

Recurrent Graph Neural Networks for Rumor Detection in Online Forums

Di Huang*
dh_599@usc.edu
University of Southern California
Los Angeles, California, USA

Jacob Bartel
bartel@google.com
Google
Mountain View, California, USA

John Palowitch
palowitch@google.com
Google Research
San Francisco, California, USA

KDD MIS2 '21

211019 Chia-Chun Ho

Outline

Introduction

Related Work

Preliminaries

Method

Experiments

Discussion

Comments

Introduction

Fake news detection

- Pew research reported that [social media outpaced print news as a news source](#) in the US in 2018.
- A 2016 Pew study found that seven out of ten [Reddit users use the platform to get their news](#).
- Following these findings, the seminal work showed that [user interaction signals](#) - differential patterns of liking, re-sharing, and commenting - distinguish between posts that link to certain categories of online content, in particular content later [identified as "rumors" versus other content](#).

Introduction

Rumor detection

- Most rumor detection models recently introduced have been tuned for and evaluated on data from [online social networks like Twitter or Facebook](#).
- These platforms have a [natural social graph](#) created by users, which provides an [inherent graph](#) on which a GNN can propagate rumor information.
- However, relatively less attention in rumor detection research has been given to [online forums like Reddit](#).

Introduction

Work of this paper

- Address **two nuances** specific to forums:
- Most forums don't have a **natural who-follows-who social graph**.
- Most forums don't **feature a "repost" option on their platform**, preventing usual inter-user cascades seen in social networks.
- Instead, each article in a forum is posted a limited number of times, each time independently by user across the platform.
- This means that each forum post (unlike social network posts) is the **start of its own discussion cascade consisting of a long comment-tree graph**, as opposed to repost/share behavior found on social networks.

Introduction

Contribution

- Illustrate the construction of emergent social networks **from online forum data**, which use both for feature learning and the downstream neural network computational graph via a GNN.
- Introduce a **Recurrent-GNN model** which can well-handle the **independent, sequential nature of article posting on forums**.
- Approach combines an RNN, to capture time-order of posts, with a GNN to capture the post comment relations of users.

Related Work

of fake news detection

- Some work ('13) uses [sentiment analysis](#) to distinguish [spam](#) comments and [trustworthy](#) comment.
- Some work (KDD'17) detect fake news via assessment on [credibility of article headlines](#).
- Other [content-based](#) approaches like [Traceminer](#)(WSDM'18) and [CSI](#)(CIKM'17) have been proposed which learn from [user interaction signals and information propagation paths](#).
- [Graph embedding](#) methods such as [Node2Vec](#), [SDNE](#) and [GCNs](#), have been widely user for network analysis and graph feature extraction in social network studies.

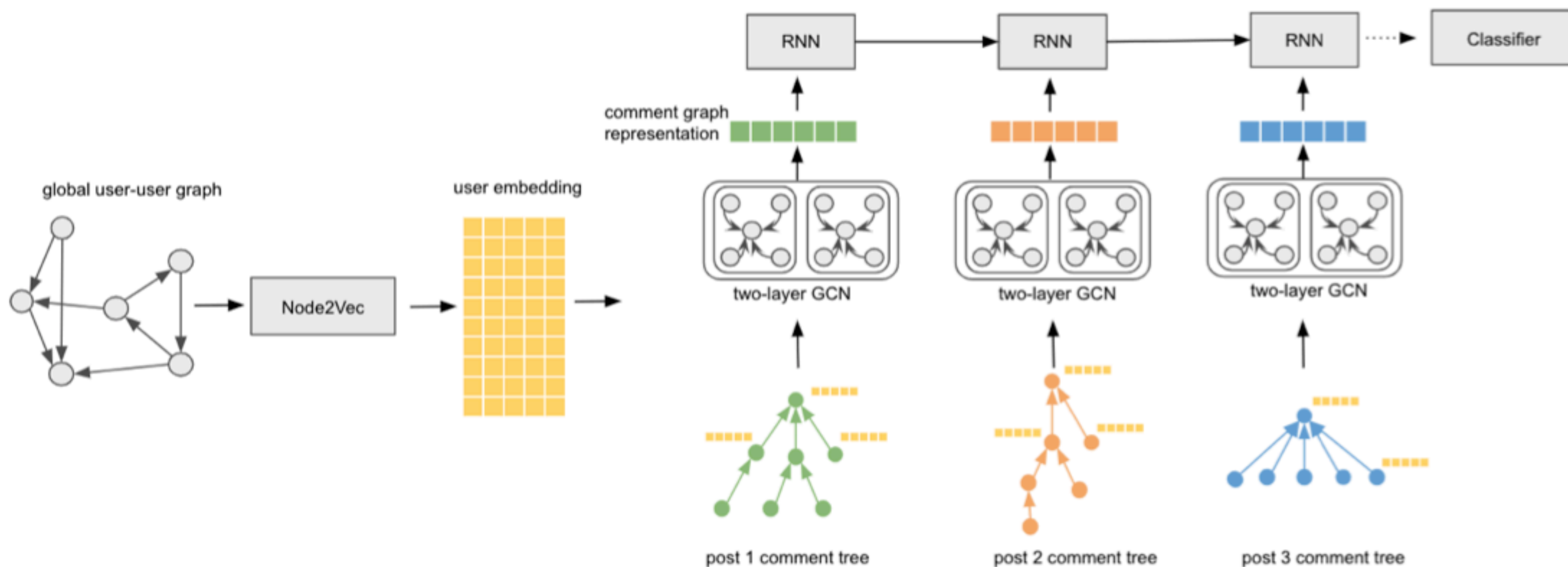
Preliminaries

Notations

- News and opinion pieces are being produced with record-breaking volume, and links to these pieces propagate swiftly on social sites like Twitter, Facebook, and Reddit.
- A **link** m is propagated by a sequence of **posts** $p = \{p_1, p_2, \dots\}$ having corresponding **authors** $a = \{a_1, a_2, \dots\}$.
- On forum sites like **Reddit**, for which our **commenter** $c = \{c_1, c_2, \dots\}$.
- $a, c \in \mathbb{U}$, total user set of the forum.
- For each task, pair each **message** m with a **categorical label** y , and train various models to predict y given feature data associated with any p, a , and c .

