Learning Disentangled Latent Topics for Twitter Rumour Veracity Classification

[1]John Dougrez-Lewis [1,2,3]Maria Liakata [2,3]Elena Kochkina [1]Yulan He
[1]Department of Computer Science, University of Warwick, UK
[2]Queen-Mary University of London, UK
[3]Alan Turing Institute, UK
{j.Dougrez-Lewis,yulan.he}@warwick.ac.uk
{m.liakata,e.kochkina}@qmul.ac.uk

Abstract

With the rapid growth of social media in the past decade, the news are no longer controlled by just a few mainstream sources. Users themselves create large numbers of potentially fictitious rumours, necessitating automated veracity classification systems. Here we present a novel approach towards automatically classifying rumours circulating on Twitter with respect to their veracity. We use a model built on Variational Autoencoder which disentangles the informational content of a tweet from the manner in which the information is written. This is achieved by obtaining latent topic vectors in an adversarial learning setting using the auxiliary task of stance classification. The latent vectors learnt in this way are used to predict rumour veracity, obtaining state-of-the-art accuracy scores on the PHEME dataset.[1]

1 Introduction

Anyone can publish rumours online with the potential to influence and pose as news. Since it is impossible to manually check the vast volume of circulating tweets, there is increasing need for machine learning algorithms to assist with rumour veracity assessment.

Given a rumour of unknown veracity introduced by a tweet in a conversation thread and the responses to it, our goal is to automatically determine the veracity of the rumour by assigning it one of the classes *true*, *false*, or *unverified*. Prior approaches to rumour veracity classification have primarily relied on careful feature engineering. For example, Li et al. (2019a) used meta-features such as user credibility together with more traditional features to top the leaderboard in SemEval 2019 Task 7 (Gorrell et al., 2019). This task encouraged teams to use the stances of responses to the rumour to assist in veracity classification, which has previously been shown to be predictive of rumour veracity (Dungs et al., 2018). A number of approaches (Kochkina et al., 2018; Li et al., 2019b) also showed benefits of using stance classification as an auxiliary task in a multitask learning setup. Some recent approaches exploit the structure of the conversation discussing a rumour. Kochkina et al. (2018) used LSTM to model linear branches extracted from the conversation tree, while Ma et al. (2018) and Bian et al. (2020) modelled a tree structure to capture information from responses.

Zeng et al. (2019) presented an unsupervised approach built on Variational Autoencoder (VAE) to jointly model topic content and discourse behaviour in microblog conversations. We propose a novel architecture which incorporates a VAE with adversarial learning to disentangle topics which are informative for stance classification from those which are not. We then derive tweet representations based on the word representations learned in the latent stance-dependent topic space. Our results show that using such tweet representations for rumour veracity classification achieves superior performance on the PHEME dataset. In summary, we have made the following contributions:

- We have proposed a disentanglement based approach to rumour veracity classification which achieves state-of-the-art performance for classifying rumours from previously unseen events, as they would emerge in real life.
- We have demonstrated that there remains significant overlap between separate rumourous events with distinct vocabularies, facilitating transfer learning between such events.

[1] The code is available at: https://github.com/JohnNLP/SAVED

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 3902–3908
August 1–6, 2021. ©2021 Association for Computational Linguistics

