Adversarial Active Learning based Heterogeneous Graph Neural Network for Fake News Detection

Yuxiang Ren*, Bo Wang†, Jiawei Zhang* and Yi Chang†
*IFM Lab, Department of Computer Science, Florida State University, FL, USA
†Artificial Intelligence, Jilin University, Jilin, China
Email: yuxiang@ifmlab.org, bowang19@mails.jlu.edu.cn, jiawei@ifmlab.org, yichang@jlu.edu.cn

***Abstract*—The explosive growth of fake news along with destructive effects on politics, economy, and public safety has increased the demand for fake news detection. Fake news on social media does not exist independently in the form of an article. Many other entities, such as news creators, news subjects, and so on, exist on social media and have relationships with news articles. Different entities and relationships can be modeled as a heterogeneous information network (HIN). In this paper, we attempt to solve the fake news detection problem with the support of a news-oriented HIN. We propose a novel fake news detection framework, namely Adversarial Active Learning-based Heterogeneous Graph Neural Network (AA-HGNN) which employs a novel hierarchical attention mechanism to perform node representation learning in the HIN. AA-HGNN utilizes an active learning framework to enhance learning performance, especially when facing the paucity of labeled data. An adversarial selector will be trained to query high-value candidates for the active learning framework. When the adversarial active learning is completed, AA-HGNN detects fake news by classifying news article nodes. Experiments with two real-world fake news datasets show that our model can outperform text-based models and other graph-based models when using less labeled data benefiting from the adversarial active learning. As a model with generalizability, AA-HGNN also has the ability to be widely used in other node classification-related applications on heterogeneous graphs.**

***Index Terms*—Heterogeneous Network, Graph Neural Network, Fake News Detection, Data Mining**

I. Introduction

With the widespread use of social networks, fake news has become a serious social problem that cannot be ignored. In politics, fake news biases people's judgments about major issues like Brexit [4] and the 2016 US presidential election [2]. A lot of fake news is spread on various social platforms during the 2016 US presidential election, e.g., on Facebook. 115 pro-Trump fake stories that were shared a total of 30 million times, and 41 pro-Clinton fake stories being shared a total of 7.6 million times are observed [2]. In the economic field, the extreme sensitivity of the capital market has caused it to suffer from fake news. For instance, \$130 billion is wiped out in stock value after a piece of fake news claimed that then-president Barack Obama was injured in an explosion [28]. In public safety affairs, people's responses to emergencies, from natural disasters to terrorist attacks, have been disrupted by the spread of false news online [23], [14], [42]. In view of this, the detection and mitigation of fake news is imperative.

However, detecting fake news on social media is particularly challenging. At first, fake news is written and published intentionally, so the content is carefully camouflaged. Fake content may account for only 1% of news articles, but it is sufficient for the purpose. This makes it difficult to detect fake news simply based on news articles. Secondly, fake news spreads much faster than real news. According to the research in [42], many more people retweeted falsehood than they did the truth on Twitter. Therefore, the detection of fake news has high requirements for timeliness. Once a large number of users have obtained false consultations, destructive effects have already been caused. What's more, it is expensive and time-consuming to check and label the credibility of news articles by experts manually. Fake news detection methods requiring a large number of labels are not practical in the real world.

On social media, focusing on news articles alone is not comprehensive, because news does not exist independently in the form of articles. In fact, there are many entities related to news articles, such as news creators, news subjects and so on. These different types of entities and their relationships provide a more comprehensive perspective on identifying news articles. A heterogeneous information network (HIN for short) [39], [36] can be utilized to represent these entities and relationships. An illustration of such a news oriented heterogenous information network (News-HIN) based on *PolitiFact*[1]

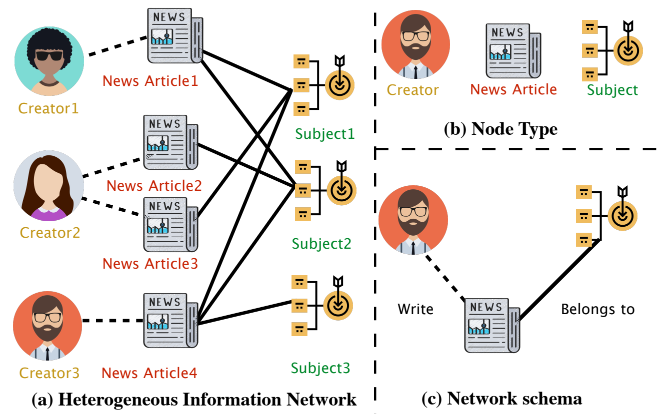


Fig. 1. An illustrative example of a heterogenous information network based on PoliticFact data (News-HIN). (a) A News-HIN consists three types of nodes and two types of links. (b) Three types of nodes (i.e., Creator, News article, Subject). (c) Network schema of News-HIN

[1] https://www.politifact.com/

data is presented in Figure 1. In addition to the information provided in the news article, we are able to collect profile information of news creators from social networks and other supplementary knowledge libraries. For the news subjects, the background and auxiliary knowledge can be collected to support the fake news detection. With the support of a News-HIN, fake news detection task can be formulated as the node classification problem. In this way, more sufficient information and knowledge can be used to check the credibility of news articles.

