DTCA: Decision Tree-based Co-Attention Networks for Explainable Claim Verification

Lianwei Wu, Yuan Rao, Yongqiang Zhao, Hao Liang, Ambreen Nazir
Lab of Social Intelligence and Complexity Data Processing,
School of Software Engineering, Xi'an Jiaotong University, China
Shannxi Joint Key Laboratory for Artifact Intelligence(Sub-Lab of Xi'an Jiaotong University), China
Research Institute of Xi'an Jiaotong University, Shenzhen, China
{stayhungry,yongqiang1210,favorablelearner,ambreen.nazir}@stu.xjtu.edu.cn
raoyuan@mail.xjtu.edu.cn

Abstract

Recently, many methods discover effective evidence from reliable sources by appropriate neural networks for explainable claim verification, which has been widely recognized. However, in these methods, the discovery process of evidence is nontransparent and unexplained. Simultaneously, the discovered evidence only roughly aims at the interpretability of the whole sequence of claims but insufficient to focus on the false parts of claims. In this paper, we propose a Decision Tree-based Co-Attention model (DTCA) to discover evidence for explainable claim verification. Specifically, we first construct Decision Tree-based Evidence model (DTE) to select comments with high credibility as evidence in a transparent and interpretable way. Then we design Co-attention Self-attention networks (CaSa) to make the selected evidence interact with claims, which is for 1) training DTE to determine the optimal decision thresholds and obtain more powerful evidence; and 2) utilizing the evidence to find the false parts in the claim. Experiments on two public datasets, RumourEval and PHEME, demonstrate that DTCA not only provides explanations for the results of claim verification but also achieves the state-of-the-art performance, boosting the F1-score by 3.11%, 2.41%, respectively.

1 Introduction

The increasing popularity of social media has brought unprecedented challenges to the ecology of information dissemination, causing rampancy of a large volume of false or unverified claims, like extreme news, hoaxes, rumors, fake news, etc. Research indicates that during the US presidential election (2016), fake news accounts for nearly 6% of all news consumption, where 1% of users are exposed to 80% of fake news, and 0.1% of users are responsible for sharing 80% of fake news (Grinberg et al., 2019), and democratic elections are vulnerable to manipulation of the false or unverified claims on social media (Aral and Eckles, 2019), which renders the automatic verification of claims a crucial problem.

Currently, the methods for automatic claim verification could be divided into two categories: the first is that the methods relying on deep neural networks learn credibility indicators from claim content and auxiliary relevant articles or comments (i.e., responses) (Volkova et al., 2017; Rashkin et al., 2017; Dungs et al., 2018). Despite their effectiveness, these methods are difficult to explain why claims are true or false in practice. To overcome the weakness, a trend in recent studies (the second category) is to endeavor to explore evidence-based verification solutions, which focuses on capturing the fragments of evidence obtained from reliable sources by appropriate neural networks (Popat et al., 2018; Hanselowski et al., 2018; Ma et al., 2019; Nie et al., 2019). For instance, Thorne et al. (2018) build multi-task learning to extract evidence from Wikipedia and synthesize information from multiple documents to verify claims. Popat et al. (2018) capture signals from external evidence articles and model joint interactions between various factors, like the context of a claim and trustworthiness of sources of related articles, for assessment of claims. Ma et al. (2019) propose hierarchical attention networks to learn sentence-level evidence from claims and their related articles based on coherence modeling and natural language inference for claim verification.

Although these methods provide evidence to solve the explainability of claim verification in a manner, there are still several limitations. **First**, they are generally hard to interpret the discovery process of evidence for claims, namely, lack the interpretability of methods themselves because these methods are all based on neural networks, belong-

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 1024–1035
July 5 - 10, 2020.

ing to nontransparent black box models. **Secondly**, the provided evidence only offers a coarse-grained explanation to claims. They are all aimed at the interpretability of the whole sequence of claims but insufficient to focus on the false parts of claims.

To address the above problems, we design **D**ecision **T**ree-based **C**o-**A**ttention networks (DTCA) to discover evidence for explainable claim verification, which contains two stages: 1) **D**ecision **T**ree-based **E**vidence model (DTE) for discovering evidence in a transparent and interpretable way; and 2) **C**o-**a**ttention **S**elf-**a**ttention networks (CaSa) using the evidence to explore the false parts of claims. Specifically, DTE is constructed on the basis of structured and hierarchical comments (aiming at the claim), which considers many factors as decision conditions from the perspective of content and meta data of comments and selects high credibility comments as evidence. CaSa exploits the selected evidence to interact with claims at the deep semantic level, which is for two roles: one is to train DTE to pursue the optimal decision threshold and finally obtain more powerful evidence; and another is to utilize the evidence to find the false parts in claims. Experimental results reveal that DTCA not only achieves the state-of-the-art performance but also provides the interpretability of results of claim verification and the interpretability of selection process of evidence. Our contributions are summarized as follows:

- We propose a transparent and interpretable scheme that incorporates decision tree model into co-attention networks, which not only discovers evidence for explainable claim verification (Section 4.4.3) but also provides interpretation for the discovery process of evidence through the decision conditions (Section 4.4.2).
- Designed co-attention networks promote the deep semantic interaction between evidence and claims, which can train DTE to obtain more powerful evidence and effectively focus on the false parts of claims (Section 4.4.3).
- Experiments on two public, widely used fake news datasets demonstrate that our DTCA achieves more excellent performance than previous state-of-the-art methods (Section 4.3.2).