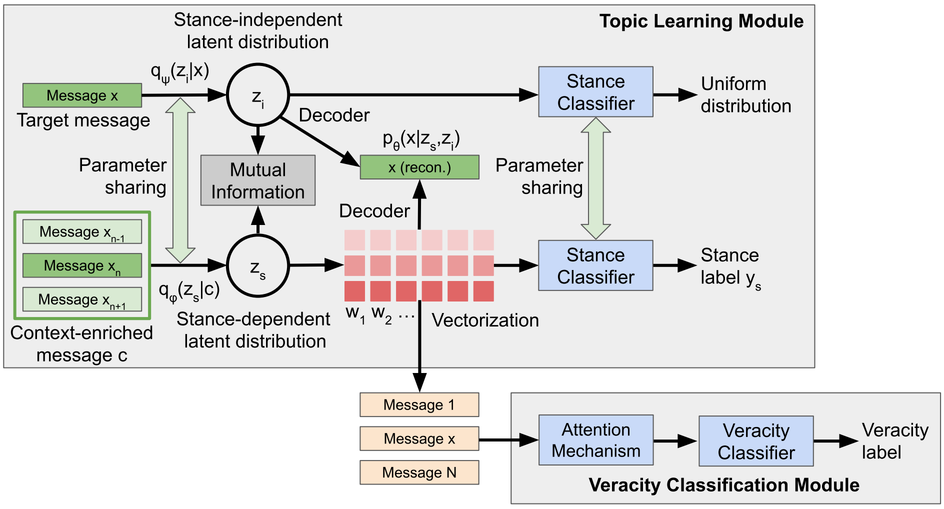

Figure 1: Model Architecture. Note the separate *Topic Learning* and *Veracity Classification* modules, and that the context-enriched message often uses a fixed size window of messages rather than the full conversation.

2 Proposed Model

Our proposed Stance-Augmented VAE Disentanglement framework (SAVED) is shown in Figure 1. It consists of two main modules, the *Topic Learning* module and the *Veracity Classification* module.

The idea is to separate the factual content of twitter rumours from their mannerisms[2], using the latter to predict rumour veracity. This technique is well-suited to rumour veracity prediction for emergent real-life events as it overcomes their factual distinctness - we hypothesise that mannerisms transfer better between different rumourous events. The PHEME dataset is designed for this purpose, with rumours grouped together according to their originating event. In our experiments we use both the source tweet originating the rumour and its conversation thread together since both are useful for veracity prediction.

2.1 Topic Learning

In microblog conversations, a source tweet could have multiple responses, forming a conversation tree. Here we flatten the tree into a chronologically ordered sequence of tweets, defined as $\boldsymbol{d} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_N\}$, where N denotes the total number of posts in the conversation. For each message $\boldsymbol{x}_n \in \boldsymbol{d}$, we construct its context-enriched message by taking a window of M messages before and M messages after the target message, denoted as $\boldsymbol{c}_n = \{\boldsymbol{x}_{n-M}, \cdots \boldsymbol{x}_{n-1}, \boldsymbol{x}_n, \boldsymbol{x}_{n+1}, \cdots, \boldsymbol{x}_{n+M}\}$.

Assuming a post $\boldsymbol{x}$[3] is associated with a stance label y and each post can be generated from a stance-dependent latent topic $\boldsymbol{z}_s$ (determined by the context-enriched message $\boldsymbol{c}$) and a stance-independent latent topic $\boldsymbol{z}_i$, we aim to learn a model which maximises the joint data and label log-likelihood, $\log p(\boldsymbol{x}, y)$:

$$
\begin{aligned}
&\log p(\boldsymbol{x}, y) = \log \int_{\boldsymbol{z}_s} \int_{\boldsymbol{z}_i} p(\boldsymbol{x}, y, \boldsymbol{z}_s, \boldsymbol{z}_i) d\boldsymbol{z}_s d\boldsymbol{z}_i \\
&\geq E_{q_\phi(\boldsymbol{z}_s|\boldsymbol{c},y), q_\psi(\boldsymbol{z}_i|\boldsymbol{x})}[\log p_\theta(\boldsymbol{x}|\boldsymbol{z}_s, \boldsymbol{z}_i)] \\
&+ E_{q_\phi(\boldsymbol{z}_s|\boldsymbol{c},y), q_\psi(\boldsymbol{z}_i|\boldsymbol{x})}[\log p_\theta(y|\boldsymbol{z}_s)] \\
&- KL(q_\phi(\boldsymbol{z}_s|\boldsymbol{c}, y)||p(\boldsymbol{z})) - KL(q_\psi(\boldsymbol{z}_i|\boldsymbol{x})||p(\boldsymbol{z}_i))
\end{aligned}
$$