The main challenges of the fake news detection problem in a News-HIN lie in the following points:

- *Paucity of Training data*: Fake news appears and spreads very quickly. The real-time nature of news also makes outdated labels worthless. Therefore, fake news detection often faces the challenge of lacking valuable training data. This requires that models can effectively detect potential fake news with the support of a small amount of training data.
- *Heterogeneity*: Multiple types of heterogeneous information exist in a News-HIN, which can provide key signals for identifying fake news article nodes. At the same time, learning effective node representations in a News-HIN considering both structural and type information is non-trivial.
- *Generalizability*: In order to ensure the applicability of the proposed model to diverse and possibly changing News-HINs, we need to provide a general detection model that can handle News-HINs containing any types of nodes and different schemas.

To solve these challenges aforementioned, we propose a novel **A**dversarial **A**ctive Learning-based **H**eterogeneous **G**raph **N**eural **N**etwork (AA-HGNN) to detect fake news in the News-HIN. For the first challenge, the proposed framework is built on an active learning framework, where a classifier and a selector are included. By continuously querying high-value candidate nodes for classifier training and tuning, excellent performance can be achieved with a small amount of labeled data. For the second challenge, a heterogeneous graph neural network with a novel **H**ierarchical **G**raph **A**ttention (HGAT) mechanism is utilized in both the classifier and the selector. Based on the two-level attention mechanism (node-level & schema-level), HGAT can get the optimal combination of different types of neighbors in a hierarchical manner. The HGAT-based classifier is responsible for conducting classification on news article nodes. The HGAT-based selector is used to evaluate the predicted label from the classifier for high-value selection. The selected candidate nodes will become part of the training set via experts labeling. The classifier and the selector are trained based on adversarial learning: with the improvement of the predicted label quality by the classifier, the evaluation ability of the selector will be improved to continuously select better candidates. The overall architecture of proposed framework is shown in Figure 2. AA-HGNN has no limitation on the structures of News-HINs, thus it has good generalizability and can solve the third challenge well.

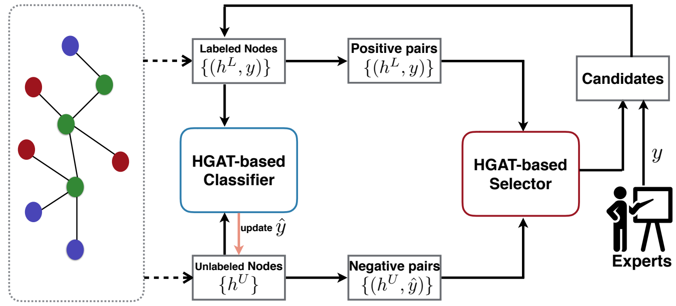


Fig. 2. Overall Framework.

We focus on applying AA-HGNN to fake news detection domain in this paper, but for more general problems of node classification on heterogeneous graphs, AA-HGNN is also applicable.

The contributions of our work are summarized as follows:

- We are the first to apply adversarial active learning to fake news detection, which can achieve excellent detection performance with much less training data. It is of great significance for fake news detection, because the urgent timeliness of fake news detection makes sufficient training data impossible.
- We propose a novel adversarial active learning-based framework AA-HGNN which can handle the heterogenity of News-HINs effectively through a two-level attention mechanism. AA-HGNN is applicable to HINs with different schemas.
- We conduct extensive experiments on two real-world datasets to demonstrate the effectiveness of AA-HGNN. The results show the superiority of AA-HGNN compared with the state-of-the-art models in detecting fake news, especially facing the paucity of training data.

II. Related Work

A. Fake News Detection

As an emerging topic, some research works in fake news detection have been proposed. Content-based fake news detection is based primarily on the deep mining of news content. [6], [35] extract the knowledge, a set of (Subject, Predicate, Object) triples [10], from the news content and assess the authenticity of news by comparing them with real knowledge. However, the timeliness and integrity of the knowledge map still limit the application of them [49]. Writing style is extracted and utilized to measure the credibility of news by some methods. [31] employs rhetorical structure theory to evaluate the authenticity in discourse level. [25], [26] capture the sentiment and readability of the news content to access the extent of falsehood. But these methods based on writing style can be hard to work in the face of carefully camouflaged fake news.

Some methods use not only the news content, but also other information related to the news. Guo et al. [13] utilize LSTM and a hierarchical attention mechanism to detect rumors, which makes use of social information through the proposed social feature. Shu et al. [7] study the explainable detection of fake news with the support of both news contents and user comments. Jin et al. evaluate news credibility within a graph optimization framework [18]. Methods based on matrix factorization [37], tensor factorization [15], and recurrent neural networks (RNNs) [32], [48] are proposed to work on the news-oriented networks.

In this paper, we model the news content and related entities as a News-HIN. Both structural information and node content of News-HIN are utilized by AA-HGNN to identify fake news.

