# Integrating Semantic and Structural Information with Graph Convolutional Network for Controversy Detection

Anonymous ACL submission

## Abstract

Identifying controversial posts on social media is a fundamental task for mining public sentiment, assessing the influence of events, and alleviating the polarized views. However, existing methods fail to 1) effectively incorporate the semantic information from content-related posts; 2) preserve the structural information for reply relationship modeling; 3) properly handle posts from topics dissimilar to those in the training set. To overcome the first two limitations, we propose Topic-Post-Comment Graph Convolutional Network (TPC-GCN), which integrates the information from the graph structure and content of topics, posts, and comments for post-level controversy detection. As to the third limitation, we extend our model to Disentangled TPC-GCN (DTPC-GCN), to disentangle topic-related and topic-unrelated features and then fuse dynamically. Extensive experiments on two real-world datasets demonstrate that our models outperform existing methods. Analysis of the results and cases proves that our models can integrate both semantic and structural information with significant generalizability.

## 1 Introduction

Social media such as Reddit[1] and Chinese Weibo[2] has been a major platform where people can easily propagate their views. In the open and free circumstance, the views expressed by the posts often spark fierce discussion and raise controversy among the engaging users. These controversial posts provide a lens of public sentiment, which bring about several tasks such as news topic selection, influence assessment (Hessel and Lee, 2019), and alleviation of polarized views (Garimella et al., 2017). As a basis of all mentioned tasks, automatically identifying the controversial posts has attracted wide attention (Addawood et al., 2017; Coletto et al., 2017; Rethmeier et al., 2018; Hessel and Lee, 2019).

[1] https://www.reddit.com/
[2] https://weibo.com/

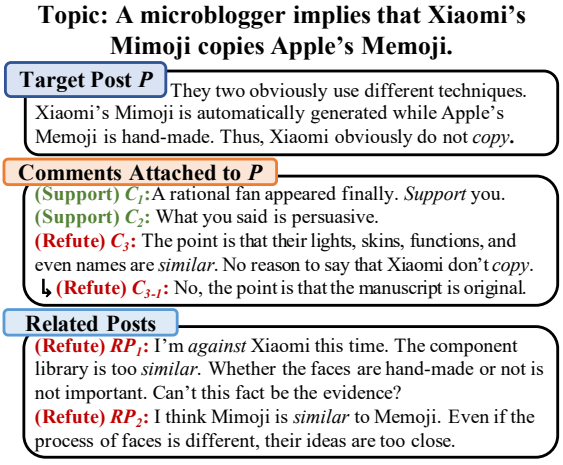


Figure 1: A controversial post $P$ about whether Xiaomi's Mimoji copies Apple's Memoji. These **Supports** and **Refutations** are to either their respective parent comments or $P$.

This work focuses on post-level controversy detection on social media, i.e., to classify if a post is controversial or non-controversial. According to (Coletto et al., 2017), a controversial post has debatable content and expresses an idea or an opinion which generates an argument in the responses, representing opposing opinions in favor or in disagreement with the post. In practice, the responses of a target post (the post to be judged) generally come from two sources, i.e., the comments attached to the post and other content-related posts. Figure 1 shows an example where the target post $P$ expresses that Xiaomi's Mimoji do not copy Apple's Memoji. We can see that: 1) The comments show more supports and fewer refutes to $P$, which raises a small controversy. However, the related posts show extra refutations and enhance the controversy of $P$. 2) $C_{3-1}$ expresses refutation literally, but it actually supports $P$ because in the comment tree, it refutes $C_3$, a refuting comment to $P$. 3) There exist two kinds of semantic clues for detection, topic-related and topic-unrelated clues. For example, *support* and *against* is unrelated to this topic, while