Following the idea of Zeng et al. (2019), we can compute a variational approximation to an intractable posterior using MLPs. We aim to minimise the reconstruction loss for each context-enriched message $\boldsymbol{c}$ and for each message $\boldsymbol{x}$ (see Figure 1), with a Monte Carlo approximation using L independent samples:

$$
\mathcal{L}_{\boldsymbol{c}} \approx \frac{1}{L} \sum_{l=1}^{L} \sum_{n=1}^{N} \log p(c_n|\boldsymbol{z}_s^{(l)}) - KL(q_\phi(\boldsymbol{z}_s|\boldsymbol{c}, y)||p(\boldsymbol{z}_s)) \tag{1}
$$

$$
\mathcal{L}_{\boldsymbol{x}} \approx \frac{1}{L} \sum_{l=1}^{L} \sum_{n=1}^{N} \log p(x_n|\boldsymbol{z}_s^{(l)}, \boldsymbol{z}_i^{(l)}) - KL(q_\psi(\boldsymbol{z}_i|\boldsymbol{x})||p(\boldsymbol{z}_i)) \tag{2}
$$

We assume that the latent stance-independent topics, $\boldsymbol{z}_i$, are independent of stance classes, and hence, when feeding into a stance classifier, should generate a uniform stance class distribution (similar to adversarial learning). On the contrary, the

[2] For an example of this, see Figure 5 in (Zeng et al., 2019).

[3] We drop the subscript n for clarity here.

latent stance-dependent, $\boldsymbol{z}_s$, should bear essential information to discriminate between stance classes. Therefore, we can define the following two cross-entropy losses for stance classification:

$$\mathcal{L}_{adv} = -E_{q_\phi(\boldsymbol{z}_i)} \sum_{s=1}^{S} \frac{1}{S} \log p(\hat{y}_s|\boldsymbol{z}_i) \qquad (3)$$

$$\mathcal{L}_{stance} = -E_{q_\psi(\boldsymbol{z}_s)} \sum_{s=1}^{S} y_s \log p(\hat{y}_s|\boldsymbol{z}_s) \qquad (4)$$

where S is the total number of stance classes, $\frac{1}{S}$ represents the uniform stance class distribution.

To disentangle the latent stance-independent topics, $\boldsymbol{z}_i$, and and latent stance-dependent topics, $\boldsymbol{z}_s$, we minimise the mutual information between them:

$$\mathcal{L}_{MI} = E_{q_\phi(\boldsymbol{z}_i)q_\psi(\boldsymbol{z}_s)} \log \frac{p(\boldsymbol{z}_i, \boldsymbol{z}_s)}{p(\boldsymbol{z}_i)p(\boldsymbol{z}_s)} \qquad (5)$$

Our final objective function is:

$$\mathcal{L} = \mathcal{L}_{\boldsymbol{c}} + \mathcal{L}_{\boldsymbol{x}} + \alpha\mathcal{L}_{adv} + \beta\mathcal{L}_{stance} + \gamma\mathcal{L}_{MI} \quad (6)$$

where α, β and γ control the relative contribution of various loss functions.

We exclude the source tweet from this input since it is worded differently from the other tweets and is hence detrimental to overall performance.