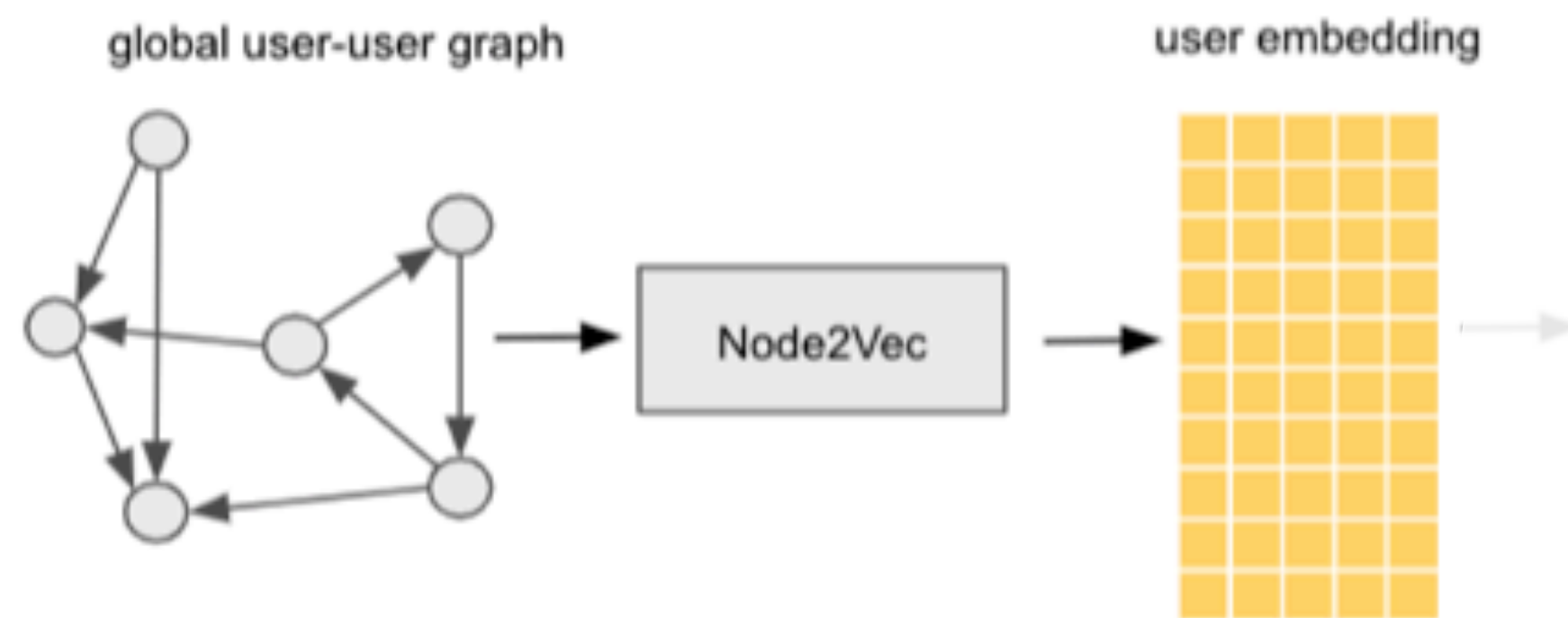
Method

Framework Overview



Method

Global User-User Interaction Graph



- Unlike Facebook, Twitter, forums like Reddit commonly **don't have a natural user-friendship graph**.

- $\mathbb{G} = (\mathbb{U}, \mathbb{E})$

- \mathbb{G} : graph from user interactions.

