Cross-Topic Rumor Detection using Topic-Mixtures

Xiaoying Ren, Jing Jiang*
Singapore Management University
80 Stamford Road
Singapore 178902
lemontreejying@gmail.com
jingjiang@smu.edu.sg

Ling Min Serena Khoo, Hai Leong Chieu
DSO National Laboratories
12 Science Park Drive
Singapore 118225
{klingmin,chaileon}@dso.org.sg

Abstract

There has been much interest in rumor detection using deep learning models in recent years. A well-known limitation of deep learning models is that they tend to learn superficial patterns, which restricts their generalization ability. We find that this is also true for cross-topic rumor detection. In this paper, we propose a method inspired by the "mixture of experts" paradigm. We assume that the prediction of the rumor class label given an instance is dependent on the topic distribution of the instance. After deriving a vector representation for each topic, given an instance, we derive a "topic mixture" vector for the instance based on its topic distribution. This topic mixture is combined with the vector representation of the instance itself to make rumor predictions. Our experiments show that our proposed method can outperform two baseline debiasing methods in a cross-topic setting. In a synthetic setting when we removed topic-specific words, our method also works better than the baselines, showing that our method does not rely on superficial features.

1 Introduction

Recently, there has been much interest in detecting online false information such as rumors and fake news. Existing work has explored different features including network structures (Ma et al., 2019a), propagation paths (Liu and Wu, 2018), user credibility (Castillo et al., 2011) and the fusion of heterogeneous data such as image and text (Wang et al., 2018). However, these proposed algorithms still cannot be easily deployed for real-world applications, and one of the key reasons is that, just like many other NLP problems, rumor or fake news detection models may easily overfit the training data and thus cannot perform well on new data. The problem can be more serious with deep learning solutions, because deep neural networks tend to learn superficial patterns that are specific to the training data but do not always generalize well (Wang et al., 2018).

In this work, we study the task of rumor detection and focus on the problem of adapting a rumor detection model trained on a set of *source* topics to a *target* topic, which we refer to as *cross-topic rumor detection*. In a recent study by Khoo et al. (2020), the authors compared the performance of rumor detection in an in-topic setting and an out-of-topic setting. They found that their model could achieve 77.4% macro F-score on the in-topic testing data but the performance of the same classifier dropped to 39.5% when applied to out-of-topic testing data, which describe events different from the training events.

In this paper, we propose a method inspired by the "mixture of experts" paradigm, abbreviated as "MOE". Understanding that the rumor prediction model may work differently for different topics, we assume that the prediction result on an instance is dependent on the topic distribution of that instance. While a standard method is to train topic-specific classifiers and then use the topic distribution to combine these topic-specific classifiers, we propose a different approach where the topic distribution is used to linearly combine a set of vectors representing different topics. This gives us a "topic-mixture" given an example. This topic-mixture vector of the example is concatenated with the vector representation of the example itself and used as the input to a neural network model for rumor label prediction.

We implement our method on top of a state-of-the-art StA-HiTPLAN model and conduct experiments using the PHEME dataset. Compared with two baseline methods that also perform debiasing, we find that our method can achieve clearly better cross-topic performance. We also experiment with

*Corresponding author

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics, pages 1534–1538
April 19 - 23, 2021.

modified within-topic data where we intentionally remove topic-specific words. This creates a setting where it is hard for models to rely on topic-specific words to make rumor predictions. We find that our method can also outperform the baselines substantially.

2 Performance Degradation in Cross-Topic Rumur Detection

In this section, we present a case study on the PHEME dataset to quantify the degree of overfitting of an existing model by analyzing the influence of topic-specific words.

Concretely, we use the PHEME dataset, which has five topics. We use four topics during training and the remaining one for out-of-domain testing. After obtaining a trained hierarchical transformer model (Vaswani et al., 2017), we perform post-hoc testing by applying it to different topics, with K topic-specific words masked to examine the performance drop. Here the topic-specific words are identified based on log-odds ratio with Dirichlet prior (Monroe et al., 2008), and we regard these topic-specific words as possible spurious patterns. It is a common way to identify words that are statistically over-represented in a particular population compared to others. For the in-domain testing, we split the data as 7:2:1 for training, testing and validation. Experiments are performed using $K \in \{20, 50, 100, 200\}$.