B. Graph Neural Network

Graph Neural Networks (GNNs) learn nodes' new feature vectors through a recursive neighborhood aggregation scheme [12], [33], [46]. A propagation model incorporating gated recurrent units to propagate information across all nodes is proposed in [22]. Recently, there is a surge of generalizing convolutional operation on the graph-structured data. Joan Bruna et al. [5] extend convolution to general graphs by a novel Fourier transformation in graphs. Kipf et al. [20] propose Graph Convolutional Network (GCN). Hamilton et al. [16] introduce GraphSAGE which generates embeddings by aggregating features from a node's local neighborhood directly. Graph Attention Network (GAT) [41] first imports the attention mechanism into graphs, which is utilized to learn the importance of neighbors and aggregates the neighbors to learn the representation of nodes in the graph. However, the above graph neural networks are presented for the homogeneous graphs. Wang et al. [45] consider the attention mechanism in heterogeneous graph learning through the model HAN, where information from multiple meta-path defined connections can be learned effectively. However, meta-path as a handcrafted feature limits HAN. In addition, HAN only considers different types of connections between target nodes through meta-path but ignores the use of node contents carried by different types of nodes.

C. Adversarial and Active Learning

The principle of adversarial learning is invented in generative adversarial networks (GANs) by Goodfellow et al. [11]. Adversarial learning principle has achieved excellent performance in many different topics, such as text classification [21], information retrieval [43], and network embedding [17], [8]. Adversarial learning method on heterogeneous network embeddings [17] can be used to learn a more efficient representation of news nodes in News-HIN. However, in order to detect fake news, HeGAN [17] still requires a large number of labeled data to train a classifier. Active learning is an effective way to train a model with less labeled data, because not all training samples are equally important [1]. The number of labels needed to learn actively can be logarithmic in the usual sample complexity of passive learning [9] . Active learning also proves its value and robustness on different topics including recommendation systems [30], social network alignment [29], image classification [44] and graph matching [34].

In this paper, AA-HGNN combines adversarial learning and active learning. Selectors trained in an adversarial manner can continuously select high-value candidates for active learning. The high-value candidates further improve the performance of the classifier.

III. Concept and Problem Definition

A. Terminology Definition

In order to make it easier to understand related concepts, we will use the *PolitiFact* data as an example to illustrate here. The *PolitiFact* data contain News articles, Subjects and Creators, which can be modeled into a heterogeneous network as three types of nodes and construct different types of links based on the connections among them. We can define News Oriented Heterogeneous Information Networks (News-HIN) formally as follows:

DEFINITION *1:* (News Oriented Heterogeneous Information Networks (News-HIN)): The news oriented heterogeneous information network (News-HIN) can be defined as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where the node set $\mathcal{V} = \mathcal{C} \cup \mathcal{N} \cup \mathcal{S}$. $\mathcal{C}, \mathcal{N}$ and $\mathcal{S}$ represent Creators, News articles and Subjects respectively. We will define different types of nodes in detail later. The link set $\mathcal{E} = \mathcal{E}_{c,n} \cup \mathcal{E}_{n,s}$ involves the "Write" links between creators and news articles, and the "Belongs to" links between news articles and subjects.

News articles refer to the news content post on social media or public platforms. We can define news articles in a formal way as:

DEFINITION *2:* (News Articles): The News articles set can be represented as $\mathcal{N} = \{n_1, n_2, \cdots, n_m\}$. For each news article $n_i \in \mathcal{N}$, it contains its textual contents.

The credibility label of n_i takes value from the label set $\mathcal{Y} = \{Fake, Real\}$. In this paper, the original label set contains 6 different class labels (True, Mostly True, Half True, Mostly False, False, Pants on Fire). We group the labels Pants on Fire, False, Mostly False as fake news and group True, Mostly True, Half True as real news. Subjects denote the central ideas of news articles, which normally are the main objectives of writing news articles.

DEFINITION *3:* (Subjects): The set of subjects can be denoted as $\mathcal{S} = \{s_1, s_2, \cdots, s_k\}$. For each subject $s_i \in \mathcal{S}$, it contains the textual description.

Creators denote people who write news articles. We can also define this concept in a formal way.

DEFINITION *4:* (Creators): The set of creators can be represented as $\mathcal{C} = \{c_1, c_2, \cdots, c_n\}$. For each creator $c_i \in \mathcal{C}$, it contains the profile information.

In the PolitiFact dataset, the creators have the profile containing their titles, political party membership, and geographical residential locations. The profile information can be described by a sequence of words.

In order to better understand the News-HIN and utilize type information, it is necessary to define the schema-level description. The schema of News-HIN serves for learning the importance of nodes and links with different types.

DEFINITION *5:* (News-HIN Schema): Formally, the schema of the given News-HIN $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ can be represented as $S_{\mathcal{G}} = (\mathcal{V}_T, \mathcal{E}_T)$, where $\mathcal{V}_T$ and $\mathcal{E}_T$ denote the set of node types and link types in the network respectively. Here, $\mathcal{V}_T = \{\phi_n, \phi_c, \phi_s\}$ and $\mathcal{E}_T = \{Write, Belongs\ to\}$.