2 Related Work

Claim Verification Many studies on claim verification generally extract an appreciable quantity of credibility-indicative features around semantics (Ma et al., 2018b; Khattar et al., 2019; Wu et al., 2020), emotions (Ajao et al., 2019), stances (Ma et al., 2018a; Kochkina et al., 2018; Wu et al., 2019), write styles (Potthast et al., 2018; Gröndahl and Asokan, 2019), and source credibility (Popat et al., 2018; Baly et al., 2018a) from claims and relevant articles (or comments). For a concrete instance, Wu et al. (2019) devise sifted multi-task learning networks to jointly train stance detection and fake news detection tasks for effectively utilizing common features of the two tasks to improve the task performance. Despite reliable performance, these methods for claim verification are unexplainable. To address this issue, recent research concentrates on the discovery of evidence for explainable claim verification, which mainly designs different deep models to exploit semantic matching (Nie et al., 2019; Zhou et al., 2019), semantic conflicts (Baly et al., 2018b; Dvořák and Woltran, 2019; Wu and Rao, 2020), and semantic entailments (Hanselowski et al., 2018; Ma et al., 2019) between claims and relevant articles. For instance, Nie et al. (2019) develop neural semantic matching networks that encode, align, and match the semantics of two text sequences to capture evidence for verifying claims. Combined with the pros of recent studies, we exert to perceive explainable evidence through semantic interaction for claim verification.

Explainable Machine Learning Our work is also related to explainable machine learning, which can be generally divided into two categories: intrinsic explainability and post-hoc explainability (Du et al., 2018). Intrinsic explainability (Shu et al., 2019; He et al., 2015; Zhang and Chen, 2018) is achieved by constructing self-explanatory models that incorporate explainability directly into their structures, which requires to build fully interpretable models for clearly expressing the explainable process. However, the current deep learning models belong to black box models, which are difficult to achieve intrinsic explainability (Gunning, 2017). Post-hoc explainability (Samek et al., 2017; Wang et al., 2018; Chen et al., 2018) needs to design a second model to provide explanations for an existing model. For example, Wang et al. (2018) combine the strengths of the embeddings-based model and the tree-based model to develop explainable recommendation, where the tree-based model obtains evidence and the embeddings-based model

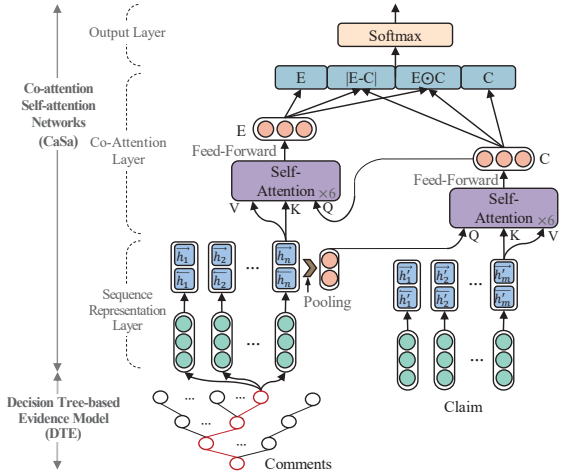


Figure 1: The architecture of DTCA. DTCA includes two stages, i.e., DTE for discovering evidence and CaSa using the evidence to explore the false parts of claims.

improves the performance of recommendation. In this paper, following the post-hoc explainability, we harness decision-tree model to explain the discovery process of evidence and design co-attention networks to boost the task performance.

3 Decision Tree-based Co-Attention Networks (DTCA)

In this section, we introduce the decision tree-based co-attention networks (DTCA) for explainable claim verification, with architecture shown in Figure 1, which involves two stages: decision tree-based evidence model (DTE) and co-attention self-attention networks (CaSa) that consist of a 3-level hierarchical structure, i.e., sequence representation layer, co-attention layer, and output layer. Next, we describe each part of DTCA in detail.

3.1 Decision Tree-based Evidence Model (DTE)

DTE is based on tree comments (including replies) aiming at one claim. We first build a tree network based on hierarchical comments, as shown in the left of Figure 2. The root node is one claim and the second level nodes and below are users' comments on the claim (R_{11}, ... ,R_{kn}), where k and n denote the depth of tree comments and the width of the last level respectively. We try to select comments with high credibility as evidence of the claim, so we need to evaluate the credibility of each node (comment) in the network and decide whether to select the comment or not. Three factors from the perspective of content and meta data of comments

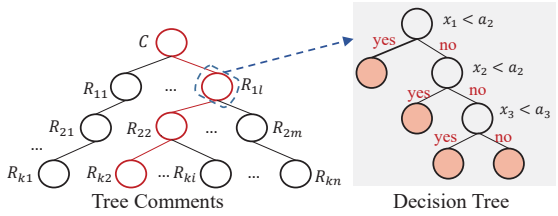


Figure 2: Overview of DTE. DTE consists of two parts: tree comment network (the left) and decision tree model (the right), which is used to evaluate the credibility of each node in the tree comment network for discovering evidence.

are considered and the details are described:

The semantic similarity between comments and claims. It measures relevancy between comments and claims and aims to filter irrelevant and noisy comments. Specifically, we adopt soft consine measure (Sidorov et al., 2014) between average word embeddings of both claims and comments as semantic similarity.

The credibility of reviewers[1]**.** It follows that "reviewers with high credibility also usually have high reliability in their comments" (Shan, 2016). Specifically, we utilize multiple meta-data features of reviewers to evaluate reviewer credibility, i.e., whether the following elements exist or not: verified, geo, screen name, and profile image; and the number of the items: followers, friends, and favorites. The examples are shown in Appendix A.

The credibility of comments. It is based on meta data of comments to roughly measure the credibility of comments (Shu et al., 2017), i.e., 1) whether the following elements exist or not: geo, source, favorite the comment; and 2) the number of favorites and content-length. The examples are shown in Appendix A.

In order to integrate these factors in a transparent and interpretable way, we build a decision tree model which takes the factors as decision conditions to measure node credibility of tree comments, as shown in the grey part in Figure 2.