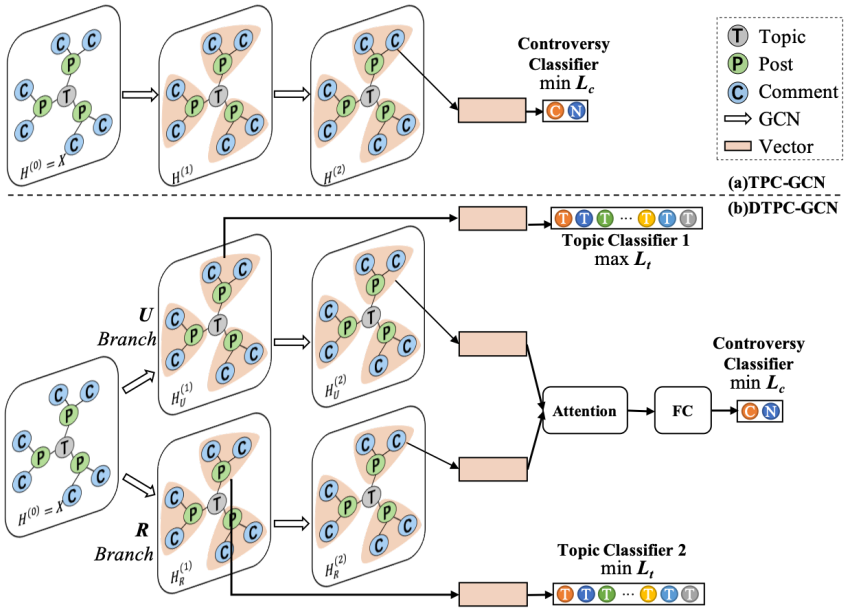


Figure 2: Architecture of (a) Topic-Post-Comment Graph Convolutional Network (TPC-GCN). (b) Disentangled TPC-GCN (DTPC-GCN). The upper post in the TPC graph is taken as an example to illustrate the methods. $H_B^{(l)}$ is the representation matrix, containing all node vectors in the $l$-th layer of Branch $B$. $X$ is the initial representation. $L_c$ and $L_t$ refer to controversy classification loss and topic classification loss respectively. FC means fully connected layer.

*copy* and *similar* are topic-related. Topic-related clues can help identify posts in a similar topic, but how effective they are for those in dissimilar topics depends on the specific situation. Therefore, to comprehensively evaluate the controversy of a post, the information from both the comments and related posts should be integrated properly on semantic and structure level.

Existing methods detecting controversy on social media have exploited the semantic feature of the target post and its comments as well as structural feature. However, three drawbacks limit their performance: 1) These methods ignore the role of the related posts on the same topic in providing extra supports or refutations on the target post. Only exploiting the information from comments is insufficient. 2) These methods use statistical structure-based features which cannot model the reply-structure relationships (like $P$-$C_1$ and $C_3$-$C_{3-1}$ in Figure 1). The stances of some comments may be misunderstood by the model (like $C_{3-1}$). 3) These methods tend to capture topic-related features that are not shared among different topics with directly using information of content (Wang et al., 2018). The topic-related features can be helpful when the testing post is from a topic similar to those in the training set but would hurt the detection otherwise.

Recently, graph convolutional networks have achieved great success in many areas (Marcheggiani et al., 2018; Ying et al., 2018; Yao et al., 2019; Li and Goldwasser, 2019) due to its ability to encode both local graph structure and features of node (Kipf and Welling, 2017). To overcome the first two drawbacks of existing works, we propose a Topic-Post-Comment Graph Convolutional Network (TPC-GCN) (see Figure 2a) that integrates the information from the graph structure and content of topics, posts, and comments for post-level controversy detection. First, we create a TPC graph to describe the relationship among topics, posts, and comments. To preserve the reply-structure information, we connect each comment node with the post/comment node it replies to. To include the information from related posts, we connect each post node with its topic node. Then, a GCN model is applied to learn node representation with content and reply-structure information fused. Finally, the updated vectors of a post and its comments are fused to predict the controversy.

TPC-GCN is mainly for detection in intra-topic mode, i.e., topics of testing posts appear in the training set, for it cannot overcome the third drawback. We thus extend a two-branch version of TPC-GCN named Disentangled TPC-GCN (DTPC-GCN) (see Figure 2b) for inter-topic mode (no testing posts

are from the topics in the training set). We use a TPC-GCN in each branch, but add an auxiliary task, topic classification. The goals of the two branches for the auxiliary task are opposite to disentangle the topic-related and topic-unrelated features. The disentangled features can be dynamically fused according to the content of test samples with attention mechanism for final decision. Extensive experiments demonstrate that our model outperforms existing methods and can exploit features dynamically and effectively. We plan to release the dataset[3] after double-blind review. The main contributions of this paper are as follows:

1. We propose two novel GCN-based models, TPC-GCN and DTPC-GCN, for post-level controversy detection. The models can integrate the information from the structure and content of topics, posts, and comments, especially the information from the related posts and reply tree. Specially, DTPC-GCN can further disentangle the topic-related features and topic-unrelated features for inter-topic detection.

2. We build a Chinese dataset for controversy detection, consisting of 5,676 posts collected from Chinese Weibo, each of which are manually labeled as controversial or non-controversial. To the best of our knowledge, this is the first released Chinese dataset for controversy detection.

3. Experiments on two real-world datasets demonstrate that the proposed models can effectively identify the controversial posts and outperform existing methods in terms of performance and generalization.

## 2 Related Work

Controversy detection on the Internet have been studied on both web pages and social media. Existing works detecting controversy on web pages mostly aims at identifying controversial articles in Wikipedia. Early methods are mainly based on statistical features, such as revision times (Kittur et al., 2007), edit history (Vuong et al., 2008; Yasseri et al., 2012; Rad and Barbosa, 2012) and dispute tag (Dori-Hacohen and Allan, 2015). Others incorporate the collaboration-network-based features, sentiment-based features (Vuong et al., 2008; Wang and Cardie, 2014), and semantic features (Linmans et al., 2018). As to the common web pages, existing works exploit the controversy on Wikipedia (Awadallah et al., 2012; Dori-Hacohen and Allan, 2013, 2015; Jang et al., 2016) and user comments (Choi et al., 2010; Tsytsarau et al., 2010) for detection.