2.2 Veracity Module

Once the Topic Learning Module is trained, we use the decoder weights, i.e., weights linking between the stance-dependent latent vector z_s and the bag-of-words representation of the reconstructed message (see the shaded pink blocks in Figure 1) to generate the input to the Veracity Module by deriving a vector for each tweet in the conversation tree - the mean average of the tweet's constituent words. This two-module approach is advantageous since it allows tweets to be weighted based on their position in the (unflattened) conversation tree, and forgoes use of the window of size M employed by the Topic Module. Vectorised tweets are grouped into 3 different importance tiers based on their positions in the conversation tree: the source tweet, direct responses, and all other responses. The rationale for this is in the observation that the later responses in long conversations tend to be less relevant to the verification of the rumour (Zubiaga et al., 2016). The Veracity Module itself consists of an attention layer followed by two dense hidden layers with leaky ReLU activation (Maas et al., 2013). The attention layer learns weights for each of the tweet tiers. The conversation representation is then obtained as a weighted average: $(\boldsymbol{w}_1\boldsymbol{v}_1 + \boldsymbol{w}_2\boldsymbol{v}_2 + \boldsymbol{w}_3\boldsymbol{v}_3)/n$, where $\boldsymbol{w}_t$ denotes the learnable weight for tier $\boldsymbol{t}$, $\boldsymbol{v}_t$ denotes the sum of the vectors of tweets in $\boldsymbol{t}$ and n denotes the number of tweets in the entire tree. In our experiments the presence of these weights increases model performance. We found that the model tends to give the highest weight to the source tweet, followed by direct responses and finally the rest. To mitigate the class imbalance, the loss attributed to instances of each class is weighted inversely to their frequency in the training data.

3 Experimental Setup

Dataset We use the PHEME-5 dataset (Kochkina et al., 2018), which consists of Twitter rumours around 5 high profile real-world events. Statistics regarding the dataset can be found in the Appendix. The PHEME dataset was chosen as it is a particularly challenging dataset due to class imbalance and a leave-one-event out cross validation setting, reflecting a real-world evaluation scenario.

Baseline Models We perform comparison of the proposed model SAVED with existing state-of-the-art models (Kochkina et al., 2018; Li et al., 2019b; Cheng et al., 2020) and several strong baselines, described in this section.

BERT We use the pretrained BERT$_{\text{BASE}}$ (Devlin et al., 2019), uncased, which consists of 12 self-attention layers, and returns a 768-dimension vector representation of a sentence. We generate BERT representations for each tweet in the conversation before feeding them into the Veracity Module.

VAED is a version of SAVED, where the Topic Learning Module is reduced to only its VAE component, without the stance classifiers from Section 2.1. The loss is $\mathcal{L}_{\boldsymbol{c}} + \mathcal{L}_{\boldsymbol{x}} + \gamma\mathcal{L}_{MI}$.

VAED Without Disentanglement (VAE) is a simplified version of VAED, where the Topic Learning Module is reduced to only using loss from the context-enriched latent factor without the target message (disentanglement). The loss is $\mathcal{L}_{\boldsymbol{c}}$.

VAED With Veracity (VAED+V) is an end-to-end classifier based on the Topic Learning Module, in which the context-enriched segment of the VAED z_s is connected to the veracity classifier. This model does not include a stance classifier, nor the adversarial component. The loss is $\mathcal{L}_{\boldsymbol{c}} + \mathcal{L}_{\boldsymbol{x}} + \beta\mathcal{L}_{veracity} + \gamma\mathcal{L}_{MI}$.

Model	**False**	**True**	**Unverified**	**Accuracy**	**MacroF1**
Kochkina et al. (2018)	0.212	**0.647**	0.330	0.492	0.396
Cheng et al. (2020)	**0.504**	0.480	0.465	0.521	**0.484**
Li et al. (2019b)	-	-	-	0.483	0.418
BERT Baseline	0.113	0.592	0.326	0.405	0.345
VAE	0.201	0.413	0.407	0.395	0.339
VAED	0.206	0.474	0.388	0.380	0.362
VAED+V	0.273	0.418	0.420	0.389	0.376
SAVED	0.164	0.642	**0.531**	**0.528**	0.434

Table 1: Comparison with baselines and previous results. For comparability, we pool together the results of all five events *before* calculating any F1 scores - the same approach used by the prior work in this table.

Parameter Settings The dimensionality of each latent factor was tuned via grid search, with peak performance found at 10 dimensions for the latent stance-dependent vector, z_s, and 6 dimensions for the latent stance-independent vector, z_i. See Appendix for further details.