Results: The partial results are shown in Table 1. It is noteworthy that the accuracy drops from 67.69% to 36.7% when we only mask the top-20 frequent event-aware words in in-domain set - the model is highly sensitive to event sensitive patterns. Besides, the little dropping in accuracy with the out-of-domain setting when we mask top-20 out-of-domain words may indicate that we mask some training unseen words compared with non-mask setting. These experiments confirm our hypothesis that the baseline classifier is primarily learning topical correlations, and motivate the need for a debiased classification approach which we will describe next.

tA-HiTPLAN	in-domain	out-of-domain
{non-mask}	0.6769	0.3441
{+MASK TOP 20}	0.3670	0.3425
{+MASK TOP 50}	0.3526	0.3255
{+MASK TOP 100}	0.3413	0.3122
{+MASK TOP 200}	0.3202	0.2903

Table 1: Accuracy on PHEME event-5 datset.

3 Method

3.1 Notation

Let x be an input, which is a thread represented as a sequence of tokens. We assume that x consists of a sequence of posts $x = x_1, x_2, \ldots, x_T$ chronologically ordered, in which x_1 represents a source post and x_i $(i > 1)$ represents a reply post. Let y be the rumor label (e.g., true rumor, false rumor, etc.) we want to predict. We assume that the training data come from a set of M different topics, and we use $\{\mathcal{S}_i\}_{i=1}^{M}$ to denote the data, where $\mathcal{S}_i \triangleq \{(x_n^i, y_n^i)\}_{n=1}^{|\mathcal{S}_i|}$. Our goal is to train a rumor detection classifier using the labeled data from the M topics such that the classifier can work well on a target example.

3.2 Mixture Of Experts

Our idea is inspired by Mixture of Experts models (Jacobs et al., 1991). Specifically, we assume that each example $\mathbf{x}$ has a distribution over the M training topics. Let t be a variable denoting topic. We model $p(y|x)$ as follows:

$$p(y|x) = \sum_{i=1}^{M} p(t=i|x)p(y|x,t=i). \quad (1)$$

Normally, to model $p(t|x)$ and $p(y|x,t)$, we can train parameterized models $p(t|x;\theta_1)$ and $p(y|x,t;\theta_2)$ using our training data, because our examples have clear topic labels. However, if the number of topics is large, or the number of training instances for each topic is small, training such topic-specific models may not work well. Moreover, if we train independent models for each training topic and combine their out-of-domain testing result as a whole, the result may be unsatisfactory because each model may be overfitting a specific topic. Our initial experimental observation also verifies that independent training method works well on in-topic setting but does not perform well on out-of-topic setting. Here we explore an alternative approach as described below.

We assume that x and t are both represented as vectors (which we will explain later). We can then use the following neural network model to model $p(y|x,t)$:

$$\begin{aligned} p(y|x,t) &= p(y|\mathbf{x} \oplus \mathbf{t}; \theta) \\ &= \frac{\exp(\theta_y \cdot (\mathbf{x} \oplus \mathbf{t}))}{\sum_{y'} \exp(\theta_{y'} \cdot (\mathbf{x} \oplus \mathbf{t}))}, \end{aligned}$$

where θ_y are vectors to be learned and $\oplus$ means vector concatenation.

Now we can make an approximation of Eqn. (1) as follows:

$$\begin{aligned} p(y|x) &= \sum_{i=1}^{M} p(y|x,t)p(t|x) \\ &= p(y|\mathbf{x} \oplus \sum_{i=1}^{M} p(t=i|x)\mathbf{t}_i;\theta), \quad (2) \end{aligned}$$

where $\mathbf{t}_i$ is a vector representation of topic i. We can see that instead of computing $p(y|x,t=i)$ for each i, and then use $p(t=i|x)$ to obtain a weighted sum of these $p(y|x,t=i)$, we first get a sum of the vector representations of different topics weighted by $p(t|x)$, and then use this weighted sum to compute $p(y|x)$.