Unlike the web pages, social media contains more diverse topics and more fierce discussion among users, which makes controversy detection on social media more challenging. Early studies assume that a topic has its intrinsic controversy, and focus on topic-level controversy detection. Popescu and Pennacchiotti (2010) detect controversial snapshots (consisting of many tweets referring to a topic) based on Twitter-based and external-knowledge features. Garimella et al. (2018) build graphs based on a Twitter topic, such as retweeting graph and following graph, and then apply graph partitioning to measure the extent of controversy. However, topic-level detection is rough, because there exists non-controversial posts in a controversial topic and vice versa. Recent works focus on post-level controversy detection by leveraging language features, such as emotional and topic-related phrases (Rethmeier et al., 2018), emphatic features, Twitter-specific features (Addawood et al., 2017). Other graph-based methods exploit the features from the following graph and comment tree (Coletto et al., 2017; Hessel and Lee, 2019). The limitations of current post-level works are that they do not effectively integrate the information from content and reply-structure, and ignore the role of posts in the same topic. Moreover, the difference between intra-topic and inter-topic mode is not realized. Only Hessel and Lee (2019) deal with topic transfer, but they train on each topic and test on others to explore the transferability, which is not suitable in practice.

## 3 Methodology

In this section, we introduce the Topic-Post-Comment Graph Convolutional Network (TPC-GCN) and its extension Disentangled TPC-GCN (DTPC-GCN), as shown in Figure 2. We first introduce the TPC graph construction and then detail the two models.

### 3.1 TPC Graph Construction

To model the paths of message passing among topics, posts, and comments, we first construct a topic-

[3] https://bit.ly/2RAUAy0

post-comment graph $G = (V, E)$ for target posts, where $V$ and $E$ denote the set of nodes and edges respectively. First, to preserve the post-comment and inter-comment relationship, we incorporate the comment tree, each comment node of which is connected with the post/comment node it replies to. Then, to facilitate the posts capturing information from related posts in the same topic that proved helpful in Section 1, we connect each post with its topic. The topic node can be regarded as a *hub* node to integrate and interchange the information. Another way is to connect post nodes in a topic pairwise, but the complexity will be high. Note that the concept *topic* here is not necessarily provided by the platform, such as the subreddit on Reddit and the hashtag (#) on Weibo. When topics are not provided, algorithms for text-based clustering can be used to construct a topic with related posts (Nematzadeh et al., 2019).

In $G$, each node may represent a topic, a post, or a comment and each edge may represent topic-post, post-comment, or comment-comment connection. We initially represent each node $v$ with an embedding vector $x$ of their text by using the pre-trained language model.

### 3.2 TPC-GCN

In this subsection, we detail the TPC-GCN, by first introducing the generic GCN and then our TPC-GCN model.

The GCN has been proved an efficient neural network that operates on a graph to encode both local graph structure and features of node (Kipf and Welling, 2017). The characteristic of GCN is consistent to our goal that integrates the semantic and structural information. In a GCN, each node is updated according to the aggregated information of its neighbor nodes and itself, so the learned representation can include information from both content and structure. For a node $v_i \in V$, the update rule in the message passing process is as follows:

$$h_i^{(l+1)} = \sigma\left(\sum_{j \in N_i} g\left(h_i^{(l)}, h_j^{(l)}\right) + b^{(l)}\right) \quad (1)$$

where $h_i^{(l)}$ is the hidden state of node $v_i$ in the $l$-th layer of a GCN and $N_i$ is the neighbor set of node $v_i$ with itself included. Incoming messages from $N_i$ are transformed by the function $g$ and then pass through the activation function $\sigma$ (such as `ReLU`) to output new representation for each node. $b^{(l)}$ is the bias term. Following Kipf and Welling (2017), we use a linear transform function $g(h_i^{(l)}, h_j^{(l)}) = W^{(l)} h_j$, where $W^{(l)}$ is a learnable weight matrix. Based on node-wise Equation 1, layer-wise propagation rule can be written as the following form:

$$H^{(l+1)} = \sigma\left(\hat{A} H^{(l)} W^{(l)} + B^{(l)}\right) \quad (2)$$

where $H^{(l)}$ contains all node vectors in the $l$-th layer and $\hat{A}$ is the normalized adjacency matrix with inserted self-loops. $W^{(l)}$ is the weight matrix and $B^{(l)}$ is the broadcast bias term.

In TPC-GCN (see Figure 2a), we input the matrix consisting of $N$ $d$-dimensional embedding vectors $H^{(0)} = X \in \mathbb{R}^{N \times d}$ to a two-layer GCN to obtain the representation after message passing $H^{(2)}$. Next, the vector of each post node $i$ and its attached comment nodes are averaged to be the fusion vector $f_i$ of the post. Finally, we apply a `softmax` function to the fusion vectors for the controversy probability of each post. The cross entropy is the loss function:

$$L_c = -\frac{1}{N} \sum_i \left( (1 - y_i^c) \log (1 - p_i^c) + y_i^c \log (p_i^c) \right) \quad (3)$$

where $y_i^c$ is a label with 1 representing *controversial* and 0 representing the *non-controversial*, $p_i^c$ is the predicted probability that the $i$-th post is controversial, and $N$ is the size of training set. The limit of TPC-GCN is that the representation tends to be topic-related as Section 1 said. The limited generalizability of TPC-GCN makes it more suitable for intra-topic detection, instead of inter-topic detection.