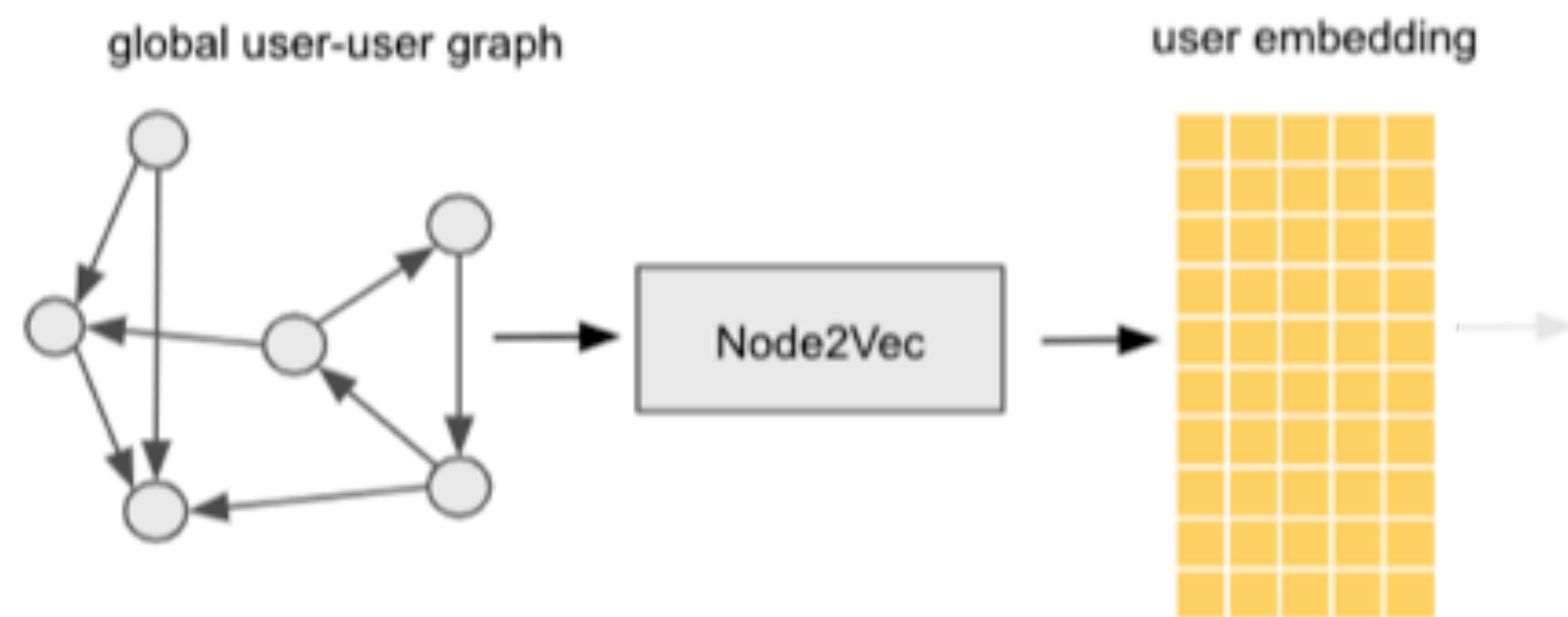
- \mathbb{U} : user set

- \mathbb{E} : edge set ($u_i \overset{w_{ij}}{\longleftrightarrow} u_j$ undirected, weighted)

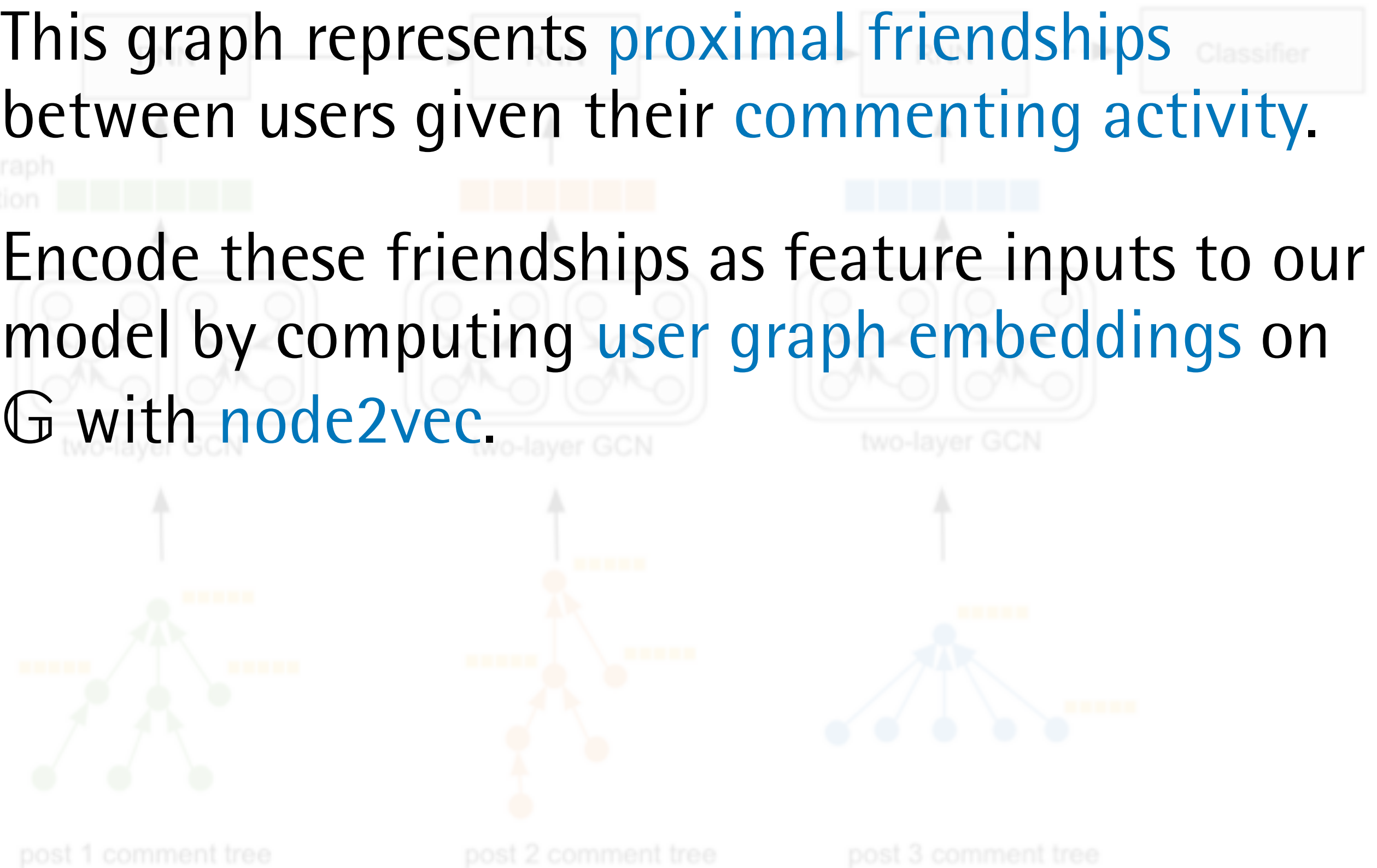
- w_{ij} : **count of comment-replies or post-replies** between u_i and u_j , on any post from Reddit.