An illustration of News-HIN Schema based on the PolitiFact data is shown in Figure 1(c).

B. Problem Formulation

Given a News-HIN $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, the fake news detection problem aims at learning a classification function $f : \mathcal{N} \rightarrow \mathcal{Y}$ to classify news article nodes in the set $\mathcal{N}$ into the correct class with the credibility label in $\mathcal{Y}$. The news article nodes with labels can be grouped as a labeled set $\mathcal{L}$ and the rest news article nodes will be denoted as the unlabeled set $\mathcal{U} = \mathcal{N} \setminus \mathcal{L}$. Based on the active learning setting, we are also allowed to query for labels of news article nodes in $\mathcal{U}$ with a upper limit budget b. We also want to propose a mechanism to achieve an optimal query set $\mathcal{U}_q$ to improve the classification function $f : \mathcal{N} \rightarrow \mathcal{Y}$. To resolve the above fake news detection problem, we will introduce the proposed adversarial active learning based heterogeneous graph neural network AA-HGNN in Section IV.

IV. Proposed Method

In this section, we propose a novel **A**dversarial **A**ctive Learning based **H**eterogeneous **G**raph **N**eural **N**etwork (AA-HGNN) to detect fake news. As shown in Figure 2, AA-HGNN consists of two major components: (1) HGAT-based classifier, and (2) HGAT-based selector. We begin with the overview of the model, followed by detailed descriptions of the hierarchical graph attention neural network (HGAT). Then we illustrate the HGAT-based classifier and HGAT-based selector respectively. At last, we elaborate on the optimization of AA-HGNN.

A. Model Overview

The architecture of AA-HGNN is shown in Figure 2. The News-HIN $\mathcal{G}$ is the input of the HGAT-based classifier. h^L and h^U denote the initial feature of a labeled node and an unlabeled node respectively. The HGAT-based classifier is trained with both labeled and unlabeled data to predict labels $\{\hat{y}\}$ for unlabeled news article nodes. The HGAT-based selector evaluates the quality of predicted labels and selects high-value candidates from them based on a query strategy. We take the pairs of labeled nodes and their ground-truth labels $\{y\}$ as positive samples, and the pairs of unlabeled nodes and their predicted labels $\{\hat{y}\}$ are used as negative samples. A portion of positive and negative pairs are sampled to train the HGAT-based selector. After being trained, the selector outputs the confidence $\mathcal{P}$ of pairs in the test set. Based on the confidence, the proposed selection strategy selects a set of high-value unlabeled nodes as candidates with the size k. These candidates will be labeled by experts. In our experiments, these candidates will be moved to the training set before next round optimization. A query budget b is pre-specified for AA-HGNN. When the query budget b is exceeded, the adversarial active learning stops.

Since Hierarchical Graph Attention Neural Network (HGAT) is the basis of the classifier and the selector, which is the key to handling the heterogeneity, we will first introduce HGAT in detail in the next section.

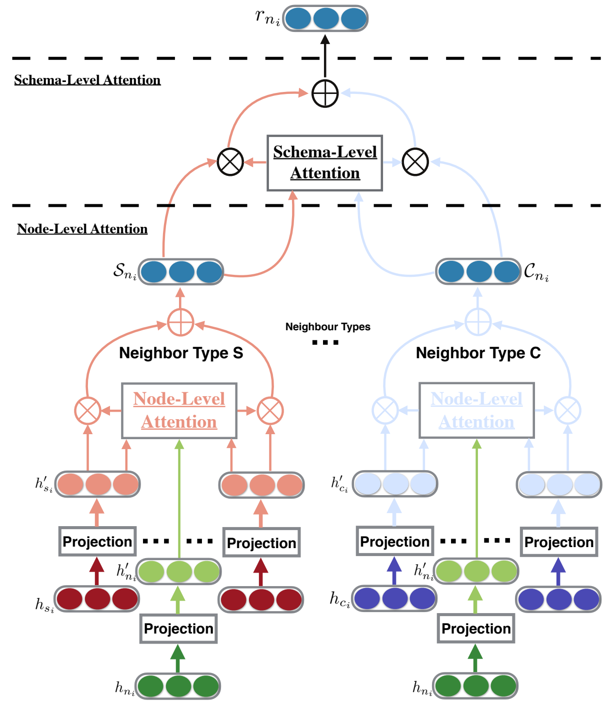


Fig. 3. Hierarchical Graph Attention Neural Network.

B. Hierarchical Graph Attention Neural Network (HGAT)

The novel HGAT employs a two-level attention mechanism including node-level attention and schema-level attention. The structure of HGAT is shown in Figure 3. Node-level attention is responsible for learning the weights of neighbors belong to the same type and aggregates them to get the type-specific neighbor representation. Schema-level attention enables HGAT to learn the information of node types and get the optimal weighted combination of the type-specific neighbor representations. Through the two-level attention mechanism, the representations of news article nodes contain both the structural and node content information.

1) Node-level attention: The node-level attention can learn the importance of neighbors belong to the same type respectively for each news article node $n_i \in \mathcal{N}$, and then aggregates the representation of same-type neighbors to form an integrated representation which we define as a schema node.