### 3.3 Disentangled TPC-GCN

Intuitively, topic-unrelated features are more effective when testing on the posts from unknown topics (inter-topic detection). However, topic-related features can help when unknown topics are similar to the topics in the training set. Therefore, both of topic-related and topic-unrelated features are useful, but their weights vary from sample to sample. This indicates that the two kinds of features should be disentangled and then dynamically fused. Based on the above analysis, we propose the extension of TPC-GCN, Disentangled TPC-GCN (see Figure 2b), for inter-topic detection. DTPC-GCN consists of two parts: the two-branch multi-task architecture for disentanglement, and attention mechanism for dynamic fusion.

**Two-branch Multi-task Architecture** To obtain the topic-related and topic-unrelated features at the same time, we use two branches of TPC-GCN with multi-task architecture, denoted as $R$ for topic-related branch and $U$ for topic-unrelated one. In both $R$ and $U$, an auxiliary task, topic classification, is introduced to guide the learning of representation oriented by the topic.

For each branch, we first train the first layer of GCN with the topic classification task. The input of the topic classifier is fusion vectors from $H^{(1)}$ which are obtained with the same process of $f_i$ in TPC-GCN. The cross entropy is used as the loss function:

$$L_t = -\frac{1}{N}\sum_k \sum_i y_{ik}^t \log(p_{ik}^t) \quad (4)$$

where $y_{ik}^t$ is a label with 1 representing the ground-truth topic and 0 representing the incorrect topic class, $p_{ik}^t$ is the predicted probability of the $i$-th post belonging to the $k$-th topic, and $N$ is the size of training set. The difference between $R$ and $U$ is that we minimize $L_t$ in Branch $R$ to obtain topic-distinctive features, but maximize $L_t$ in Branch $U$ to obtain topic-confusing features.

Then we include the second layer of GCN and train on two tasks, i.e., topic and controversy classification, for each branch individually. Branch $U$ and $R$ are expected to evaluate controversy effectively with different features in terms of the relationship with the topics.

**Attention Mechanism** After the individual training, Branch $U$ and $R$ are expected to capture the topic-related and topic-unrelated features respectively. We further fuse the features from the two branches dynamically. Specifically, we freeze the parameters of $U$ and $R$, and further train the dynamic fusion component. For the weighted combination of fusion vectors $f_U$ and $f_R$ from the two branches, we use the attention mechanism as follows:

$$\mathcal{F}(f_b) = v^T \tanh(W_{\mathcal{F}} f_b + b_{\mathcal{F}}), b \in \{U, R\} \quad (5)$$

$$\alpha_b = \frac{\exp(\mathcal{F}(f_b))}{\sum_{b\in\{U,R\}} \exp(\mathcal{F}(f_b))} \quad (6)$$

$$u = \sum_{b\in\{U,R\}} \alpha_b f_b \quad (7)$$

where $W_{\mathcal{F}}$ is the weight matrix and $b_{\mathcal{F}}$ is the bias term. $v^T$ is a transposed weight vector and $\mathcal{F}(\cdot)$ outputs the score of the input vector. The scores of features from Branch $U$ and $R$ are normalized via a `softmax` function as the branch weight. The weighted sum of the two fusion vectors $u$ is finally used for controversy classification. The loss function is the same as Equation 3.

| Number | Weibo | Reddit |
|---|---|---|
| Topics(Hashtags/Subreddits) | 49 | 6 |
| Controversial Posts | 1,992 | 7,515 |
| Non-controversial Posts | 3,684 | 7,518 |
| All Posts | 5,676 | 15,033 |
| Comments of Controversial Posts | 35,632 | 578,879 |
| Comments of Non-Controversial Posts | 34,565 | 1,461,697 |
| All Comments | 70,197 | 2,040,576 |

Table 1: Statistics of two datasets.

## 4 Experiment

In this section, we conduct experiments to compare our proposed models and other baseline models. Specifically, we mainly answer the following evaluation questions:

**EQ1:** Are TPC-GCN and DTPC-GCN able to improve the performance of controversy detection?

**EQ2:** How effective are different information in TPC-GCN, including the content of topics, posts, and comments as well as the topic-post-comment structure?

**EQ3:** Can DTPC-GCN learn disentangled features and dynamically fuse them for controversy detection?

### 4.1 Dataset

We perform our experiments on two real-world datasets in different languages. Table 1 shows the statistics of the two datasets. The details are as follows:

**Reddit Dataset** The Reddit dataset released by Hessel and Lee (2019) and Jason Baumgartner of `pushshift.io` is the only accessible English dataset for controversy detection of social media posts. This dataset contains six subreddits (which can be regarded as over-arching topics): AskMen, AskWomen, Fitness, LifeProTips, personalfinance, and relationships. Each post belongs to a subreddit and the number of attached comments is ensured to be over 30. The tree structure of the comments is also maintained. We use the comment data in the first hour after a post is published.