Method

Global User-User Interaction Graph



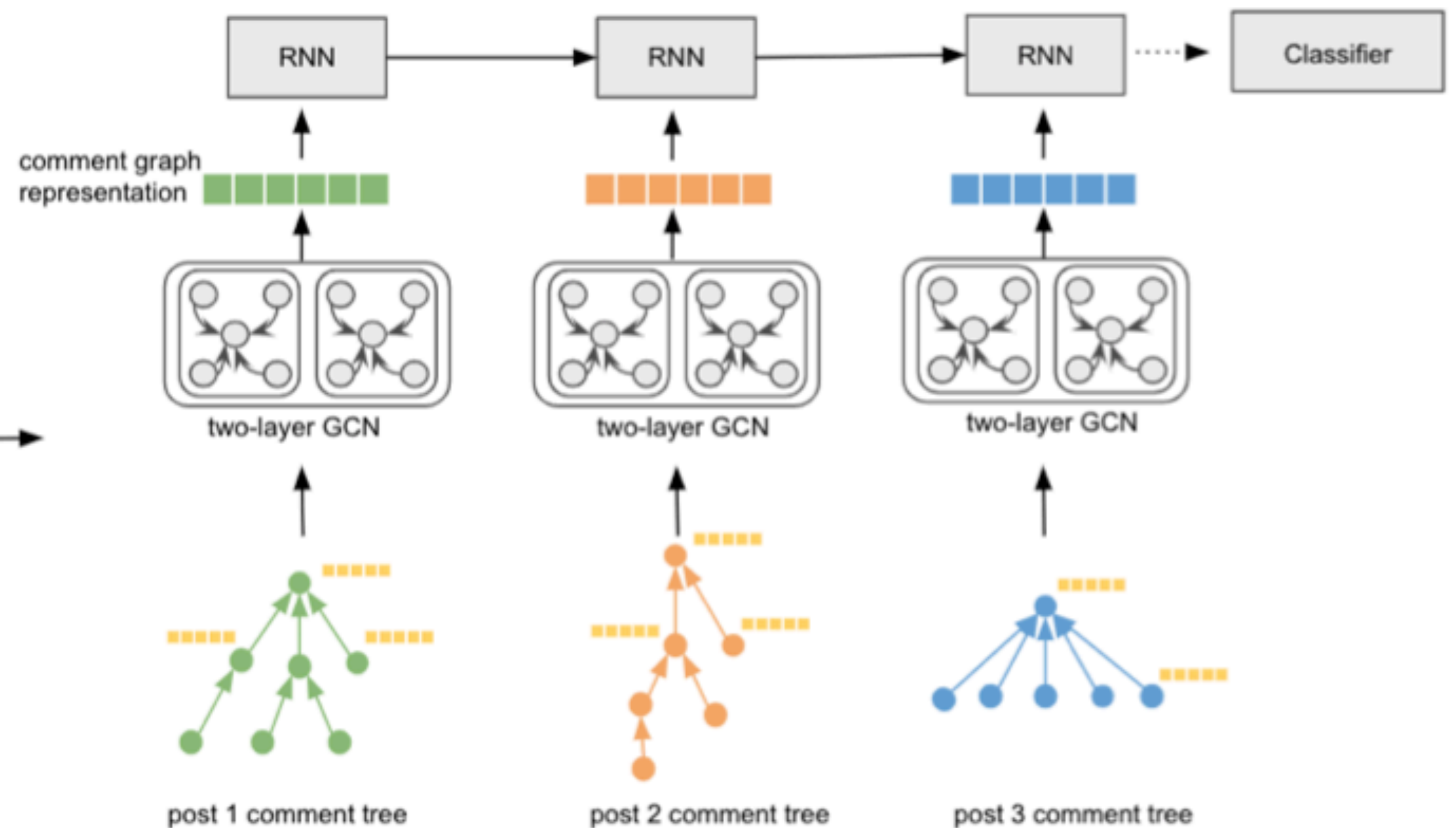
- This graph represents **proximal friendships** between users given their **commenting activity**.
- Encode these friendships as feature inputs to our model by computing **user graph embeddings** on \mathbb{G} with **node2vec**.



