good initialization for the relation representations. However, it is worth noting that performance of our framework further improves by using ensemble of our MEAN_ENT and MEAN_REL models for prediction. This demonstrates that the relation embedding learned does encode some additional signal for the task.

Model	Micro F1	Macro F1
SkipThought †	68.67	-
HLSTM †	74.59	-
HLSTM_Embed	76.45	74.95
HLSTM_Output	76.66	75.39
MEAN_Glove_ENT	78.22	77.19
MEAN_Glove_REL	77.85	76.70
MEAN_Glove_Ensemble	**80.56**	**79.62**
HLSTM_ELMo	80.11	79.02
BERT	79.58	77.91
MEAN_ELMo_ENT	80.87	80.00
MEAN_ELMo_REL	79.25	77.93
MEAN_ELMo_Ensemble	**82.32**	**81.30**

Table 2: Test Results on Allsides Dataset. † indicates results reported in (Li and Goldwasser, 2019).

4.4.2 Results on SemEval

The performance of various models on SemEval dataset can be found in Table 3. Again the results are grouped based on word representation used. CNN_Glove and CNN_ELMo are results reported by the winning team in the SemEval competition. They proposed an ensemble of multiple CNN models. Still, our model outperforms the winning team, showing the advantages of representing text with respect to different aspects. The other trends hold as well in SemEval dataset although the margin is smaller comparing to Allsides. This is partially due to the limited size of this dataset. Again, although MEAN_REL does not outperform baselines themselves, it helps to achieve the best accuracy score when combined with MEAN_ENT.

4.4.3 Ablation Study

We show the results for ablations of our MEAN_Glove_ENT model. The performance drops slightly when removing entity embedding at attention computation or not using multi-head attention. If both entity embedding and multi-head attention are removed, there is a dramatic decrease in performance, signaling these two modules complement each other in this task. Note that when both entity embedding and multi-head attention are not used, our model is equivalent to HLSTM. We attribute the difference in performance between our result and that reported in (Li and Goldwasser, 2019) to random initialization and hyper-parameters setting.

Model	Accuracy
CNN_Glove ‡	79.63
HLSTM	81.58
HLSTM_Embed	81.71
HLSTM_Output	81.25
MEAN_Glove_ENT	82.65
MEAN_Glove_REL	80.78
MEAN_Glove_Ensemble	**83.12**
CNN_ELMo ‡	84.04
HLSTM_ELMo	83.28
BERT	83.41
MEAN_ELMo_ENT	84.51
MEAN_ELMo_REL	83.09
MEAN_ELMo_Ensemble	**85.22**

Table 3: Test Results on SemEval Dataset. ‡ indicates results reported in (Jiang et al., 2019). Our full model outperforms the system ranked first in SemEval-2019 Hyperpartisan News Detection Task.

Model	Micro F1	Macro F1
MEAN_Glove_ENT	78.22	77.19
w/o Entity Embedding	76.69	75.03
w/o Multi-head attention	77.82	76.42
w/o Both	73.99	72.47

Table 4: Ablation Study on Allsides Dataset.

4.4.4 Qualitative Results

Sentiment Lemmas for Entities and Relations To better understand the effectiveness and meaning of the learnt attention scores, we find the most attended to sentiment lemmas in Allsides dataset with respect to a certain entity or relation. We calculate the attention given to a token x_{it} by an article as the sentence attention multiplied by word attention in the sentence $\alpha_{x_{it}} = \alpha_i * \alpha_{it}$. We average the attention given by multiple heads in this evaluation. To aggregate information better, we lemmatized all tokens.

Given the lemma attention definition above, we can compute the attention scores of a lemma across the dataset with respect to an aspect by averaging the attention score of each occurrence of that lemma. We extract lemmas with most attention and filter out neutral ones using the VADER sentiment lexicon (Hutto and Gilbert, 2014). We present top five lemmas for some prominent entities and relations between them from Democratic Party and Republican Party in Table 5. The phrases are selected from articles with left or right bias. There are several interesting findings from the table:

1. The top lemmas from left and right articles

Entities/Relations	Left Articles	Right Articles
Barack Obama	admire, motivate, blame, murder, amaze	blame, terrorist, admire, like, love
Hillary Clinton	brutal, admire, disgusting, promote, disturbing	super, lie, hack, destroy, defeat
Bernie Sanders	rig, destroy, kill, accuse, dedicated	rig, fire, help, win, clear
Donald Trump	ugly, fascinate, mislead, damn, scary	special, bizarre, suspect, loyal, super
Mitch McConnell	special, accuse, argue, regret, criticize	like, illegal, best, promised, clear
Mitt Romney	illegal, support, entitle, create, interest	accuse, illegal, great, support, argue
Donald Trump - Hillary Clinton	insult, dam, honest, horrible, amaze	hack, positive, warn, great, kill
Donald Trump - Mitch McConnell	condemn, respect, scream, tick, love	respect, wish, bright, like, happy
Hillary Clinton - Barack Obama	relax, respect, benefit, enjoy, compliment	innocent, hope, hate, great, super
Hillary Clinton - Bernie Sanders	complain, insult, promote, mourn, enjoy	destroy, merry, excite, cheat, wrong