**Weibo Dataset** We built a Chinese dataset for controversy detection on Weibo in this work. We first manually selected 49 widely discussed, multi-

domain topics from July 2017 to August 2019 (see Appendix). Then, we crawled and preserved those posts with at least two comments. Here we rebuilt the comment tree according to the comment time and usernames due to the lack of officially-provided structure. Finally, annotators labeled the posts after being shown the definition and examples. In total, this dataset contains 1,992 controversial posts and 3,684 non-controversial posts, which indicates the imbalance in the real world. As far as we know, this is the first released dataset for controversy detection on Chinese social media. We use at most 15 comments of each post due to the computation limit.

In the intra-topic experiment: For Weibo dataset, we randomly divided with a ratio of 4:1:1 in each topic and merged them respectively across all topics. For Reddit dataset, we apply the data partition provided by the authors. The ratio is 3:1:1.

In the inter-topic experiments: For Weibo and Reddit dataset, we still divided with a ratio of 4:1:1, but on topic-level.

### 4.2 Implementation Details

In the (D)TPC-GCN model, each node is initialized with its textual content using the pre-trained BERT [4] and the padding size for each is 45. We only fine-tune the last layer, namely layer 11 of BERT for simplicity and then apply a dense layer with a `ReLU` activation function to reduce the dimensionality of representation from 768 to 300. In TPC-GCN, the sizes of hidden states of two GCN layer are 100 and 2, with `ReLU` for the first GCN layer. To avoid overfitting, a dropout layer is added between the two layers with a rate of 0.35. We apply a `softmax` function to the fusion vector for obtaining the controversy probability. In DTPC-GCN, the size of hidden states of the first and second GCN layers in each branch are 32 and 16. The dropout rate between two GCN layers in each branch is set to 0.4.

### 4.3 Baselines

To validate the effectiveness of our methods, we implemented several representative methods including content-based, structure-based and fusion methods as baselines.

**Content-based Methods**

We implement mainstream text classification models including **TextCNN** (Kim, 2014), **BiLSTM-Att** (bi-directional LSTM with attention) BiLSTM (Graves and Schmidhuber, 2005; Bahdanau et al., 2015), **BiGRU-Att** (bi-directional GRU with attention) (Cho et al., 2014),**BERT** (Devlin et al., 2019) (only fine-tune the last layer for simplicity). For a fair comparison, we concatenate the post and its attached comments together as the input, instead of feeding the post only.

**Structure-based Methods**

Considering that structure-based features of the post and its comment tree are rare and non-systematic in previous works, we integrate the plausible features in (Coletto et al., 2017) and (Hessel and Lee, 2019). As the latter paper does, we feed them into a series of classifiers and choose a best model for classification. We name the method **SFC**. For a post-comment graph, the feature set contains the average depth (average length of root-to-leaf paths), the maximum relative degree (the largest node degree divided by the degree of the root), C-RATE features (the logged reply time between the post and comments, or over pairs of comments), and C-TREE features (statistics in a comment tree, such as maximum depth/total comment ratio).

**Fusion Method**

The compared fusion method from (Hessel and Lee, 2019) aims to identify the controversial posts with semantic and structure information. They extract text features of topics, posts, and comments by BERT and structural feature including the C-RATE and C-TREE features mentioned above. In addition, publish time features are also exploited.

### 4.4 Performance Comparison

To answer **EQ1**, we compare the performance of proposed (D)TPC-GCN with mentioned baselines on the two datasets. The evaluation metrics include the macro average precision (Avg. P), macro average recall (Avg. R), macro average F1 score (Avg. F1), and accuracy (Acc.).Table 2 and 3 show the performance of all compared methods for intra-topic detection and inter-topic detection respectively.

In the intra-topic experiments, we can see that 1) TPC-GCN outperforms all compared methods on the two datasets. This indicates that our model can effectively detect controversy with a significant generalizability on different datasets. 2) The structure-based model, SFC, reports the low scores on the two datasets, indicating that the statistical structural information is insufficient to timely iden-

[4] https://github.com/google-research/bert

| Method | | Weibo Dataset | | | | Reddit Dataset | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Avg. P | Avg. R | Avg. F1 | Acc. | Avg. P | Avg. R | Avg. F1 | Acc. |
| **Content-based** | TextCNN | 72.80 | 68.49 | 69.08 | 72.83 | 56.58 | 56.33 | 55.92 | 56.33 |
| | BiLSTM-Att | 69.97 | 70.31 | 70.10 | 71.28 | 62.74 | 60.66 | 58.98 | 60.66 |
| | BiGRU-Att | 71.35 | 72.21 | 71.50 | 72.21 | 59.95 | 59.86 | 59.77 | 59.86 |
| | BERT | 72.17 | 72.72 | 72.37 | 73.35 | 60.80 | 60.80 | 60.80 | 60.80 |
| **Structure-based** | SFC | 68.15 | 66.27 | 66.72 | 70.10 | 59.47 | 59.47 | 59.47 | 59.47 |
| **Fusion** | (Hessel and Lee, 2019) | 72.52 | 70.82 | 71.34 | 73.82 | 63.03 | 63.03 | 63.03 | 63.03 |
| | TPC-GCN | **74.65** | **75.33** | **74.88** | **75.72** | **67.00** | **66.97** | **66.95** | **66.97** |