The inputs of the node-level attention layer are the node initial feature vectors $\{h\}$. Because multiple types of nodes exist in the News-HIN, the initial feature vectors belong to feature spaces with different dimensions. In order to enable the attention mechanism to output comparable and meaningful weights between different types of nodes, we first utilize a type-specific transformation matrix to project features with different dimensions into the same feature space. We take the news article node $n_i \in \mathcal{N}$ as an example. The transformation matrix for type ϕ_n is $\mathbf{M}^{\phi_n} \in \mathbb{R}^{F \times F^{\phi_n}}$, where F^{ϕ_n} is the dimension of the initial feature $h_{n_i} \in \mathbb{R}^{F^{\phi_n}}$ of the news article node n_i and F is the dimension of the feature space mapped to. The projection process can be shown as follows:

$$h'_{n_i} = \mathbf{M}^{\phi_n} \cdot h_{n_i} \tag{1}$$

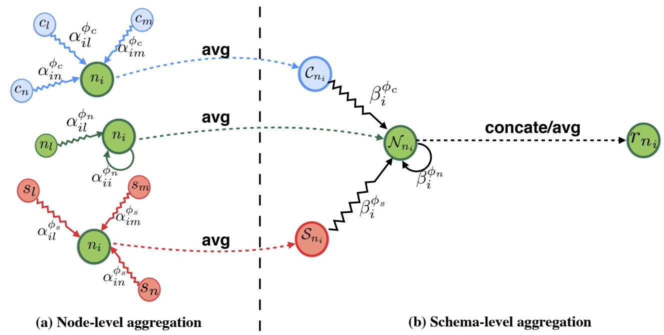


Fig. 4. Explanation of aggregating process in node-level and schema-level.

The h'_{n_i} is the projected feature of node n_i. The F is the same for all type-specific transformation matrices. Through the type-specific projection operation, the feature space of nodes with different types can be unified where the self-attention mechanism can work on to learn the weight among various kinds of nodes.

In the face of fake news detection, the target node is the news article node $n_i \in \mathcal{N}$. The neighbors of it belong to $\mathcal{N} \cup \mathcal{S} \cup \mathcal{C}$. It should be noted that we also regard the target node itself as a neighbor node to cooperate the self-attention mechanism. We let $T \in \{\mathcal{N}, \mathcal{S}, \mathcal{C}\}$ and nodes in T have the type ϕ_t. For n_i's neighbor nodes in T, the node-level attention can learn the importance $e_{ij}^{\phi_t}$ which means how important node $t_j \in T$ will be for n_i. The importance of the node pair (n_i, t_j) can be formulated as follows:

$$e_{ij}^{\phi_t} = att(h'_{n_i}, h'_{t_j}; \phi_t) \tag{2}$$

Here, the node-level attention att denotes the same deep neural network as [41]. att is shared for all neighbor nodes with the same type ϕ_t. The masked attention captures the network structure information where only node $t_j \in neighbor_{n_i}$ (being neighbors of node n_i) will be calculated and recorded as $e_{ij}^{\phi_t}$. Otherwise, the attention weight will be 0. We normalize them to get the weight coefficient $\alpha_{ij}^{\phi_t}$ via softmax function:

$$\alpha_{ij}^{\phi_t} = \text{softmax}_j(e_{ij}^{\phi_t}) = \frac{exp(e_{ij}^{\phi_t})}{\sum_{t_k \in neighbor_{n_i}} e_{ik}^{\phi_t}} \tag{3}$$

Then, the schema node T_{n_i} can be aggregated by the neighbor's projected features with the corresponding weights as follows:

$$T_{n_i} = \sigma(\sum_{t_j \in neighbor_{n_i}} \alpha_{ij}^{\phi_t} \cdot h'_{t_j}) \tag{4}$$

Similar to Graph Attention Network (GAT) [41], a multi-head attention mechanism can be used to stabilize the learning process of self-attention in node-level attention. In details, K independent node-level attentions execute the transformation of Equation (4), and then the features achieved by K heads will be concatenated, resulting in the output representation of the schema node:

$$T_{n_i} = \Big\Vert_{k=1}^{K} \sigma(\sum_{t_j \in neighbor_{n_i}} \alpha_{ij}^{\phi_t} \cdot h'_{t_j}) \tag{5}$$

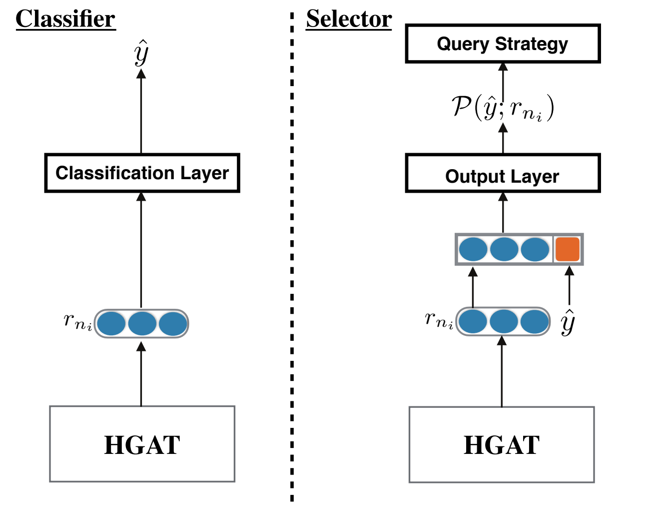


Fig. 5. HGAT-based Classifier and HGAT-based Selector.

where $\Vert$ represents concatenation. In the problem we face, every target node n_i has 3 schema nodes corresponding to 3 different types neighbors (include itself) based on the Definition 5. They can be denoted as $\mathcal{N}_{n_i}$, $\mathcal{C}_{n_i}$, $\mathcal{S}_{n_i}$.