Evaluation Metrics For comparability with prior work, F1 scores are calculated *after* combining the results of each fold.

4 Experimental Results

Overall Results The results of our experiments are shown in Table 1. All of the models outperformed the VAE baseline. The VAED model alone (with disentanglement, without any other modifications or stance/veracity classifiers) scores 0.363, showcasing the efficacy of disentanglement *per se*. The BERT-based model only outperformed VAE.

The proposed SAVED model outperforms those of prior work on overall accuracy and the *True* and *Unverified* classes. However, results for the *False* class are rather low - which is in fact the case for most of the models in Table 1, with only Cheng et al. (2020) being an exception. Results of VAED+V are lower than that of SAVED, in line with the knowledge that stance is related to veracity (Dungs et al., 2018). This suggests that stance is also a worthwhile intermediate classification target.

Event	**False**	**True**	**Unverified**	**MacroF1**
Charlie Hebdo	0.223	0.505	0.324	0.351
Ferguson	0.129	0.080	0.906	0.372
Germanwings Crash	0.033	0.520	0.289	0.281
Ottawa Shooting	0.058	0.735	0.119	0.304
Sydney Siege	0.157	0.700	0.140	0.332
Overall	**0.164**	**0.642**	**0.531**	**0.434**

Table 2: Per-fold evaluation results of SAVED.

Per-fold Results Table 2 shows the per-fold results in our leave-one-event-out setting. Interestingly, the model tends to perform best on rumors of *True* veracity and worst on those which are *False*. Performance on the *Unverified* class is adequate, except for the ‘Ferguson’ event in which the model F1 score is 0.906. Overall, the F1 scores of *True* and *Unverified* classes are rather high (0.642 and 0.531 respectively) compared with that of the *False* class (0.164).

Ablation studies The results of ablation studies are shown in Table 3, which were obtained by varying which latent factor of the Topic Learning Module were fed to the Veracity Module. We found that stance-dependent latent vectors performed better than stance-independent ones for rumour veracity classification, although each performed adequately. Creating an ensemble of both latent vectors was not helpful since the respective models had similar strengths and weaknesses.

Components Used	**MacroF1**
Stance-dependent	0.434
Stance-independent	0.375
Both together	0.395

Table 3: Varying the latent factor used by the Veracity Classification Module of SAVED.

Visualisation To examine the effectiveness of our conversation tree representations derived from the Topic Learning Module, we visualised the representative vector for each tree using t-SNE (van der Maaten and Hinton, 2008). This was done in the context of the model SAVED (See Sec. 2). Since learned tree-position weights $\boldsymbol{w}_t$ are an important part of generating the representation of the conversation but are not part of the Topic Learning Module, we obtained them from the attention component of the veracity module. Figure 2 depicts the resulting clusters of points, with each cluster roughly corresponding to a class. The "*Unverified*" cluster is particularly distinct. This cluster is largely comprised of tweets from the 'Ferguson' event, which contains most of the unverified rumours in the dataset. This further demonstrates that our Stance-Augmented VAE Model generates representations which are useful for veracity prediction.

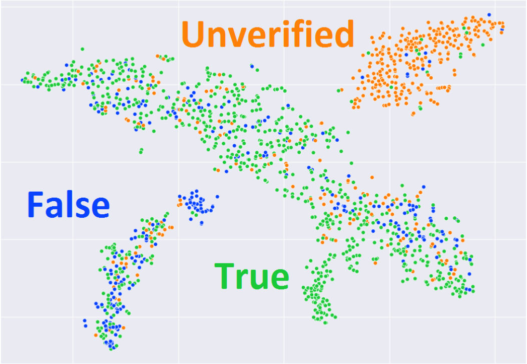

Figure 2: Visualisation of the stance-dependent latent factor for the 'Charlie Hebdo'-excluded fold.