Table 2: Performance(%) comparison of the intra-topic experiments.

| Method | | Weibo Dataset | | | | Reddit Dataset | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | Avg. P | Avg. R | Avg. F1 | Acc. | Avg. P | Avg. R | Avg. F1 | Acc. |
| **Content-based** | TextCNN | 71.55 | 72.63 | 69.63 | 69.76 | 54.20 | 54.18 | 54.12 | 54.18 |
| | BiLSTM-Att | 67.09 | 68.09 | 67.10 | 68.00 | 60.96 | 59.76 | 58.63 | 59.76 |
| | BiGRU-Att | 68.04 | 67.08 | 67.35 | 70.18 | 58.49 | 58.17 | 57.76 | 58.17 |
| | BERT | 68.77 | 68.16 | 68.42 | 72.22 | 60.41 | 59.96 | 59.53 | 59.96 |
| **Structure-based** | SFC | 63.06 | 63.69 | 63.04 | 64.03 | 58.87 | 58.86 | 58.86 | 58.86 |
| **Fusion** | (Hessel and Lee, 2019) | 69.25 | 67.15 | 67.63 | 70.84 | 60.77 | 60.76 | 60.74 | 60.76 |
| | TPC-GCN | 73.84 | 72.00 | 71.53 | 72.11 | 63.39 | 63.24 | 63.14 | 63.24 |
| | DTPC-GCN | **75.57** | **75.31** | **75.27** | **75.35** | **68.76** | **67.63** | **67.14** | **67.63** |

Table 3: Performance(%) comparison of the inter-topic experiments.

tify the controversy. 3) The fusion models outperform or are comparable to the other baselines, which proves that information fusion of content and structure is necessary to improve the performance.

In the inter-topic experiments, we can see that 1) DTPC-GCN outperforms all baselines by 6.4% of F1 score at least, which validates that DTPC-GCN can detect controversy on unseen or dissimilar topics. 2) DTPC-GCN outperforms TPC-GCN by 3.74% on Weibo and 4.00% on Reddit. This indicates that feature disentanglement and dynamic fusion can significantly improve the performance of inter-topic controversy detection.

### 4.5 Ablation Study

To answer **EQ2** and part of **EQ3**, we also evaluate several internal models, i.e., the simplified variations of (D)TPC-GCN by removing some components or masking some representations. By the ablation study, we aim to investigate the impact of content and structural information in TPC-GCN and topic-related and topic-unrelated information in DTPC-GCN.

**Ablation Study of TPC-GCN**

We delete certain type of nodes (and the edges connect to them) to investigate their overall impact and mask the content by randomizing the initial representation to investigate the impact of content. Specifically, we investigate on the following simplified models of TPC-GCN:

**PC-GCN / TP-GCN**: discard the topic / comment nodes.

**(RT)PC-GCN / T(RP)C-GCN / TP(RC)-GCN**: randomly initialize the representation of topic / post / comment nodes.

From Table 4, we have the following observations: 1) TPC-GCN outperforms all simplified models, indicating that the necessity of structure and content from all types of nodes. 2) The models deleting comment information, i.e., TP-GCN and TP(RC)-GCN, experience a dramatic drop of performance, which shows the most importance of the comment information. 3) The impact of structural information vary between topics and comments. For comments, the structure can individually work (TP(RC)-GCN $>$ TP-GCN), while for topics, the structure have to collaborate with the content ((RT)PC-GCN $<$ PC-GCN on Weibo dataset).

**Ablation Study of DTPC-GCN**

We focus on the roles of the $U$ (topic-unrelated) branch and $R$ (topic-related) branch:

**U branch only**: Only $U$ branch is trained to capture topic-unrelated features.

**R branch only**: Only $R$ branch is trained to capture topic-related features.

Table 5 shows that both of the two branches can identify controversial posts well, but their performances are worse than the fusion model. Specifically, the $U$ branch performs slightly better than $R$, indicating the topic-unrelated features are more suitable for inter-topic detection. We infer that the two branches can learn good but different representation under the guide of the auxiliary task.