To obtain a vector representation of x, we can use BERT to process the sequence of tokens in x and then use the vector representing the `[CLS]` token at the top layer as $\mathbf{x}$. For each topic t, since we have instances of x belonging to each topic, here we explore two ways of deriving $\mathbf{t}_i$ for topic i: (1) We use the average of the vectors $\mathbf{x}$ belonging to topic i to form $\mathbf{t}_i$. We refer to this as **Avg**. (2) We use the parameters at the top layer of the topic classification model $p(t|x)$ as vector representations for the different topics. We refer to this as **Param**. During test time, since our instance x does not have a t associated with it, we use a topic classification model trained on the training data where each example has its correct topic labeled to estimate $p(t|x)$.

4 Experiments

4.1 Implementation Details

We follow the model architecture StA-PLAN in (Khoo et al., 2020) as our backbone. StA-PLAN is a hierarchical transformer which contains 12 post-level multi-head attention layer (MHA) and 2 token-level MHA layers. As claimed in (Khoo et al., 2020) that BERT (Devlin et al., 2018) did not improve results and was time-consuming, we apply GLOVE-300d (Pennington et al., 2014) to embed each token in a post. The initial learning rate was set as 0.01 with 0.3 dropout and we used the ADAM optimizer with 6000 warm start-up steps. Batch size is set as 256 for all cross-validation tasks.

4.2 Dataset

We use the public PHEME dataset (Zubiaga et al., 2016) for our evaluation. PHEME was collected based on 9 breaking news stories and can be categorised into four classes: true rumor, false rumour, unverified rumour and non-rumour. Following the setting in (Kumar and Carley, 2019), we select five breaking events from PHEME and split them into two sets. Four events are chosen for training and in-domain testing, and the remaining one is used as out-of-domain testing set.

4.3 Baselines and Our Methods

We consider a state-of-the-art model and some baselines that are also addressing cross-domain issues.

StA-HiTPLAN: Replicating (Khoo et al., 2020), we train a hierarchical transformer model which is a state-of-the art model and can be viewed as a feature extractor in the following experiments.

Ensemble-based model (EM): Following (He et al., 2019; Clark et al., 2019), we take topical words as bias features and introduce an auxiliary bias_only model f_b taking bias priori features as input. Then using this bias_only model to train a robust model through an ensemble model. We firstly obtain the class distribution $p_b(y|x)$ using this biased model. Then we train an ensemble model that combines the former biased model with a robust model through this function: $p(\hat{y}|x) = T(p(y|x) + p_b(y|x))$. In the testing stage, only the robust model $p(y|x)$ is used for prediction.

Adversary-based model (AM): This is a common way to learn domain-invariant features. We implement a recent work (Wang et al., 2018; Ma et al., 2019b) and replace their Bi-LSTM with (Khoo et al., 2020) as backbones for fair comparison. The parameter of the gradient reversal layer is set as 1.

MOE-Avg and **MOE-Param:** These are our proposed models, where MOE-Avg and MOE-Param are according to our descriptions given in Section 3.2.

4.4 Cross-Topic and In-Topic Settings

We use two settings to evaluate the effectiveness of our method for cross-topic rumor detection. The first setting is the standard setting where we train on a set of source topics and test the performance of the model on a different target topic. For PHEME dataset, we use 4 topics as training topics and the remaining topic as the test topic. We repeat this 5

Method	Orig.		Mask-20		Mask-50		Mask-100	
	Accuracy	Macro F	Accuracy	Macro F	Accuracy	Macro F	Accuracy	Macro F
StA-HiTPLAN	67.69	62.69	36.70	34.12	35.26	30.86	34.13	29.78
EM	72.31	72.77	38.46	32.28	37.65	31.92	37.58	31.62
AM	66.81	60.65	47.47	38.54	43.35	34.99	42.15	33.93
MOE-Avg	76.48	73.72	50.01	39.56	47.42	37.11	44.54	33.02
MOE-Param	76.54	73.85	50.23	39.74	47.51	37.46	44.82	33.17

Table 2: Average accuracy and macro-F scores (%) of the in-topic setting on PHEME. Orig. refers to the original data. Mask-k refers to the setting where we artificially mask k topic-specific words.

times with different split of training/test topics, and report the average performance. We refer to this as the "cross-topic" setting. We also experiment with a second in-topic setting, where we train and test on the same topic, but we artificially remove topic-specific words. We refer to this as our "in-topic" setting. In Table 2, these are labeled as Mask-20, Mask-30 and Mask-50, depending on how many topic-specific words we mask (i.e., remove).