Number of Responses and Performance We investigated how the number of responses to a rumour affect the model's accuracy. Whilst rumours with more responses provide more context for the model, responses too far down the response-tree have been noted to lose some of their relevance to the source rumour. We found the average number of responses for correctly classified rumours from the 'Charlie' fold, where 'Charlie Hebdo' is used as a testing set, to be 14.8 tweets, whilst incorrectly classified ones had an average length 12.2. This seemed stronger for the 'Ferguson' fold with averages of 22.2 and 14.7 respectively (note that the model only observes up to 20 responses per thread).

Using the 2-tailed Mann-Whitney test, these results approach statistical significance (p=0.08 for each fold). If there is an effect here, it can partially be explained by rumours with 3 or fewer responses, since when these were excluded the averages became 17.1 vs 15.4 for 'Charlie' and 25.2 vs 20.2 for 'Ferguson'.

Error Analysis Our model, similar to those of Kochkina et al. (2018) and others but not Cheng et al. (2020), fails to perform well for the *False* class. To our knowledge, this under-performance has not been previously investigated.

The PHEME dataset is imbalanced, containing 1,012 *True*, 393 *False* and *571* Unverified rumours. Although we attempted to account for this imbalance by weighting the model's loss inversely to class frequency, it is possible that the imbalance contributed to the poor performance on *False*. Interestingly more rumours were classified as *Unverified* than as *False*, although there was no clear pattern of misclassification.

The numbers of responses are largely unchanged when restricted to the *False* class alone, with 13.0 (correct) and 10.7 (incorrect) for the 'Charlie' fold, so this does not explain the performance deficit.

Manual investigation of the rumours themselves led to the observation that correctly classified *False* rumours tended to be more straightforward than those which were incorrectly classified. The latter seemed more likely to be vague or contain multiple parts, examples of which can be found in the Appendix. Thus a *False* rumour may contain both false and true statements, potentially lowering the utility of user responses for classification. Accordingly, the model by Cheng et al. (2020) appears to rely less on responses than ours and that of Kochkina et al. (2018).

5 Conclusion

We present a novel disentanglement-based approach to rumour veracity classification, achieving state-of-the-art results for accuracy, towards classification on previously unseen events from the PHEME dataset. Our results suggest that although unique events each have their own vocabulary, there is still sufficient common ground between them for stance-dependent driven rumour veracity classification to be effective.

Acknowledgements

This work was supported by a UKRI/EPSRC grant (EP/V048597/1) to Profs Yulan He and Maria Liakata as well as project funding from the Alan Turing Institute, grant EP/N510129/1. JDL was funded by the EPSRC Doctoral Training Grant. ML and YH are supported by Turing AI Fellowships (EP/V030302/1, EP/V020579/1).

References

Tian Bian, Xi Xiao, Tingyang Xu, Peilin Zhao, Wenbing Huang, Yu Rong, and Junzhou Huang. 2020. Rumor detection on social media with bi-directional graph convolutional networks. In *Proceedings of* the AAAI Conference on Artificial Intelligence, volume 34, pages 549–556.

Mingxi Cheng, Shahin Nazarian, and Paul Bogdan. 2020. Vroc: Variational autoencoder-aided multi-task rumor classifier based on text. In *Proceedings of The Web Conference 2020*, WWW '20, page 2892–2898, New York, NY, USA. Association for Computing Machinery.

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.

Sebastian Dungs, Ahmet Aker, Norbert Fuhr, and Kalina Bontcheva. 2018. Can rumour stance alone predict veracity? In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3360–3370, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Genevieve Gorrell, Elena Kochkina, Maria Liakata, Ahmet Aker, Arkaitz Zubiaga, Kalina Bontcheva, and Leon Derczynski. 2019. SemEval-2019 task 7: RumourEval, determining rumour veracity and support for rumours. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 845–854, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Elena Kochkina, Maria Liakata, and Arkaitz Zubiaga. 2018. All-in-one: Multi-task learning for rumour verification. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3402–3413, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Quanzhi Li, Qiong Zhang, and Luo Si. 2019a. eventAI at SemEval-2019 task 7: Rumor detection on social media by exploiting content, user credibility and propagation information. In *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 855–859, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Quanzhi Li, Qiong Zhang, and Luo Si. 2019b. Rumor detection by exploiting user credibility information, attention and multi-task learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1173–1179, Florence, Italy. Association for Computational Linguistics.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2018. Rumor detection on twitter with tree-structured recursive neural networks. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1980–1989, Melbourne, Australia. Association for Computational Linguistics.