We represent the structure of a decision tree model as $Q = \{V, E\}$, where V and E denote nodes and edges, respectively. Nodes in V have two types: decision (a.k.a. internal) nodes and leaf nodes. Each decision node splits a decision condition x_i (one of the three factors) with two decision edges (decision results) based on the specific decision threshold a_i. The leaf node gives the decision result (the red circle), i.e., whether the comment is

[1]People who post comments

selected or not. In our experiments, if any decision nodes are yes, the evaluated comment in the tree comment network will be selected as a piece of evidence. In this way, each comment is selected as evidence, which is transparent and interpretable, i.e., interpreted by decision conditions.

When comment nodes in the tree network are evaluated by the decision tree model, we leverage post-pruning algorithm to select comment subtrees as evidence set for CaSa (in section 3.2) training.

3.2 Co-attention Self-attention Networks (CaSa)

In DTE, the decision threshold a_i is uncertain, to say, according to different decision thresholds, there are different numbers of comments as evidence for CaSa training. In order to train decision thresholds in DTE so as to obtain more powerful evidence, and then exploit this evidence to explore the false parts of fake news, we devise CaSa to promote the interaction between evidence and claims. The details of DTCA are as follows:

3.2.1 Sequence Representation Layer

The inputs of CaSa include a sequence of evidence (the evidence set obtained by DTE model is concatenated into a sequence of evidence) and a sequence of claim. Given a sequence of length l tokens $\mathbf{X} = \{\mathrm{x}_1, \mathrm{x}_2, ..., \mathrm{x}_l\}$, $\mathbf{X} \in \mathbb{R}^{l \times d}$, which could be either a claim or the evidence, each token $\mathrm{x}_i \in \mathbb{R}^d$ is a d-dimensional vector obtained by pre-trained BERT model (Devlin et al., 2019). We encode each token into a fixed-sized hidden vector $\mathbf{h}_i$ and then obtain the sequence representation for a claim $\mathbf{X}^c$ and evidence $\mathbf{X}^e$ via two BiLSTM (Graves et al., 2005) neural networks respectively.

$$\overrightarrow{\mathbf{h}_\mathrm{i}} = \overrightarrow{\mathbf{LSTM}}(\overrightarrow{\mathbf{h}_{\mathrm{i}-1}}, \mathrm{x}_i) \quad (1)$$

$$\overleftarrow{\mathbf{h}_\mathrm{i}} = \overleftarrow{\mathbf{LSTM}}(\overleftarrow{\mathbf{h}_{\mathrm{i}+1}}, \mathrm{x}_i) \quad (2)$$

$$\mathbf{h}_\mathrm{i} = [\overrightarrow{\mathbf{h}_\mathrm{i}}; \overleftarrow{\mathbf{h}_\mathrm{i}}] \quad (3)$$

where $\overrightarrow{\mathbf{h}_i} \in \mathbb{R}^h$ and $\overleftarrow{\mathbf{h}_i} \in \mathbb{R}^h$ are hidden states of forward LSTM $\overrightarrow{\mathbf{LSTM}}$ and backward LSTM $\overleftarrow{\mathbf{LSTM}}$. h is the number of hidden units of LSTM. ; denotes concatenation operation. Finally, $\mathbf{R}^e \in \mathbb{R}^{l \times 2h}$ and $\mathbf{R}^c \in \mathbb{R}^{l \times 2h}$ are representations of sequences of both evidence and a claim. Additionally, experiments confirm BiLSTM in CaSa can be replaced by BiGRU (Cho et al., 2014) for comparable performance.

3.2.2 Co-attention Layer

Co-attention networks are composed of two hierarchical self-attention networks. In our paper, the sequence of evidence first leverages one self-attention network to conduct deep semantic interaction with the claim for capturing the false parts of the claim. Then semantics of the interacted claim focus on semantics of the sequence of evidence via another self-attention network for concentrating on the key parts of the evidence. The two self-attention networks are both based on the multi-head attention mechanism (Vaswani et al., 2017). Given a matrix of l query vectors $\mathbf{Q} \in \mathbb{R}^{l \times 2h}$, keys $\mathbf{K} \in \mathbb{R}^{l \times 2h}$, and values $\mathbf{V} \in \mathbb{R}^{l \times 2h}$, the scaled dot-product attention, the core of self-attention networks, is described as

$$\mathbf{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathbf{softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}})\mathbf{V} \quad (4)$$

Particularly, to enable claim and evidence to interact more directly and effectively, in the first self-attention network, $\mathbf{Q} = \mathbf{R}^e_{pool}$ ($R^e_{pool} \in \mathbb{R}^{2h}$) is the max-pooled vector of the sequence representation of evidence, and $\mathbf{K} = \mathbf{V} = \mathbf{R}^c$, $\mathbf{R}^c$ is the sequence representation of claim. In the second self-attention network, $\mathbf{Q} = \mathbf{C}$, i.e., the output vector of self-attention network for claim (the details are in Eq. 7), and $\mathbf{K} = \mathbf{V} = \mathbf{R}^e$, $\mathbf{R}^e$ is the sequence representation of evidence.

To get high parallelizability of attention, multi-head attention first linearly projects queries, keys, and values j times by different linear projections and then j projections perform the scaled dot-product attention in parallel. Finally, these results of attention are concatenated and once again projected to get the new representation. Formally, the multi-head attention can be formulated as:

$$\mathrm{head}_i = \mathbf{Attention}(\mathbf{Q}\mathbf{W}_i^Q, \mathbf{K}\mathbf{W}_i^K, \mathbf{V}\mathbf{W}_i^V) \quad (5)$$

$$\begin{aligned}\mathbf{O}' &= \mathbf{MultiHead}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) \\ &= [\mathrm{head}_1; \mathrm{head}_2; ...; \mathrm{head}_j]\mathbf{W}^o \end{aligned} \quad (6)$$

where $\mathbf{W}_i^Q \in \mathbb{R}^{2h \times D}$, $\mathbf{W}_i^K \in \mathbb{R}^{2h \times D}$, $\mathbf{W}_i^V \in \mathbb{R}^{2h \times D}$, and $\mathbf{W}^o \in \mathbb{R}^{2h \times 2h}$ are trainable parameters and D is $2h/j$.