2) Schema-level attention: Through the node-level attention, we fuse information from neighbor nodes with the same type into the representation of a schema node. Now, HGAT needs to learn the representation of news article nodes from all schema nodes. Different schema nodes contain type-specific information, which requires us to learn the importance of different node types. Here, the schema-level attention is proposed to learn the importance of different schema nodes, and finally use the learned coefficients for weighted combination.

In order to obtain sufficient expressive power to calculate the attention weights between schema nodes, one learnable linear transformation is applied to the schema nodes. The linear transformation is parametrized by a weight matrix $\mathbf{W} \in \mathbb{R}^{F' \times KF}$, where K is the number of heads in node-level attention. The schema-level attention $schema$ is a single-layer feedforward neural network applying the activating function Sigmoid with the dimension $2F'$. For the schema node T_{n_i}, the importance of it can be denoted as $w_i^{\phi_t}$:

$$w_i^{\phi_t} = schema(\mathbf{W}T_{n_i}, \mathbf{W}\mathcal{N}_{n_i}) \tag{6}$$

We normalize the imoportance of each schema nodes through a softmax function. Then coefficients of the final fusion can be denoted as $\beta_i^{\phi_t}$, which can be calculated as follows:

$$\beta_i^{\phi_t} = \text{softmax}_t(w_i^{\phi_t}) = \frac{\exp(w_i^{\phi_t})}{\sum_{\phi \in \mathcal{V}_T} \exp(w_i^{\phi})} \tag{7}$$

Based on the learned coefficients, we can fuse all schema nodes to get the final representation $r_{n_i} \in \mathbb{R}^{F'}$ of the target node n_i:

$$r_{n_i} = \sum_{\phi_t \in \mathcal{V}_T} \beta_i^{\phi_t} \cdot T_{n_i} \tag{8}$$

We also describe the two-level aggregating process in Figure 4 for reference.

C. HGAT-based Classifier

As shown in the left side of Figure 5, HGAT and a classification layer constitute a HGAT-based classifier. The input of HGAT-based classifier is the same as HGAT, which are the initial feature vectors of nodes. The classification layer can output the predicted labels $\{\hat{y}\}$ of unlabeled news article nodes. In our experiments, a logistic regression layer works as the classification layer.

For the fake news detection tasks, the optimization objective function of the HGAT-based classifier can leverage the cross-entropy loss minimization. The HGAT-based classifier can be optimized in an end-to-end manner by backpropagation. We define the set of labeled news article nodes as $\mathcal{N}_L$ and the set of unlabelled news article nodes as $\mathcal{N}_U$, then the cross-entropy loss we used can be written as:

$$Loss_{classifier} = -\sum_{n_i \in \mathcal{N}_L} (y_{n_i} \log(p_{n_i}) + (1-y_{n_i}) \log(1-p_{n_i})) \tag{9}$$

Here, y_{n_i} is a binary indicator (0 or 1) indicating if the binary class label is the correct classification for the news article node representation r_{n_i}. p_{n_i} is the predicted probability of labeled news article node n_i.

When the optimization is completed, the predicted probability of unlabeled news article nodes in $\mathcal{N}_U$ are rounded and cast into predicted labels $\{\hat{y}\}$. The predicted labels $\{\hat{y}\}$ will be evaluated by the HGAT-based selector which is described in the next section.

D. HGAT-based Selector

The structure of a HGAT-based selector is shown in the right side of Figure 5. The inputs of the layers of HGAT are the initial feature vectors $\{h\}$. Based on the learned representation r_{n_i}, we then concatenate r_{n_i} with the predicted label $\hat{y}$ (or the groud-truth label y of the labeled node). We denote this concatenated vector as $z_{n_i} \in \mathbb{R}^{(F'+1)}$:

$$z_{n_i} = [r_{n_i}, \hat{y}] \tag{10}$$

The purpose of the HGAT-based selector is to evaluate the probability that how likely the z_{n_i} is from the set of labeled news article nodes $\mathcal{N}_L$. A higher possibility represents that a news article node matches the predicted label better. At the same time, if a node does not match the predicted label, it is likely to indicate that the predicted label is wrong. The output layer is responsible for predicting the probability $\mathcal{P}(\hat{y}; r_{n_i})$. Here, we use a logistic regression layer as the output layer. We sample $z_{n_j}, n_j \in \mathcal{N}_L$ as the positive samples, and the same number of $z_{n_k}, n_k \in \mathcal{N}_U$ are sampled as the negative samples. These positive and negative samples constitute the training set for the HGAT-based selector. The loss function used by HGAT-based selector is a cross-entropy loss:

$$Loss_{selector} = -\sum (y \log(\mathcal{P}) + (1-y) \log(1-\mathcal{P})) \tag{11}$$

$y \in \{0,1\}$ denotes the negative-positive label of the concatenated vector in training set. $\mathcal{P}$ is the predicted probability of label being positive. This loss function can be optimized by backpropagation.