Table 5: Top Sentiment Lemmas with Most Attention Scores

Sentence with Attention	Human Annotation	Entity
President Donald Trump announced Friday a short - term plan that will reopen the government for three weeks so that border security negotiations may continue without the devastating effects of the partial government shutdown .	devastating	Donald Trump
Netanyahu , who has a famously frosty relationship with President Obama , mentions neither Obama nor Republican challenger Mitt Romney , with whom Netanyahu worked in the mid-1970s at Boston Consulting Group .	famously frosty	Barack Obama
In the last few weeks , the fight turned particularly nasty – with Trump canceling a Democratic congressional trip to Afghanistan after House Speaker Nancy Pelosi called on Trump to delay his State of the Union address or submit it in writing .	Whole Sentence	Nancy Pelosi
However , Democrats rejected the plan even before Trump announced it , and a Senate version of the plan failed to get the 60 votes needed on Thursday .	However, . . . announced it	Democratic Party

Table 6: Comparison between Model Attention and Human Annotation

show different sentiment sometimes but not always. One cause of this is we do not know how a sentiment word is used with only unigrams. The bias would be totally different when someone is blamed or blame others for an event.

2. Different entities may pay attention to the same lemma since attention in our setting encodes "relatedness to bias prediction" instead of "association to a specific bias". For example, the lemma "illegal", which may refer to the illegal immigrants issue, receives high attention score with respect to both Mitch McConnell and Mitt Romney, indicating the opinion expressed toward this topic can reflect the bias of an article.

3. For relations, the sentiment lemmas reflect bias. For rivals from different party (e.g. Donald Trump and Hillary Clinton), the negative sentiment dominates in both sides. However, the depiction of relationship differs for both sides for allies. (e.g. Donald Trump and Mitch McConnell).

Human Annotation Comparison The BASIL dataset (Fan et al., 2019) has human annotation of bias spans. It contains 300 articles on 100 events with 1727 bias spans annotated. On the sentence level, spans of lexical and informational bias are identified by annotators by analyzing whether the text tends to affect a reader's feeling towards one of the main entities. We show example sentences with attention assigned by our model and human annotated bias span in Table 6.

5 Conclusion

In this work, we propose an entity-centric framework for political perspective detection. Entity and relation representations learnt from external knowledge source and text corpus are utilized to compute attention at both word and sentence levels. A document representation with respect to each aspect in the article is then generated for prediction. Empirical experiments on two recent news article datasets show that our model achieve significantly better performance in bias detection comparing to traditional text models and other strategies of incorporating entity information.

In fact, relations are highly dependent on individual entities. We intend to extend this work to learn better relation representations given entity embeddings based on description of entity interactions in text. Moreover, we would like to weigh the importance of each aspect toward the overall perspective of an article instead of having all of them contribute equally to the final prediction.

References

E. Baumer, E. Elovic, Y. Qin, F. Polletta, and G. Gay. 2015. Testing and comparing computational approaches for identifying the language of framing in political news. In *NAACL*, pages 1472–1482, Denver, Colorado. Association for Computational Linguistics.

C. Budak, S. Goel, and J. M. Rao. 2016. Fair and balanced? quantifying media bias through crowdsourced content analysis. *Public Opinion Quarterly*, 80(S1):250–271.

Dallas Card, Justin Gross, Amber Boydstun, and Noah A. Smith. 2016. Analyzing framing through the casts of characters in the news. In *EMNLP*, pages 1410–1420, Austin, Texas. Association for Computational Linguistics.

Huimin Chen, Maosong Sun, Cunchao Tu, Yankai Lin, and Zhiyuan Liu. 2016. Neural sentiment classification with user and product attention. In *EMNLP*, pages 1650–1659, Austin, Texas. Association for Computational Linguistics.

Peng Chen, Zhongqian Sun, Lidong Bing, and Wei Yang. 2017. Recurrent attention network on memory for aspect sentiment analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 452–461, Copenhagen, Denmark. Association for Computational Linguistics.

J. Daiber, M. Jakob, C. Hokamp, and P. N. Mendes. 2013. Improving efficiency and accuracy in multilingual entity extraction. In *I-SEMANTICS*, I-SEMANTICS '13, pages 121–124, New York, NY, USA. ACM.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Heba Elfardy, Mona Diab, and Chris Callison-Burch. 2015. Ideological perspective detection using semantic features. In *STARSEM*, pages 137–146, Denver, Colorado. Association for Computational Linguistics.

Feifan Fan, Yansong Feng, and Dongyan Zhao. 2018. Multi-grained attention network for aspect-level sentiment classification. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3433–3442, Brussels, Belgium. Association for Computational Linguistics.

L. Fan, M. White, E. Sharma, R. Su, P. K. Choubey, R. Huang, and L. Wang. 2019. Media bias through the lens of factual reporting. In *EMNLP*.