Method

GCN Post Encoding

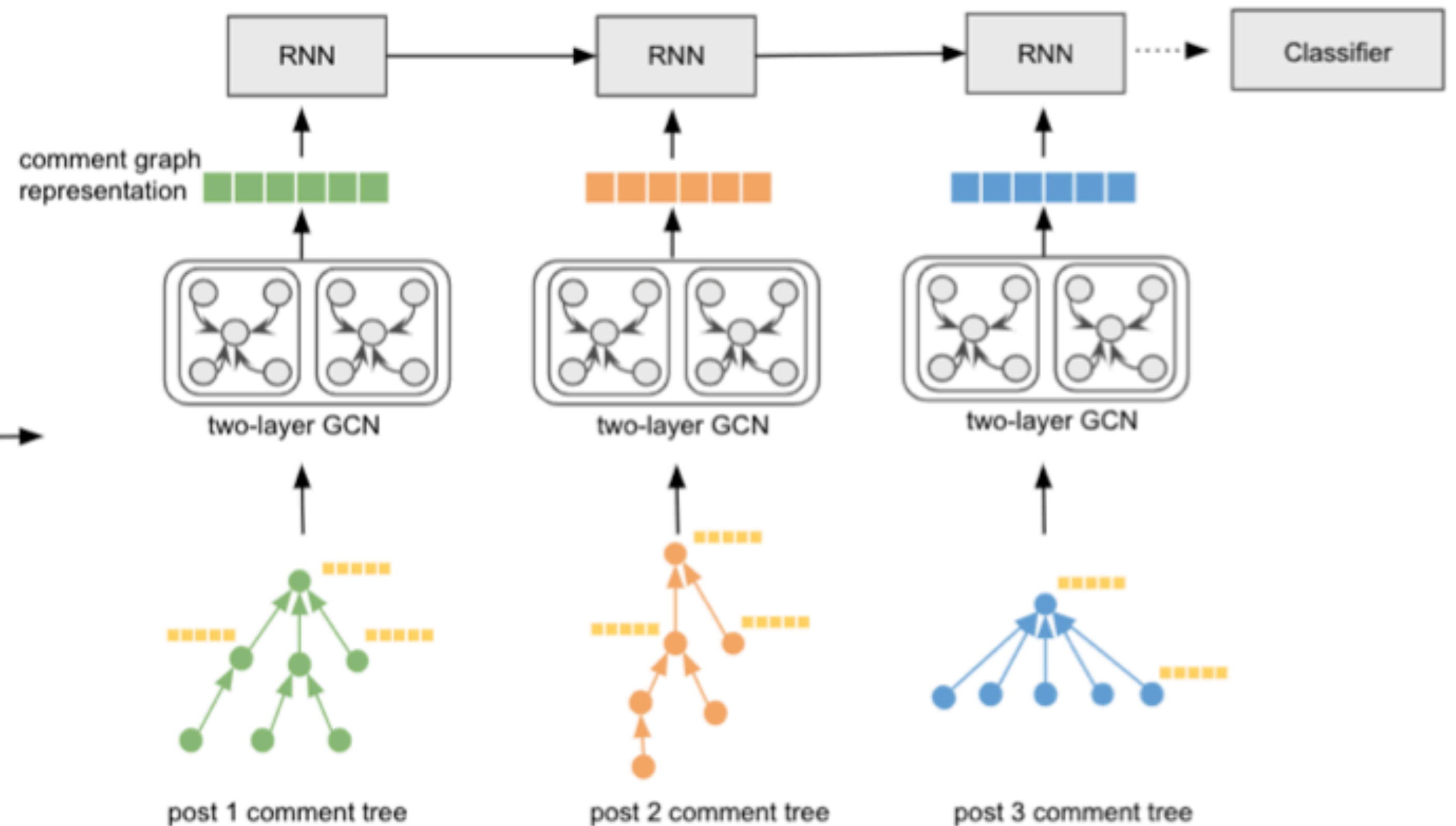
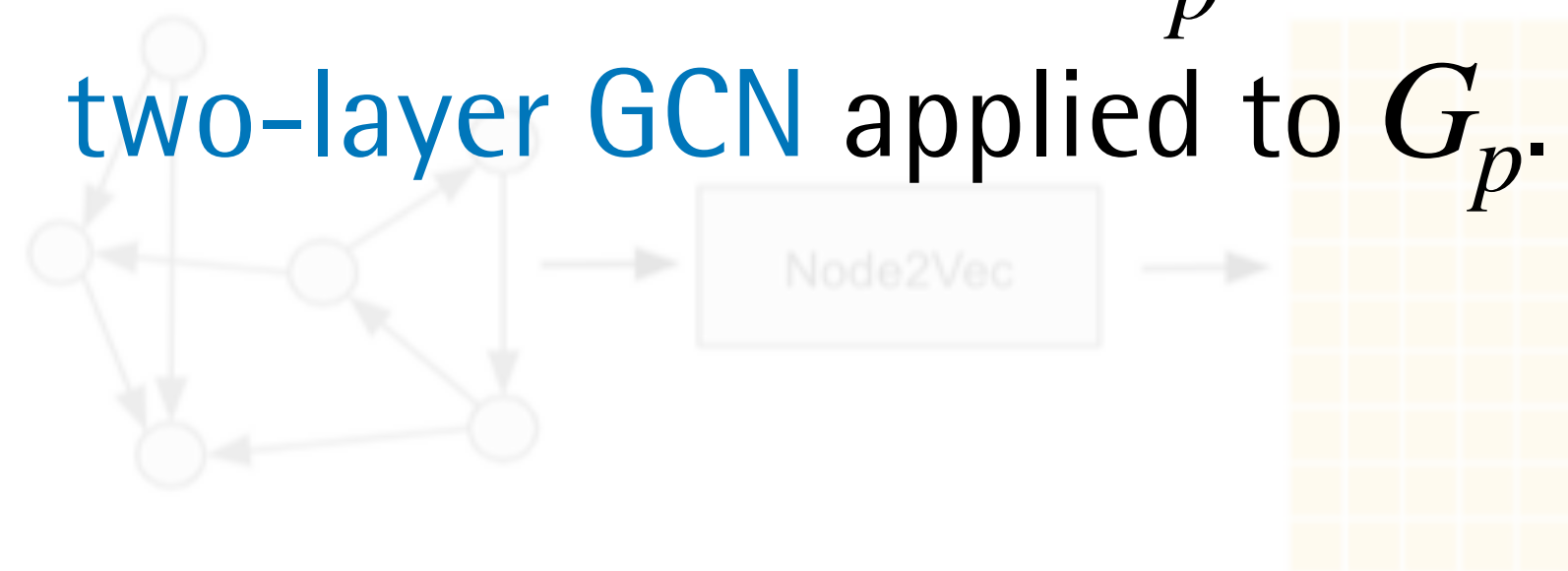
- Construct a **local reply-graph** G_p for each post p .
- $G_p = (U_p, E_p)$ consists of
 - U_p : users who commented on p (including author)
 - E_p : weighted edges represents the **number of times each pair pf commenters replied to each other**.