Algorithm 1: Adversarial Active optimization of AA-HGNN

Input: The News-HIN $\mathcal{G} = (\mathcal{V}, \mathcal{E})$; The set of labeled news article nodes $\mathcal{N}_L$; The set of unlabeled news article nodes $\mathcal{N}_U$; The query budget b; The query batch size k; Number of samples m;

$\mathcal{U}_q = \emptyset$;
while $|\mathcal{U}_q| < b$ **do**
  ▷ **Optimization for HGAT-based classifier**;
  begin
    Train the HGAT-based classifier on $\mathcal{N}_L$ via Eq.9;
    Predict the labels of nodes in $\mathcal{N}_U$;
    Update the set of predicted labels $\{\hat{y}\}$;
  ▷ **Optimization for HGAT-based selector**;
  begin
    Sample m nodes from $\mathcal{N}_L$ to construct positive samples via Eq.10, i.e., $z_{n_j}, n_j \in \mathcal{N}_L$;
    Sample m nodes from $\mathcal{N}_U$ to construct negative samples via Eq.10, i.e., $z_{n_k}, n_k \in \mathcal{N}_U$;
    Train the HGAT-based selector on positive and negative samples;
    Predict the probability $\mathcal{P}$ via Eq.11;
    Query k candidates based on Definition 6;
  $\mathcal{U}_q = \mathcal{U}_q \cup \{candidates\}$;
  Labeling k candidates by experts;
  $\mathcal{N}_L = \mathcal{N}_L \cup \{candidates\}$;
  $\mathcal{N}_U = \mathcal{N}_U \setminus \{candidates\}$;
return *The set of predicted labels* $\{\hat{y}\}$

The rest concatenated vectors of unlabeled news article nodes are in the testing set. After training, the HGAT-based selector will output the probability $\mathcal{P}$ for testing samples.

Based on the probability, we propose a query strategy to select high-value candidates for active learning. As we mentioned before, a lower probability $\mathcal{P}$ indicates that the unlabeled news article node and the predicted label do not match. It also represents there is a high probability that the predicted label will be wrong. Obviously, if the news article node we query was not able to be classified correctly by the HGAT-based classifier, then it will be more "informative" than the nodes that have been correctly classified. Besides, we can make it as part of the training set in the next round of training after experts labeling, thereby correcting the misclassified nodes in the test set for similar reasons. So the query strategy is:

DEFINITION *6:* (Query Strategy): All samples in the test set will be sorted in ascending order according to the predicted probability $\mathcal{P}$, the top k candidates will be added to $\mathcal{U}_q$. Here, k denotes the query batch size.

E. Adversarial Active Optimization

In AA-HGNN, the HGAT-based classifier and the HGAT-based selector cooperate in an adversarial active manner. We adopt the iterative optimization to train these components in AA-HGNN. In each iteration, the HGAT-based classifier and the HGAT-based selector have trained alternately. Specifically, we first train the HGAT-based classifier to output the predicted labels. Then the HGAT-based selector will be trained by the predicted labels from the classifier. Based on the optimized selector, k candidates will be queried in one iteration and be added to $\mathcal{U}_q$ used as training data in the next iteration. Each time k candidates are obtained, the classification performance of the HGAT-based classifier can be improved in the next iteration. As a consequence, the credibility of predicted labels will be increased. Better predicted labels further improve the evaluation performance of the HGAT-based selector. We repeat the above iteration until the size of $\mathcal{U}_q$ exceeds the query budget b. The adversarial active optimization of AA-HGNN is described in Algorithm 1.

V. Experiments

To test the effectiveness of AA-HGNN, extensive experiments are designed and conducted on two real-world fake news datasets. We first introduce the datasets. Then experimental settings are provided. We aim to answer the following evaluation questions based on experimental results together with the detailed analysis:

- **Question 1**: Can AA-HGNN improve fake news detection performance by modeling data as a News-HIN?
- **Question 2**: Can Hierarchical Graph Attention (HGAT) mechanism handle the heterogeneity of the News-HIN effectively?
- **Question 3**: Can the active learning setting of AA-HGNN overcome the paucity of training data?
- **Question 4**: Can adversarial learning between the classifier and the selector significantly help improve the performance?