| Method | Weibo Dataset | | | | Reddit Dataset | | | |
|---|---|---|---|---|---|---|---|---|
| | Avg. P | Avg. R | Avg. F1 | Acc. | Avg. P | Avg. R | Avg. F1 | Acc. |
| TPC-GCN | **74.65** | **75.33** | **74.88** | **75.72** | **67.00** | **66.97** | **66.95** | **66.97** |
| PC-GCN | 73.49 | 74.16 | 73.72 | 74.59 | 66.48 | 65.60 | 65.14 | 65.60 |
| TP-GCN | 58.72 | 59.16 | 58.20 | 58.68 | 52.97 | 52.83 | 52.28 | 52.83 |
| (RT)PC-GCN | 71.78 | 71.07 | 71.35 | 73.14 | 65.86 | 65.80 | 65.77 | 65.80 |
| T(RP)C-GCN | 72.30 | 72.65 | 72.45 | 73.55 | 65.25 | 64.73 | 64.43 | 64.73 |
| TP(RC)-GCN | 59.66 | 59.80 | 59.71 | 61.36 | 62.98 | 62.80 | 62.67 | 62.80 |

Table 4: Ablation study of TPC-GCN in the intra-topic experiments (%).

| Method | Weibo Dataset | | | | Reddit Dataset | | | |
|---|---|---|---|---|---|---|---|---|
| | Avg. P | Avg. R | Avg. F1 | Acc. | Avg. P | Avg. R | Avg. F1 | Acc. |
| DTPC-GCN | **75.57** | **75.31** | **75.27** | **75.35** | **68.76** | **67.63** | **67.14** | **67.63** |
| U branch only | 74.06 | 74.06 | 74.05 | 74.05 | 63.95 | 63.94 | 63.94 | 63.94 |
| R branch only | 74.16 | 73.33 | 73.15 | 73.41 | 63.41 | 63.15 | 62.97 | 63.15 |

Table 5: Ablation study of DTPC-GCN in the inter-topic experiments (%).

### 4.6 Case Study

We conduct a case study to further answer **EQ3** from the perspective of data. We compare the attention weight of the $U$ and $R$ branch in DTPC-GCN and exhibit some examples where the final decisions lean on one of the two branches.

Figure 3 shows two examples in the testing set of Weibo dataset. The DTPC-GCN rely more on the topic-unrelated features from Branch $U$ when classifying Post 1 ($0.874 > 0.126$), while more on the topic-related features from Branch $R$ when classifying Post 2 ($0.217 < 0.783$). The topic of Post 1, *Cancel the Driving License*, is weakly relevant to topics in training set, and the comments mostly use topic-unspecific words such as simple *support* and *good proposal*. Thus, the topic-unrelated features are more beneficial for judging. In contrast, Post 2 discusses the death penalty for women and children traffickers, relevant to one of the topics in the training set, *Improve Sentencing Standards for Sexually Assault on Children*. Further, both of the two topics are full of comments on *death penalty*. Exploiting more of the topic-related features is reasonable for the final decision.

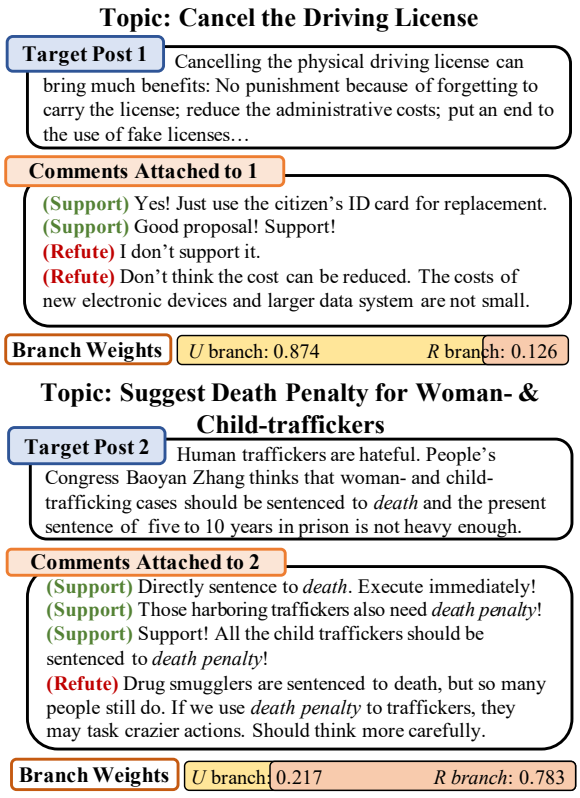


Figure 3: Examples of controversial posts that rely more on one of the two branches. The attention weights of the two posts are on the horizontal bars (left: Branch $U$, right: Branch $R$). Post 1 rely more on $U$ ($0.874 > 0.126$) while Post 2 more on $R$ ($0.217 < 0.783$).

## 5 Conclusion

In this paper, we propose a novel method TPC-GCN to integrate the information from the graph structure and content of topics, posts, and comments for post-level controversy detection on social media. Unlike the existing works, we exploit the information from related posts in the same topic and the reply structure for more effective detection. To improve the performance of our model for inter-topic detection, we propose an extension of TPC-GCN named DTPC-GCN, to disentangle the topic-related and topic-unrelated features and then dynamically fuse them. Extensive experiments conducted on two datasets demonstrate that our proposed models outperform the compared methods and prove that our models can integrate both semantic and structural information with significant genaralizablity.