Subsequently, co-attention networks pass a feed forward network (FFN) for adding non-linear features while scale-invariant features, which contains a single hidden layer with an ReLU.

$$\mathbf{O} = \mathbf{FFN}(\mathbf{O}') = \mathbf{max}(0, \mathbf{O}'\mathbf{W}_1 + b_1)\mathbf{W}_2 + \mathbf{b}_2 \quad (7)$$

where $\mathbf{W}_1$, $\mathbf{W}_2$, $\mathbf{b}_1$, and $\mathbf{b}_2$ are the learned parameters. $\mathbf{O} = \mathbf{C}$ and $\mathbf{O} = \mathbf{E}$ are output vectors of two

self-attention networks aiming at the claim and the evidence, respectively.

Finally, to fully integrate evidence and claim, we adopt the absolute difference and element-wise product to fuse the vectors $\mathbf{E}$ and $\mathbf{C}$ (Wu et al., 2019).

$$\mathbf{EC} = [\mathbf{E}; |\mathbf{E} - \mathbf{C}|; \mathbf{E} \odot \mathbf{C}; \mathbf{C}] \quad (8)$$

where $\odot$ denotes element-wise multiplication operation.

3.2.3 Output Layer

As the last layer, softmax function emits the prediction of probability distribution by the equation:

$$\mathbf{p} = \mathbf{softmax}(\mathbf{W}_p\mathbf{EC} + \mathbf{b}_p) \quad (9)$$

We train the model to minimize cross-entropy error for a training sample with ground-truth label $\mathbf{y}$:

$$\mathbf{Loss} = -\sum \mathbf{y} log \mathbf{p} \quad (10)$$

The training process of DTCA is presented in Algorithm 1 of Appendix B.

4 Experiments

As the key contribution of this work is to verify claims accurately and offer evidence as explanations, we design experiments to answer the following questions:

- **RQ1:** Can DTCA achieve better performance compared with the state-of-the-art models?
- **RQ2:** How do decision conditions in the decision tree affect model performance (to say, the interpretability of evidence selection process)?
- **RQ3:** Can DTCA make verification results easy-to-interpret by evidence and find false parts of claims?

4.1 Datasets

To evaluate our proposed model, we use two widely used datasets, i.e., RumourEval (Derczynski et al., 2017) and PHEME (Zubiaga et al., 2016). **Structure.** Both datasets respectively contain 325 and 6,425 Twitter conversation threads associated with different newsworthy events like Charlie Hebdo, the shooting in Ottawa, etc. A thread consists of a claim and a tree of comments (a.k.a. responses) expressing their opinion towards the claim. **Labels.** Both datasets have the same labels, i.e., true, false, and unverified. Since our goal is to verify whether a claim is true or false, we filter out unverified tweets. Table 1 gives statistics of the two datasets.

Subset	Veracity	RumourEval		PHEME	
		#posts	#comments	#posts	#comments
Training	True	83	1,949	861	24,438
	False	70	1,504	625	17,676
	Total	153	3,453	1,468	42,114
Validation	True	10	101	95	1,154
	False	12	141	115	1,611
	Total	22	242	210	2,765
Testing	True	9	412	198	3,077
	False	12	437	219	3,265
	Total	21	849	417	6,342

Table 1: Statistics of the datasets.

In consideration of the imbalance label distributions, besides accuracy (A), we add precision (P), recall (R) and F1-score (F1) as evaluation metrics for DTCA and baselines. We divide the two datasets into training, validation, and testing subsets with proportion of 70%, 10%, and 20% respectively.

4.2 Settings

We turn all hyper-parameters on the validation set and achieve the best performance via a small grid search. For hyper-parameter configurations, (1) in DTE, the change range of semantic similarity, the credibility of reviewers, and the credibility of comments respectively belong to [0, 0.8], [0, 0.8], and [0, 0.7]; (2) in CaSa, word embedding size d is set to 768; the size of LSTM hidden states h is 120; attention heads and blocks are 6 and 4 respectively; the dropout of multi-head attention is set to 0.8; the initial learning rate is set to 0.001; the dropout rate is 0.5; and the mini-batch size is 64.

4.3 Performance Evaluation (RQ1)

4.3.1 Baselines

SVM (Derczynski et al., 2017) is used to detect fake news based on manually extracted features.

CNN (Chen et al., 2017) adopts different window sizes to obtain semantic features similar to n-grams for rumor classification.

TE (Guacho et al., 2018) creates article-by-article graphs relying on tensor decomposition with deriving article embeddings for rumor detection.

DeClarE (Popat et al., 2018) presents attention networks to aggregate signals from external evidence articles for claim verification.

TRNN (Ma et al., 2018b) proposes two tree-structured RNN models based on top-down and down-top integrating semantics of structure and content to detect rumors. In this work, we adopt the top-down model with better results as the baseline.

Dataset	Measure	SVM	CNN	TE	DeClarE	TRNN	MTL-LSTM	Bayesian-DL	Sifted-MTL	Ours
RumourEval	A (%)	71.42	61.90	66.67	66.67	76.19	66.67	80.95	81.48	**82.54**
	P (%)	66.67	54.54	60.00	58.33	70.00	57.14	77.78	72.24	**78.25**
	R (%)	66.67	66.67	66.67	77.78	77.78	**88.89**	77.78	86.31	85.60
	F1 (%)	66.67	59.88	63.15	66.67	73.68	69.57	77.78	78.65	**81.76**
PHEME	A (%)	72.18	59.23	65.22	67.87	78.65	74.94	80.33	81.27	**82.46**
	P (%)	78.80	56.14	63.05	64.68	77.11	68.77	78.29	73.41	**79.08**
	R (%)	75.75	64.64	64.64	71.21	78.28	87.87	79.29	**88.10**	86.24
	F1 (%)	72.10	60.09	63.83	67.89	77.69	77.15	78.78	80.09	**82.50**

Table 2: The performance comparison of DTCA against the baselines.