Andrew L Maas, Awni Y Hannun, and Andrew Y Ng. 2013. Rectifier nonlinearities improve neural network acoustic models. In *Proc. icml*, volume 30, page 3. Citeseer.

Laurens van der Maaten and Geoffrey Hinton. 2008. Viualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

Jichuan Zeng, Jing Li, Yulan He, Cuiyun Gao, Michael R Lyu, and Irwin King. 2019. What you say and how you say it: Joint modeling of topics and discourse in microblog conversations. *Transactions of the Association for Computational Linguistics*, 7:267–281.

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):e0150989.

A Dataset

Table A1 shows the overall statistics of the PHEME dataset. It can be observed that the dataset is highly imbalanced with nearly 50% ‘True’ instances and relatively few ‘False’ instances.

Event	#Rumours	#True	#False	#Unverified
Charlie Hebdo	458	193	116	149
Ferguson	284	10	8	266
Germanwings Crash	238	94	111	33
Ottawa Shooting	470	329	72	69
Sydney Siege	522	382	86	54
Total	1972	1012	393	571

Table A1: Dataset (PHEME-5) overview.

B Data Preprocessing

We perform pre-processing on the PHEME data by using special tokens for URLs, username, hashtags, and numbers. We also lowercase text and expand words with apostrophes (e.g., ‘*we’re*’ becomes ‘*we are*’). The Topic-Learning module also excludes words which occur fewer than 20 times throughout the training corpus.

C Model Details

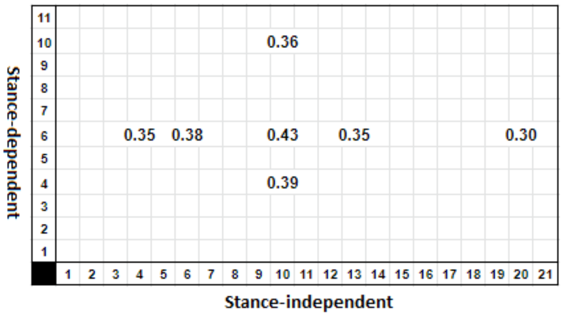

Figure A1: Grid search of the dimensions of the latent stance-dependent and stance-independent vectors.

The Topic-Learning Model has 15M parameters and takes around 6 hours to train on a computer with RTX 1080 Ti. The Veracity Model takes around 15 minutes to train on the same machine. We perform grid search on the dataset to identify the optimal setting of the dimensions of latent stance-dependent and stance-independent vectors, with the dimension of the former varying between 4 and 20, while the dimension of the latter varying between 4 and 10. The results are shown in Figure A1. The veracity classification results are obtained by evaluating on the validation set. It can be observed that the model achieves the best result when the dimension of the latent stance-dependent vector is set to 10 while that of the latent stance-independent vector is set to 6. For further model details, such as layer sizes and activations, the reader is advised to look at the linked source code.

D False Rumour Examples

Note that a rumour being of a certain type (as below) does not guarantee its predicted class or the correctness of its classification.

D.1 Basic (classified correctly)

breaking three gunmen involved in attack on charlie hebdo magazine , french interior minister bernard cazeneuve says . URL

D.2 Vague (classified incorrectly)

HASHTAG banksy’s response to today’s incident in paris via his official HASHTAG instagram acct URL HASHTAG charliehebdo URL

D.3 Multi (classified incorrectly)

two police officers have been injured in a shooting in HASHTAG montrouge in southern HASHTAG paris - there is no direct link with the HASHTAG charliehebdo attack