Method

GCN Post Encoding

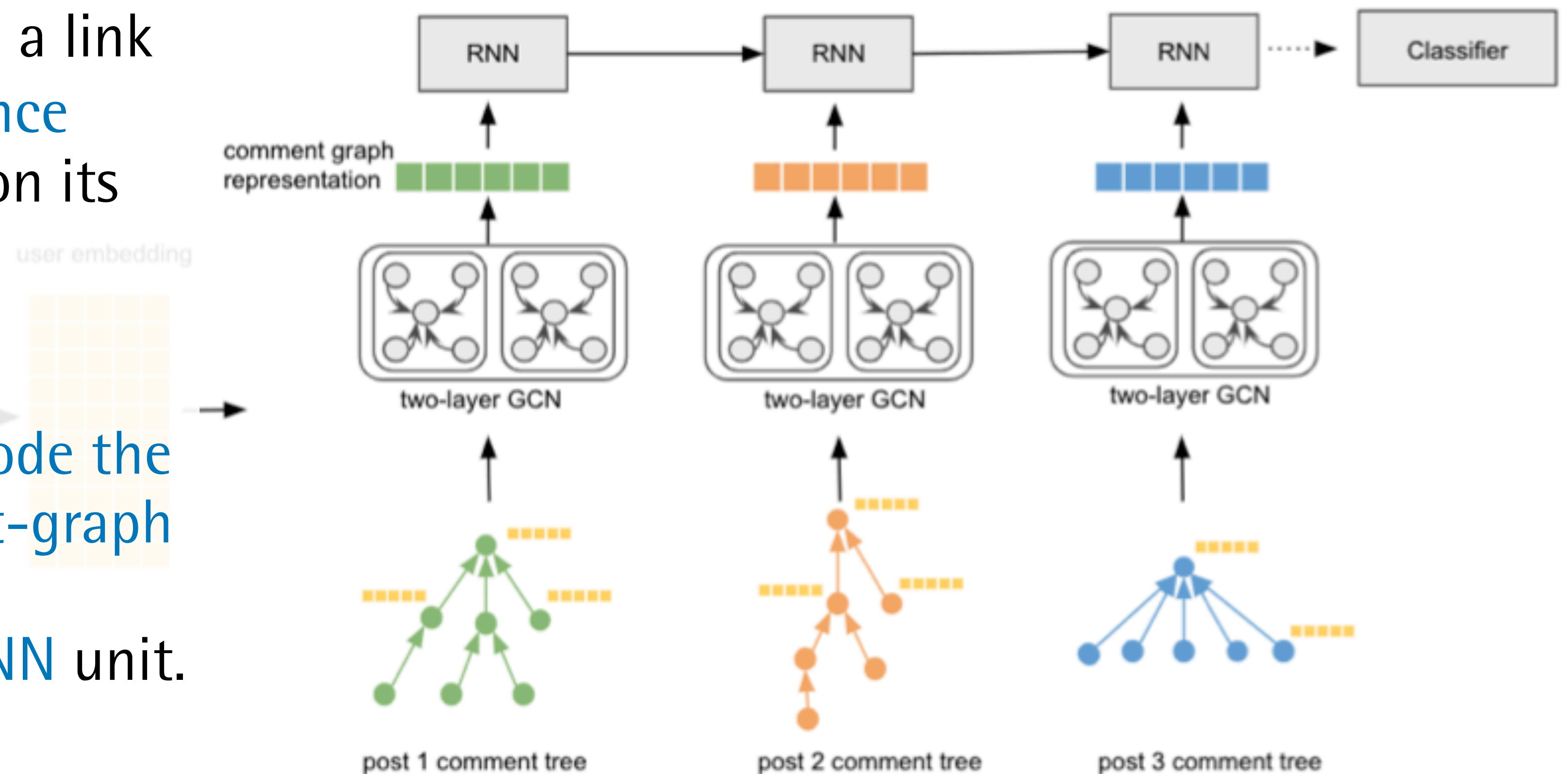
- With the aforementioned graph embeddings as user features, encode each post p into a hidden vector v_p with a **two-layer GCN** applied to G_p .