MTL-LSTM (Kochkina et al., 2018) jointly trains rumor detection, claim verification, and stance detection tasks, and learns correlations among these tasks for task learning.

Bayesian-DL (Zhang et al., 2019) uses Bayesian to represent the uncertainty of prediction of the veracity of claims and then encodes responses to update the posterior representations.

Sifted-MTL (Wu et al., 2019) is a sifted multi-task learning model that trains jointly fake news detection and stance detection tasks and adopts gate and attention mechanism to screen shared features.

4.3.2 Results of Comparison

Table 2 shows the experimental results of all compared models on the two datasets. We observe that:

- SVM integrating semantics from claim content and comments outperforms traditional neural networks only capturing semantics from claim content, like CNN and TE, with at least 4.75% and 6.96% boost in accuracy on the two datasets respectively, which indicates that semantics of comments are helpful for claim verification.
- On the whole, most neural network models with semantic interaction between comments and claims, such as TRNN and Bayesian-DL, achieve from 4.77% to 9.53% improvements in accuracy on the two datasets than SVM without any interaction, which reveals the effectiveness of the interaction between comments and claims.
- TRNN, Bayesian-DL, and DTCA enable claims and comments to interact, but the first two models get the worse performance than DTCA (at least 1.06% and 1.19% degradation in accuracy respectively). That is because they integrate all comments indiscriminately and might introduce some noise into their models, while DTCA picks more valuable comments by DTE.
- Multi-task learning models, e.g. MTL-LSTM and Sifted-MTL leveraging stance features show at most 3.29% and 1.86% boosts in recall than DTCA on the two datasets respectively, but they also bring out noise, which achieve from 1.06% to 21.11% reduction than DTCA in the other three metrics. Besides, DTCA achieves 3.11% and 2.41% boosts than the latest baseline (sifted-MTL) in F1-score on the two datasets respectively. These elaborate the effectiveness of DTCA.

4.4 Discussions

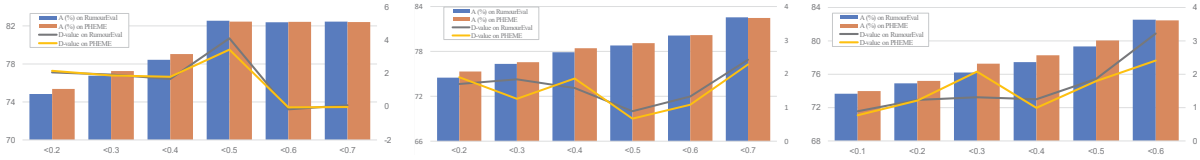
4.4.1 The impact of comments on DTCA

In Section 4.3, we find that the use of comments can improve the performance of models. To further investigate the quantitative impact of comments on our model, we evaluate the performance of DTCA and CaSa with 0%, 50%, and 100% comments. The experimental results are shown in Table 3. We gain the following observations:

- Models without comment features present the lowest performance, decreasing from 5.08% to 9.76% in accuracy on the two datasets, which implies that there are a large number of veracity-indicative features in comments.
- As the proportion of comments expands, the performance of models is improved continuously. However, the rate of comments for CaSa raises from 50% to 100%, the boost is not significant, only achieving 1.44% boosts in accuracy on RumourEval, while DTCA obtains better performance, reflecting 3.90% and 3.28% boosts in accuracy on the two datasets, which fully proves that DTCA can choose valuable comments and ignore unimportant comments with the help of DTE.

4.4.2 The impact of decision conditions of DTE on DTCA (RQ2)

To answer RQ2, we analyze the changes of model performance under different decision conditions.



(a) Different semantic similarity (b) Different credibility of reviewers (c) Different credibility of comments

Figure 3: The accuracy comparison of DTCA in different decision conditions. Broken lines represent the performance difference (D-value) between the current decision condition and the previous decision condition.

		A	P	R	F1
No (0%) Comments					
RumourEval	CaSa	72.78	67.03	72.87	69.83
	DTCA	72.78	67.03	72.87	69.83
PHEME	CaSa	73.21	71.26	74.74	72.96
	DTCA	73.21	71.26	74.74	72.96
50% Comments					
RumourEval	CaSa	76.42	70.21	76.78	73.35
	DTCA	78.64	73.43	80.06	76.60
PHEME	CaSa	77.65	74.18	78.11	76.09
	DTCA	79.18	75.24	80.66	77.86
All (100%) Comments					
RumourEval	CaSa	77.86	71.92	79.24	75.40
	DTCA	82.54	78.25	85.60	81.76
PHEME	CaSa	79.85	75.06	80.35	77.61
	DTCA	82.46	79.08	86.24	82.50

Table 3: The performance comparison of models on different number of comments.

Different decision conditions can choose different comments as evidence to participate in the model learning. According to the performance change of the model verification, we are capable of well explaining the process of evidence selection through decision conditions. Specifically, we measure different values (interval [0, 1]) as thresholds of decision conditions so that DTE could screen different comments. Figure 3(a), (b), and (c) respectively present the influence of semantic similarity ($simi$), the credibility of reviewers (r_cred), and the credibility of comments (c_cred) on the performance of DTCA, where the maximum thresholds are set to 0.7, 0.7, and 0.6 respectively because there are few comments when the decision threshold is greater than these values. We observe that:

- When $simi$ is less than 0.4, the model is continually improved, where the average performance improvement is about 2 % (broken lines) on the two datasets when $simi$ increases by 0.1. Especially, DTCA earns the best performance when $simi$ is set to 0.5 ($<$0.5), while it is difficult to improve performance after that. These exemplify that DTCA can provide more credibility features under appropriate semantic similarity for verification.

- DTCA continues to improve with the increase of r_cred, which is in our commonsense, i.e., the more authoritative people are, the more credible their speech is. Analogously, DTCA boosts with the increase of c_cred. These show the reasonability of the terms of both the credibility of reviewers and comments built by meta data.