## References


Aseel Addawood, Rezvaneh Rezapour, Omid Abdar, and Jana Diesner. 2017. Telling apart tweets associated with controversial versus non-controversial topics. In *Proceedings of the Second Workshop on NLP and Computational Social Science*, pages 32–41, Vancouver, Canada. Association for Computational Linguistics.

Rawia Awadallah, Maya Ramanath, and Gerhard Weikum. 2012. Harmony and dissonance: organizing the people's voices on political controversies. In *Proceedings of the fifth ACM International Conference on Web Search and Data Mining*, pages 523–532. ACM.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the third International Conference on Learning Representations*.

Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, pages 103–111. Association for Computational Linguistics.

Yoonjung Choi, Yuchul Jung, and Sung-Hyon Myaeng. 2010. Identifying controversial issues and their subtopics in news articles. In *Pacific-Asia Workshop on Intelligence and Security Informatics*, pages 140–153. Springer.

Mauro Coletto, Kiran Garimella, Aristides Gionis, and Claudio Lucchese. 2017. A motif-based approach for identifying controversy. In *Eleventh International AAAI Conference on Web and Social Media*. AAAI.

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.

Shiri Dori-Hacohen and James Allan. 2013. Detecting controversy on the web. In *Proceedings of the 22nd ACM Intedlrnational Conference on Information and Knowledge Management*, pages 1845–1848. ACM.

Shiri Dori-Hacohen and James Allan. 2015. Automated controversy detection on the web. In *European Conference on Information Retrieval*, pages 423–434. Springer.

Kiran Garimella, Gianmarco De Francisci Morales, Aristides Gionis, and Michael Mathioudakis. 2017. Reducing controversy by connecting opposing views. In *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, pages 81–90. ACM.

Kiran Garimella, Gianmarco De Francisci Morales, Aristides Gionis, and Michael Mathioudakis. 2018. Quantifying controversy on social media. *ACM Transactions on Social Computing*, 1(1):3.

Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural networks*, 18(5-6):602–610.

Jack Hessel and Lillian Lee. 2019. Something's brewing! early prediction of controversy-causing posts from discussion features. 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 1648–1659, Minneapolis, Minnesota. Association for Computational Linguistics.

Myungha Jang, John Foley, Shiri Dori-Hacohen, and James Allan. 2016. Probabilistic approaches to controversy detection. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management*, pages 2069–2072. ACM.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *Proceedings of the fifth International Conference on Learning Representations*.

Aniket Kittur, Bongwon Suh, Bryan A Pendleton, and Ed H Chi. 2007. He says, she says: conflict and coordination in wikipedia. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 453–462. ACM.

Chang Li and Dan Goldwasser. 2019. Encoding social information with graph convolutional networks forPolitical perspective detection in news media. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2594–2604. Association for Computational Linguistics.

Jasper Linmans, Bob van de Velde, and Evangelos Kanoulas. 2018. Improved and robust controversy detection in general web pages using semantic approaches under large scale conditions. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pages 1647–1650. ACM.

Diego Marcheggiani, Joost Bastings, and Ivan Titov. 2018. Exploiting semantics in neural machine translation with graph convolutional networks. 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 486–492. Association for Computational Linguistics.

Azadeh Nematzadeh, Grace Bang, Xiaomo Liu, and Zhiqiang Ma. 2019. Empirical study on detecting controversy in social media. *arXiv preprint arXiv:1909.01093*.

Ana-Maria Popescu and Marco Pennacchiotti. 2010. Detecting controversial events from twitter. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, pages 1873–1876. ACM.

Hoda Sepehri Rad and Denilson Barbosa. 2012. Identifying controversial articles in wikipedia: A comparative study. In *Proceedings of the eighth Annual International Symposium on Wikis and Open Collaboration*, page 7. ACM.

Nils Rethmeier, Marc Hübner, and Leonhard Hennig. 2018. Learning comment controversy prediction in web discussions using incidentally supervised multi-task CNNs. In *Proceedings of the 9th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*, pages 316–321. Association for Computational Linguistics.

Mikalai Tsytsarau, Themis Palpanas, and Kerstin Denecke. 2010. Scalable discovery of contradictions on the web. In *Proceedings of the 19th International Conference on World Wide Web*, pages 1195–1196. ACM.

Ba-Quy Vuong, Ee-Peng Lim, Aixin Sun, Minh-Tam Le, Hady Wirawan Lauw, and Kuiyu Chang. 2008. On ranking controversies in wikipedia: models and evaluation. In *Proceedings of the first International Conference on Web Search and Data Mining*, pages 171–182. ACM.

Lu Wang and Claire Cardie. 2014. A piece of my mind: A sentiment analysis approach for online dispute detection. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 693–699. Association for Computational Linguistics.

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. ACM.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Graph convolutional networks for text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7370–7377. AAAI.

Taha Yasseri, Robert Sumi, András Rung, András Kornai, and János Kertész. 2012. Dynamics of conflicts in wikipedia. *PloS one*, 7(6):1–12.

Rex Ying, Ruining He, Kaifeng Chen, Pong Eksombatchai, William L Hamilton, and Jure Leskovec. 2018. Graph convolutional neural networks for web-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 974–983. ACM.