Method

RNN+GCN Post-Sequence Encoding

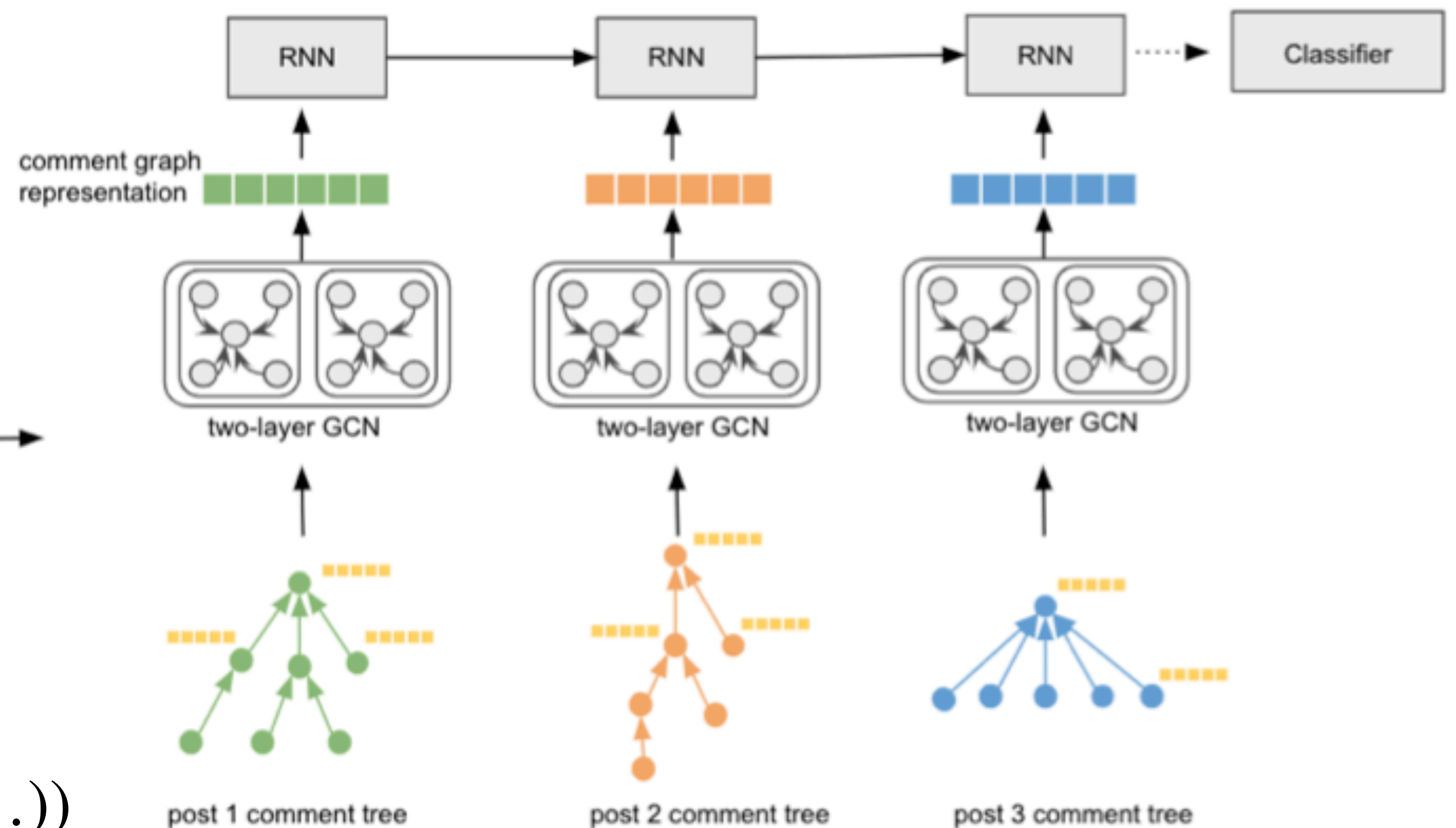
- Formulate inference on a link m as a **temporal sequence classification problem** on its time-ordered posts $\{p_{m1}, p_{m2}, \dots\}$.
- At each time-step, **encode the post with the comment-graph GCN**, and pass that representation to an **RNN** unit.



Method

RNN+GCN Post-Sequence Encoding

- Finally, predict y with a **multi-layer perceptron (MLP)** applied to the RNN encoding.
- Formally, given a link m and its corresponding post sequence p_1, p_2, \dots , apply the GCN to obtain post encodings v_1, v_2, \dots , and **infer a predicted \hat{y}** as
$$\hat{y} = \arg \max MLP(RNN(v_1, v_2, \dots))$$