- When $simi$ is set to 0.5 ($<$0.5), r_cred is 0.7 ($<$0.7), c_cred is 0.6 ($<$0.6), DTCA wins the biggest improvements, i.e., at least 3.43%, 2.28%, and 2.41% on the two datasets respectively. At this moment, we infer that comments captured by the model contain the most powerful evidence for claim verification. This is, the optimal evidence is formed under the conditions of moderate semantic similarity, high reviewer credibility, and higher comment credibility, which explains the selection process of evidence.

4.4.3 Explainability Analysis (RQ3)

To answer RQ3, we visualize comments (evidence) captured by DTE and the key semantics learned by CaSa when the training of DTCA is optimized. Figure 4 depicts the results based on a specific sample in PHEME, where at the comment level, red arrows represent the captured evidence and grey arrows denote the unused comments; at the word level, darker shades indicate higher weights given to the corresponding words, representing higher attention. We observe that:

- In line with the optimization of DTCA, the comments finally captured by DTE contain abundant evidence that questions the claim, like 'presumably Muslim? How do you arrive at that?', 'but it should be confirmed 1st before speculating on air.', and 'false flag', to prove the falsity of the claim (the label of the claim is false), which indicates that DTCA can effectively discover evidence to explain the results of claim verification.

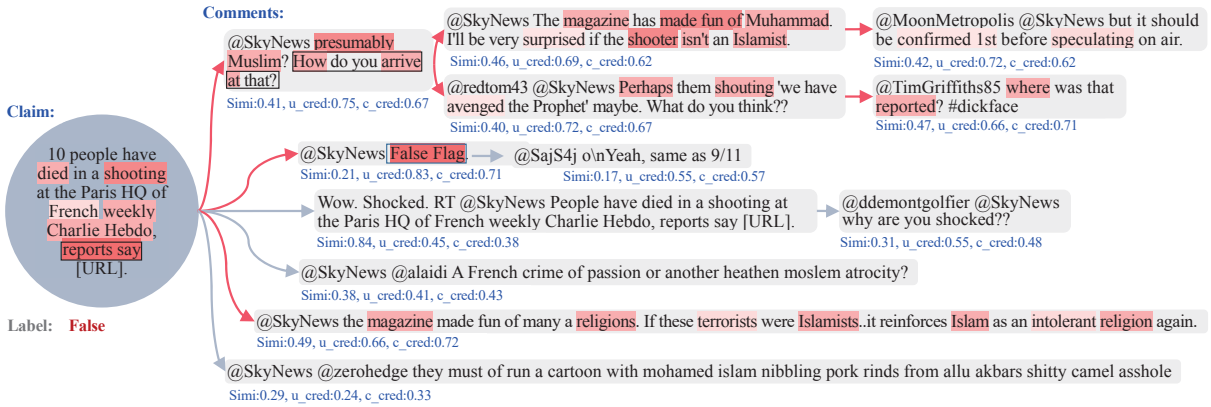


Figure 4: The visualization of a sample (labeled false) in PHEME by DTCA, where the captured evidence (red arrows) and the specific values of decision conditions (blue) are presented by DTE, and the attention of different words (red shades) is obtained by CaSa.

Additionally, there are common characteristics in captured comments, i.e., moderate semantic similarity (interval [0.40, 0.49]), high reviewer credibility (over 0.66), and high comment credibility (over 0.62). For instance, the values of the three characteristics of evidence '@TimGriffiths85 where was that reported? #dickface' are 0.47, 0.66, and 0.71 respectively. These phenomena explain that DTCA can give reasonable explanations to the captured evidence by decision conditions of DTE, which visually reflects the interpretability of DTCA method itself.

- At the word level, the evidence-related words 'presumably Muslim', 'made fun of', 'shooter', and 'isn't Islamist' in comments receive higher weights than the evidence-independent words 'surprised', 'confirmed 1st' and 'speculating', which illustrates that DTCA can earn the key semantics of evidence. Moreover, 'weekly Charlie Hebdo' in the claim and 'Islamist' and 'Muhammad' in comments are closely focused, which is related to the background knowledge, i.e., weekly Charlie Hebdo is a French satirical comic magazine which often publishes bold satire on religion and politics. 'report say' in claim is queried in the comments, like 'How do you arrive at that?' and 'false flag'. These visually demonstrate that DTCA can uncover the questionable and even false parts in claims.

4.4.4 Error Analysis

Table 4 provides the performance of DTCA under different claims with different number of comments. We observe that DTCA achieves the satisfactory performance in claims with more than 8 comments, while in claims with less than 8 comments, DTCA does not perform well, underperforming its best performance by at least 4.92% and 3.10% in accuracy on the two datasets respectively. Two reasons might explain the issue: 1) The claim with few comments has limited attention, and its false parts are hard to be found by the public; 2) DTCA is capable of capturing worthwhile semantics from multiple comments, but it is not suitable for verifying claims with fewer comments.

Claims	Datasets	A (%)	P (%)	R (%)	F1 (%)
Claims with less than 3 comments	RumourEval	73.42	68.21	73.50	70.76
	PHEME	74.10	72.45	75.32	73.86
Claims with comments $\in$ [3, 8]	RumourEval	75.33	69.16	75.07	71.99
	PHEME	77.26	74.67	79.03	76.79
Claims with more than 8 comments	RumourEval	80.25	75.61	83.45	79.34
	PHEME	80.36	75.52	84.33	79.68

Table 4: The performance comparison of DTCA under different claims with different number of comments.

5 Conclusion

We proposed a novel framework combining decision tree and neural attention networks to explore a transparent and interpretable way to discover evidence for explainable claim verification, which constructed decision tree model to select comments with high credibility as evidence, and then designed co-attention networks to make the evidence and claims interact with each other for unearthing the false parts of claims. Results on two public datasets demonstrated the effectiveness and explainability of this framework. In the future, we will extend the proposed framework by considering more context (meta data) information, such as time, storylines, and comment sentiment, to further enrich our ex-

plainability.