A. Field, D. Kliger, S. Wintner, J. Pan, D. Jurafsky, and Y. Tsvetkov. 2018. Framing and agenda-setting in Russian news: a computational analysis of intricate political strategies. In *EMNLP*, pages 3570–3580, Brussels, Belgium. Association for Computational Linguistics.

M. Gardner, J. Grus, M. Neuman, O. Tafjord, P. Dasigi, N. F. Liu, M. Peters, M. Schmitz, and L.S. Zettlemoyer. 2017. Allennlp: A deep semantic natural language processing platform.

Matthew Gentzkow and Jesse M Shapiro. 2010. What drives media slant? evidence from us daily newspapers. *Econometrica*, 78(1):35–71.

Matthew Gentzkow and Jesse M Shapiro. 2011. Ideological segregation online and offline. *The Quarterly Journal of Economics*, 126(4):1799–1839.

Stephan Greene and Philip Resnik. 2009. More than words: Syntactic packaging and implicit sentiment. In *NAACL-HLT*, pages 503–511, Boulder, Colorado. Association for Computational Linguistics.

K. Hanawa, S. Sasaki, H. Ouchi, J. Suzuki, and K. Inui. 2019. The sally smedley hyperpartisan news detector at SemEval-2019 task 4. In *SemEval*, pages 1057–1061, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

S. Hochreiter and J. Schmidhuber. 1997. Long short-term memory. *Neural Comp.*, 9(8):1735–1780.

C. Hutto and E. Gilbert. 2014. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *ICWSM*.

Mohit Iyyer, Peter Enns, Jordan Boyd-Graber, and Philip Resnik. 2014. Political ideology detection using recursive neural networks. In *ACL*, pages 1113–1122, Baltimore, Maryland. Association for Computational Linguistics.

Ye Jiang, Johann Petrak, Xingyi Song, Kalina Bontcheva, and Diana Maynard. 2019. Hyperpartisan news detection using ELMo sentence representation convolutional network. In *SemEval*, pages 840–844, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

J. Kiesel, M. Mestre, R. Shukla, E. Vincent, P. Adineh, D. Corney, B. Stein, and M. Potthast. 2019. SemEval-2019 task 4:hyperpartisan news detection. pages 829–839, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

R. Kiros, Y. Zhu, R R Salakhutdinov, R. Zemel, R. Urtasun, A. Torralba, and S. Fidler. 2015. Skip-thought vectors. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *NIPS*, pages 3294–3302. Curran Associates, Inc.

C. Li and D. Goldwasser. 2019. Encoding social information with gcn for political perspective detection in news media. In *ACL*, pages 2594–2604, Florence, Italy. Association for Computational Linguistics.

W-H Lin, T. Wilson, J. Wiebe, and A. Hauptmann. 2006. Identifying perspectives at the document and sentence levels. In *CoNLL*, CoNLL-X '06, pages 109–116, Stroudsburg, PA, USA. Association for Computational Linguistics.

F. Morstatter, L. Wu, U. Yavanoglu, S. R. Corman, and H. Liu. 2018. Identifying framing bias in online news. *Trans. Soc. Comput.*, 1(2):5:1–5:18.

A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer. 2017. Automatic differentiation in PyTorch. In *NIPS Autodiff Workshop*.

J. Pennington, R. Socher, and C. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

M. Recasens, C. Danescu-Niculescu-Mizil, and D. Jurafsky. 2013. Linguistic models for analyzing and detecting biased language. In *ACL*, pages 1650–1659, Sofia, Bulgaria. Association for Computational Linguistics.

Youwei Song, Jiahai Wang, Tao Jiang, Zhiyue Liu, and Yanghui Rao. 2019. Attentional encoder network for targeted sentiment classification. *arXiv preprint arXiv:1902.09314*.

Duyu Tang, Bing Qin, Ting Liu, and Yuekui Yang. 2015. User modeling with neural network for review rating prediction. In *IJCAI*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Hongwei Wang, Fuzheng Zhang, Xing Xie, and Minyi Guo. 2018. Dkn: Deep knowledge-aware network for news recommendation. In *WWW*, WWW' 18, pages 1835–1844, Republic and Canton of Geneva, Switzerland. International World Wide Web Conferences Steering Committee.

J. Wang, Z. Wang, D. Zhang, and J. Yan. 2017. Combining knowledge with deep convolutional neural networks for short text classification. In *IJCAI*, IJCAI, pages 2915–2921. AAAI Press.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based LSTM for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615, Austin, Texas. Association for Computational Linguistics.

Z. Wu, X.Y Dai, C. Yin, S. Huang, and J. Chen. 2018. Improving review representations with user attention and product attention for sentiment classification. In *AAAI*.

I. Yamada, A. Asai, H. Shindo, H. Takeda, and Y. Takefuji. 2018. Wikipedia2vec: An optimized tool for learning embeddings of words and entities from wikipedia. *arXiv*.

Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California. Association for Computational Linguistics.

Biqing Zeng, Heng Yang, Ruyang Xu, Wu Zhou, and Xuli Han. 2019. Lcf: A local context focus mechanism for aspect-based sentiment classification. *Applied Sciences*, 9:3389.