Experiments

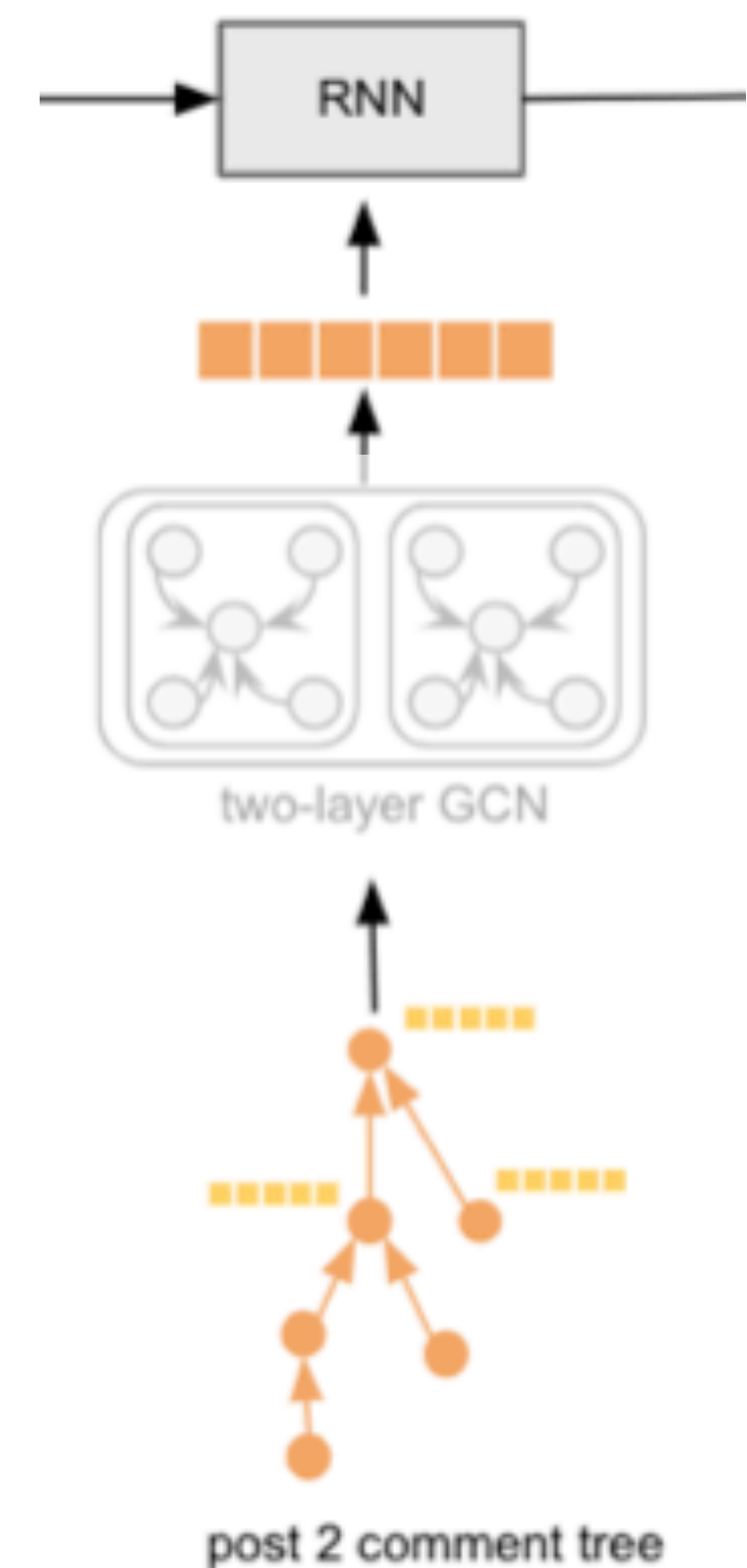
Settings

- Evaluation of our R-GNN model against 5 baselines on 2 tasks:
 - Article categorization
 - Rumor detection
- 4 baselines are standard (non-neural) machine learning methods applied to simplified features.
- 5th baseline is an established RNN-based method call TraceMiner, which has been previously evaluated on similar tasks using Twitter data.

Experiments

Variants of R-GNN

- R-GNN (-replygraph)
 - Remove the **GCN component** from proposed model, concatenating all authors a and commenter sequences $\{c_1, c_2, \dots\}$ associated with a post into a **single sequence**, which feed to an RNN.
 - Removes the influence of the comment graph signal from the learning process.



Experiments

Research Questions

- All of experiments implicitly test hypothesis that a “proximal” friendship graph can be derived from user interactions as a useful signal in these tasks.
- RQ1: Can signals derived purely from user interactions (absent a natural social graph) be successful in classifying links that shared in online forums?
- RQ2: Can diffusion process modeling with deep neural networks outperform standard ML models, applied to online forums?
- RQ3: Can our RNN+GCN hybrid model outperform simpler RNN only baselines, especially for rumor detection?

Experiments

Baselines

- SVM/XGBoost
 - Apply SVM and XGBoost directly to the **average embedding vectors of all users that authored or commented on any post** with a given link m .
- Traceminer (WSDM'18)
 - RNN-based diffusion model. It directly uses the **post-author graph embedding as the post representation** for RNN input.
 - Importantly, Traceminer **doesn't use any commenter** or comment-tree information.

Experiments

Rumor Detection

- Kaggle Dataset (Snopes, Politfact, and Emergent)
- Negative sampling same amount of news links from UCI dataset.
- 7,532 news links, with equal positive and negative examples.
- With 201.1k nodes and 413.0k edges.

Experiments

Rumor Detection

Model	Accuracy	F1
SVM(author)	0.6963	0.7025
XGBoost(author)	<u>0.6908</u>	0.6886
SVM(author+commenter)	0.6790	0.6447
XGoost(author+commenter)	0.6646	0.6594
Traceminer(author)	0.6401	<u>0.7536</u>
R-GNN(-replygraph)	0.7057	0.7485
R-GNN	0.6609	0.7731

- R-GNN model has the highest F1 score.
- R-GNN(-replygraph) achieves the highest accuracy.
- SVM(author) and Traceminer(author) ranks the second on accuracy and F1 score respectively.
- Overall, sequential modeling with deep learning achieved better performance than non-neural baselines on this task.

Experiments

Analysis

- The features for all models were **computed solely from a user-interaction based** derived proximal friendship.
- RQ1: Can signals derived purely from **user interactions (absent a natural social graph)** be successful in classifying links that shared in online forums?
 - As all models performed far better than random chance on each task, can answer RQ1 in the affirmative, that this **derived graph provides a useful signal** for link classification in online forums.

Experiments

Analysis

Model	Accuracy	F1
SVM(author)	0.6963	0.7025
XGBoost(author)	<u>0.6908</u>	0.6886
SVM(author+commenter)	0.6790	0.6447
XGoost(author+commenter)	0.6646	0.6594
Traceminer(author)	0.6401	<u>0.7536</u>
R-GNN(-replygraph)	0.7057	0.7485
R-GNN	0.6609	0.7731

- RQ2: Can diffusion process modeling with **deep neural networks outperform standard ML models**, applied to online forums?
- RQ3: Can our **RNN+GCN hybrid model outperform simpler RNN only baselines**, especially for rumor detection?
 - Full R-GNN model outperformed both neural and non-neural baselines, and R-GNN(-replygraph).
 - This suggests that the **diffusion/user-reply processes** which feed into the GCN are **more useful signals** for this task.
 - Thus in this case can answer RQ2 and RQ3 in the affirmative.

Discussion

- Introduced an approach for rumor detection and (more generally) link classification on forum websites, and evaluated this approach on [Reddit data](#).
- R-GNN is able to capture both the diffusion process of each link through post [comment-graphs via a GCN](#), and [simultaneously the sequential nature of link-posting on forums via an RNN](#).
- Proposed approach had superior performance to all baselines.
- To authors' knowledge, this is the 1st appearance of an [RGNN](#) in this space, and among the 1st demonstrations of deep learning on online user interactions [without a natural social graph](#).

Discussion

- In this short paper, have examined a narrow subset of potential tasks, model architectures, and available features in this space.
- For instance, to better understand the effect of **various interaction-based graph signals** – which let the model learn from diffusion processes on the directed graph of user actions.
- This approach have **disregarded the many content-based signals**, e.g. text or images, available for the tasks in this paper and others.
 - Have shed light on the capacity of state-of-the-art GNNs to model article sharing in forums with interaction-based features alone.

Comments of R-GNN

- Only create user-comment interaction to determine.
- The proposed method performance unsatisfactory.
- But weighted edge may can use in other social network graph-based framework.