Acknowledgments

The research work is supported by National Key Research and Development Program in China (2019YFB2102300); The World-Class Universities (Disciplines) and the Characteristic Development Guidance Funds for the Central Universities of China (PY3A022); Ministry of Education Fund Projects (18JZD022 and 2017B00030); Shenzhen Science and Technology Project (JCYJ20180306170836595); Basic Scientific Research Operating Expenses of Central Universities (ZDYF2017006); Xi'an Navinfo Corp.& Engineering Center of Xi'an Intelligence Spatial-temporal Data Analysis Project (C2020103); Beilin District of Xi'an Science & Technology Project (GX1803). We would like to thank the anonymous reviewers for their insightful comments.

References

Oluwaseun Ajao, Deepayan Bhowmik, and Shahrzad Zargari. 2019. Sentiment aware fake news detection on online social networks. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2507–2511. IEEE.

Sinan Aral and Dean Eckles. 2019. Protecting elections from social media manipulation. *Science*, 365(6456):858–861.

Ramy Baly, Georgi Karadzhov, Dimitar Alexandrov, James Glass, and Preslav Nakov. 2018a. Predicting factuality of reporting and bias of news media sources. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3528–3539.

Ramy Baly, Mitra Mohtarami, James Glass, Lluís Màrquez, Alessandro Moschitti, and Preslav Nakov. 2018b. Integrating stance detection and fact checking in a unified corpus. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 21–27.

Xu Chen, Yongfeng Zhang, Hongteng Xu, Yixin Cao, Zheng Qin, and Hongyuan Zha. 2018. Visually explainable recommendation. *arXiv preprint arXiv:1801.10288*.

Yi-Chin Chen, Zhao-Yang Liu, and Hung-Yu Kao. 2017. Ikm at semeval-2017 task 8: Convolutional neural networks for stance detection and rumor verification. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 465–469.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.

Leon Derczynski, Kalina Bontcheva, Maria Liakata, Rob Procter, Geraldine Wong Sak Hoi, and Arkaitz Zubiaga. 2017. Semeval-2017 task 8: Rumoureval: Determining rumour veracity and support for rumours. *arXiv preprint arXiv:1704.05972*.

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.

Mengnan Du, Ninghao Liu, and Xia Hu. 2018. Techniques for interpretable machine learning. *arXiv preprint arXiv:1808.00033*.

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.

Wolfgang Dvořák and Stefan Woltran. 2019. Complexity of abstract argumentation under a claim-centric view. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 2801–2808.

Alex Graves, Santiago Fernández, and Jürgen Schmidhuber. 2005. Bidirectional lstm networks for improved phoneme classification and recognition. In *International Conference on Artificial Neural Networks*, pages 799–804. Springer.

Nir Grinberg, Kenneth Joseph, Lisa Friedland, Briony Swire-Thompson, and David Lazer. 2019. Fake news on twitter during the 2016 us presidential election. *Science*, 363(6425):374–378.

Tommi Gröndahl and N Asokan. 2019. Text analysis in adversarial settings: Does deception leave a stylistic trace? *ACM Computing Surveys (CSUR)*, 52(3):45.

Gisel Bastidas Guacho, Sara Abdali, Neil Shah, and Evangelos E Papalexakis. 2018. Semi-supervised content-based detection of misinformation via tensor embeddings. In *2018 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 322–325. IEEE.

David Gunning. 2017. Explainable artificial intelligence (xai). *Defense Advanced Research Projects Agency (DARPA), nd Web*, 2.

Andreas Hanselowski, Hao Zhang, Zile Li, Daniil Sorokin, Benjamin Schiller, Claudia Schulz, and Iryna Gurevych. 2018. Ukp-athene: Multi-sentence textual entailment for claim verification. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 103–108.

Xiangnan He, Tao Chen, Min-Yen Kan, and Xiao Chen. 2015. Trirank: Review-aware explainable recommendation by modeling aspects. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pages 1661–1670. ACM.

Dhruv Khattar, Jaipal Singh Goud, Manish Gupta, and Vasudeva Varma. 2019. Mvae: Multimodal variational autoencoder for fake news detection. In *The World Wide Web Conference*, pages 2915–2921. ACM.

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.

Jing Ma, Wei Gao, Shafiq Joty, and Kam-Fai Wong. 2019. Sentence-level evidence embedding for claim verification with hierarchical attention networks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2561–2571.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2018a. Detect rumor and stance jointly by neural multi-task learning. In *Companion Proceedings of the The Web Conference 2018*, pages 585–593. International World Wide Web Conferences Steering Committee.

Jing Ma, Wei Gao, and Kam-Fai Wong. 2018b. 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.

Yixin Nie, Haonan Chen, and Mohit Bansal. 2019. Combining fact extraction and verification with neural semantic matching networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6859–6866.

Kashyap Popat, Subhabrata Mukherjee, Andrew Yates, and Gerhard Weikum. 2018. Declare: Debunking fake news and false claims using evidence-aware deep learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 22–32.

Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendorff, and Benno Stein. 2018. A stylometric inquiry into hyperpartisan and fake news. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 231–240.

Hannah Rashkin, Eunsol Choi, Jin Yea Jang, Svitlana Volkova, and Yejin Choi. 2017. Truth of varying shades: Analyzing language in fake news and political fact-checking. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2931–2937.

Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. 2017. Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models. *arXiv preprint arXiv:1708.08296*.

Yan Shan. 2016. How credible are online product reviews? the effects of self-generated and system-generated cues on source credibility evaluation. *Computers in Human Behavior*, 55:633–641.

Kai Shu, Limeng Cui, Suhang Wang, Dongwon Lee, and Huan Liu. 2019. defend: Explainable fake news detection. In *KDD*.

Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 19(1):22–36.

Grigori Sidorov, Alexander Gelbukh, Helena Gómez-Adorno, and David Pinto. 2014. Soft similarity and soft cosine measure: Similarity of features in vector space model. *Computación y Sistemas*, 18(3):491–504.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification. *arXiv preprint arXiv:1803.05355*.

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.

Svitlana Volkova, Kyle Shaffer, Jin Yea Jang, and Nathan Hodas. 2017. Separating facts from fiction: Linguistic models to classify suspicious and trusted news posts on twitter. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 647–653.

Xiang Wang, Xiangnan He, Fuli Feng, Liqiang Nie, and Tat-Seng Chua. 2018. Tem: Tree-enhanced embedding model for explainable recommendation. In *Proceedings of the 2018 World Wide Web Conference*, pages 1543–1552. International World Wide Web Conferences Steering Committee.

Lianwei Wu and Yuan Rao. 2020. Adaptive interaction fusion networks for fake news detection. ArXiv preprint arXiv:2004.10009.

Lianwei Wu, Yuan Rao, Haolin Jin, Ambreen Nazir, and Ling Sun. 2019. Different absorption from the same sharing: Sifted multi-task learning for fake news detection. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4636–4645.

Lianwei Wu, Yuan Rao, Ambreen Nazir, and Haolin Jin. 2020. Discovering differential features: Adversarial learning for information credibility evaluation. *Information Sciences*, 516:453–473.

Qiang Zhang, Aldo Lipani, Shangsong Liang, and Emine Yilmaz. 2019. Reply-aided detection of misinformation via bayesian deep learning. In *The World Wide Web Conference*, pages 2333–2343. ACM.

Yongfeng Zhang and Xu Chen. 2018. Explainable recommendation: A survey and new perspectives. *arXiv preprint arXiv:1804.11192*.

Jie Zhou, Xu Han, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2019. Gear: Graph-based evidence aggregating and reasoning for fact verification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 892–901.

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.

	Element	Standard of criterion	Credibility score
Whether the elements exist or not	verified	True	3
	geo	True	3
	screen name	True	1
	profile image	True	2
The value of elements	followers	[0, 100)	2
		[100, 500)	5
		[500, ∞)	10
	friends	[0, 100)	1
		[100, 200)	2
		[200, ∞)	5
	favourites	[0, 100)	2
		[100, 200)	3
		[200, ∞)	5

Table 5: The credibility score of reviewers

	Element	Standard of criterion	Credibility score
Whether the elements exist or not	geo	True	3
	source	True	3
	favorite the comment	True	1
The value of elements	the number of favorites	[0, 100)	5
		[100, ∞)	7
	the length of content	[0, 10)	3
		[10, ∞)	7

Table 6: The credibility score of comments

A The Details of Decision Conditions

The paper has introduced the details of semantic similarity between claims and comments, here we introduce the details of the other two decision conditions.

Table 5 shows some scores of meta data related to reviewer credibility. In elements 'whether the elements exist or not', if the element is false, the score will be zero. The credibility score of reviewer (r_cred) is formulated as follows:

$$r\_cred = \frac{A1}{B1} \tag{11}$$

where A1 denotes the specific score accumulation of all metadata related to reviewer credibility and B1 means the total credibility score of reviewers.

Table 6 describes some credibility score of comments. Like the credibility score of reviewer, in elements 'whether the elements exist or not', if the element is false, the score will be zero. The credibility score of comments (c_cred) is as follows:

$$c\_cred = \frac{A2}{B2} \tag{12}$$

where A2 denotes the specific score accumulation of all metadata related to comment credibility and B2 means the total credibility score of comments.

B Algorithm of DTCA

The training procedure of DTCA is shown in Algorithm 1.

Algorithm 1: Training Procedure of DTCA.

Require: Dataset $S = \{C_i, R_i, y\}_1^T$ with T training samples, where C_i denotes one claim and R_i means tree comments, i.e., $R_i = r_1, r_2, ..., r_k$. Particularly, k is different for different claims; the thresholds of the three decision conditions are a_1, a_2, a_3, respectively; the evidence set E; model parameters Θ; learning rate ϵ.

Initial parameters;
Repeat
  For $i = 1$ to T **do**
    // Part 1: DTE
    A series of subtree sets S obtained by depth-first search of tree comments R_i, i.e., S=[S_1,S_2, ...,S_n];
    On each subtree S_i:
    For r_i in S_i **do**
      // The semantic similarity between
      // comments and claims
      If $simi(r_i, C_i) \in interval[a, b]$:
        $E = E + r_i$;
      // The credibility of reviewers
      If $r\_cred(r_i) > a_2$:
        $E = E + r_i$;
      // The credibility of comments
      If $c\_cred(r_i) > a_3$:
        $E = E + r_i$;
    End For
    By traversing all subtrees, the final evidence set E that meets the conditions is captured.
    //Part 2: CaSa
    Word embeddings of evidence E:
    $X^e = BERT\_embed(E)$;
    Word embeddings of claim C_i:
    $BERT\_embed(C_i)$;
    Get representations of evidence and claim by Eq. (1-3), i.e., R^e and R^c;
    Get R^e_{pool} by maximum pooling operation of R^e;
    Get deep interaction semantics C of claim concerned by evidence through integrating R^e_{pool} into self-attention networks by R^e_{pool} Eq. (4-7);
    Get deep interaction semantics E of evidence concerned by the claim through integrating C into self-attention networks by Eq. (4-7);
    Get fused vectors between evidence and claim by Eq. (8);
    Compute loss $L(\Theta)$ using Eq. (9,10);
    Compute gradient $\nabla(\Theta)$;
    Update model: $\Theta \leftarrow \Theta - \epsilon\nabla(\Theta)$;
  End For
Update parameters a_1, a_2, a_3;
Until $a_1 = 0.8$, $a_2 = 0.8$, and $a_3 = 0.7$.