4.5 Results and Analysis

We present our experiments on the PHEME dataset in Table 2 and Table 3. Several observations can be made from the experiment results:

1) From Table 3, we can see that MOE-Avg and MOE-Param are both effective strategies that mitigate the topic overfitting problem. The accuracy improves from 34.41% to 41.24% and 41.33%, respectively, when we only intervene feature without modify the backbone network. 2) Adversarial training model AM works better than ensemble methods EM in the early stage but deteriorates after we mask more than 50 event sensitive words. One reason is ensemble-based model depends on the bias only model : the model is sensitive to the choice of bias, and seems more robust when we mask more irrelevant words. 3) Instead of unstable adversarial training method, we show that MOE-Avg and MOE-Param can make the model robust to topic bias and increase generalization ability. 4) Instead of using the average of the vector representation of x for those x belonging to the same topic, we also aggregate the final layer parameters of topic classifier. MOE-Param works slightly better than MOE-Avg method. More attention can be given to how to better represent a topic embedding in future work.

Method	NON-MASK	
	Accuracy	Macro F
StA-HiTPLAN	34.41	32.69
EM	39.96	34.79
AM	38.03	34.32
MOE-Avg	41.24	36.84
MOE-Param	41.33	36.95

Table 3: Average accuracy and macro-F score (%) on PHEME data for the cross-topic setting.

5 Conclusion and Future work

In this work, we propose a new cross-topic rumor detection task base on mixture of experts, which can reinforce the generalization capacity of a model when adapting to new topics. we suggest that: 1) instead of training an unstable adversarial component or removing bias directly from semantic contents, the mixture of experts provides us with another way to increase generalization ability. 2) in this work, we use feature concatenation and train one classifier rather than several expert classifiers, and utilize a fixed confidence score. In the future, we can learn adaptive weights to make the model more flexible. For example, we could use variational inference methods to dynamically learn the best mixture of topics for a given held-out topic.

6 Acknowledgment

We thank the reviewers for their valuable comments. This research is supported by DSO grant DSOCL18009.

References

Carlos Castillo, Marcelo Mendoza, and Barbara Poblete. 2011. Information credibility on twitter. In *Proceedings of the 20th international conference on World wide web*, pages 675–684.

Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2019. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. *arXiv preprint arXiv:1909.03683*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

He He, Sheng Zha, and Haohan Wang. 2019. Unlearn

dataset bias in natural language inference by fitting the residual. *arXiv preprint arXiv:1908.10763*.

Robert A Jacobs, Michael I Jordan, Steven J Nowlan, and Geoffrey E Hinton. 1991. Adaptive mixtures of local experts. *Neural computation*, 3(1):79–87.

Ling Min Serena Khoo, Hai Leong Chieu, Zhong Qian, and Jing Jiang. 2020. Interpretable rumor detection in microblogs by attending to user interactions. *arXiv preprint arXiv:2001.10667*.

Sumeet Kumar and Kathleen M Carley. 2019. Tree lstms with convolution units to predict stance and rumor veracity in social media conversations. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5047–5058.

Yang Liu and Yi-Fang Brook Wu. 2018. Early detection of fake news on social media through propagation path classification with recurrent and convolutional networks. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Jing Ma, Wei Gao, Shafiq Joty, and Kam-Fai Wong. 2019a. Sentence-level evidence embedding for claim verification with hierarchical attention networks. Association for Computational Linguistics.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2019b. Detect rumors on twitter by promoting information campaigns with generative adversarial learning. In *The World Wide Web Conference*, pages 3049–3055.

Burt L Monroe, Michael P Colaresi, and Kevin M Quinn. 2008. Fightin'words: Lexical feature selection and evaluation for identifying the content of political conflict. *Political Analysis*, 16(4):372–403.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Yaqing Wang, Fenglong Ma, Zhiwei Jin, Ye Yuan, Guangxu Xun, Kishlay Jha, Lu Su, and Jing Gao. 2018. Eann: Event adversarial neural networks for multi-modal fake news detection. In *Proceedings of the 24th acm sigkdd international conference on knowledge discovery & data mining*, pages 849–857.

Arkaitz Zubiaga, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Peter Tolmie. 2016. Analysing how people orient to and spread rumours in social media by looking at conversational threads. *PloS one*, 11(3).