A. Dataset Description

TABLE I
Properties of the Heterogeneous Networks

	PolitiFact Network		BuzzFeed Network	
# node	article	14,055	article	182
	creator	3,634	twitter user	15,257
	subject	152	publisher	9
# link	creator-article	14,055	publisher-article	182
	article-subject	48,756	article-twitter user	25,240

We use two datasets to verify our model in experiments. The main dataset is collected from the platform with fact-checking: *PolitiFact*, which is operated by Tampa Bay Times. The news after fact-checking from *PolitiFact* mainly are the statements or news articles posted by the politicians (Congress members, White House staffs, lobbyists) and political groups. They are creators of news articles in our experiments. Regarding these news articles, *PolitiFact* will provide the original contents, fact-checking results and comprehensive fact-checking reports on the website. When presenting these news articles, the platform will categorize them into different subjects based on contents and topics. A brief description of each subject will be provided as well. The fact-checking results can indicate the credibility of corresponding news articles and take values from {True, Mostly True, Half True, Mostly False, False, Pants on Fire!}. In the *PolitiFact* dataset, 1322 news articles are marked as "Pants on Fire", while the number of news articles with "False" is 2601. Besides, 2539 "Mostly False" news articles and 2765 "Half True" news articles exist in the dataset. The number of "Mostly True" and "True" news is 2676 and 2149 respectively. We group the labels {Pants on fire, False, Mostly False} as fake news and group {True, Mostly True, Half True} as real news, the quantity of fake news is 6465 corresponding to 7590 real news. The fact-checking results will be used as the ground truth in experiments. We won't make use of comprehensive fact-checking reports in this paper. We have established a heterogeneous information network based on the *PolitiFact* dataset. The HIN includes three types of nodes: article, creator and subject and two types of links: Write (between article and creator) and Belongs to (between article and subject). In order to verify the generalization and stability of AA-HGNN, we use a public dataset *BuzzFeed*[2] from Shu et al.[38]. *BuzzFeed* contains 91 real news articles and 91 fake news articles. We also construct a HIN based on *BuzzFeed* dataset. There exist three types of nodes: article, twitter user and publisher. The key statistical data describing the HINs can be found in Table I.

B. Experimental Settings

1) Experimental Setup: In the experiments, we are able to acquire the set of news article nodes which are the target node to conduct the classification. For the *PolitiFact* dataset, the fact-checking results corresponding to news articles are used as the ground truth for model learning and evaluation. We group fact-checking results {Pants on fire, False, Mostly False} as a Fake class and group {True, Mostly True, Half True} as a Real class. Because our target is to detect fake news, we treat Fake class as the positive class and Real class as the negative class. For all comparison methods, we use 20% of news article nodes as the training set and 10% of the nodes as the validation set. In addition, the testing ratio is fixed as 10%. For AA-HGNN, we use 1000 nodes to initialize the active learning. The query budget b is 1800 and the query batch size k is 200. In this way, 2800 nodes (20% of news article nodes) are utilized to train AA-HGNN finally. *BuzzFeed* dataset has only two types of labels: True and fake, we can use it directly. The rest setting is the same as the *PolitiFact* dataset. We run the experiments on a Dell PowerEdge T630 Server with 2 20-core Intel CPUs and 256GB memory and the other Server with 3 GTX-1080 ti GPUs. Code is available at the *link*[3].

2) Data Preprocessing: Two datasets both contain textual data with different length. In order to fit to the non-sequential models, we have to transform the input features of each type of nodes into a vector with a fixed length. To deal with the problem, we use *TfidfVectorizer* in *Sklearn* package to extract features. For the *PolitiFact* dataset, the dimensions of initial features of news articles, creators, and subjects are 3000, 3109, and 191 respectively. For the *BuzzFeed* dataset, the parameter *max_features* for the news article nodes is set as 3000.

[2]https://github.com/KaiDMML/FakeNewsNet/tree/old-version

[3]https://www.dropbox.com/sh/bmgz7d1q3tq5429/AAAAcmbgKOp-gtftVWhz533ua?dl=0

3) Comparison Methods: We classify comparison methods into three categories: Graph neural network methods, Text classification methods, and Network embedding methods.

Graph neural network methods

- **AA-HGNN**: AA-HGNN is the proposed model.
- **AA-HGNN**$_{entropy}$: We keep the active learning setting of AA-HGNN, but query the candidates according to entropy. Here, we define that the closer the probability of this node being fake news to 0.5, the higher its entropy.
- **AA-HGNN**$_{random}$: Here, we query the candidates for active learning randomly.
- **HGAT-based classifier**: It is the classifier in the proposed AA-HGNN. We test the performance without active learning setting.
- **HAN** [47]: HAN employs node-level attention and semantic-level attention to capture the information from all meta-paths. In our experiments, we utilize two meta-paths (article-creator-article, article-subject-article) in HAN.
- **GAT** [41]: GAT is also an attention-based graph neural network for the node classification, but it is designed for homogeneous graphs. The News-HIN is treated as a homogeneous graph (ignore the type information) when testing the model.
- **GCN** [20]: GCN is a semi-supervised methods for the node classification in homogeneous graphs. The News-HIN is treated as a homogeneous graph when testing it.

Text classification methods

- **SVM**: SVM is a classic supervised learning model. The feature vector used for building the SVM model is extracted merely based on the news article contents with TF-IDF.
- **Text-CNN** [19]: Text-CNN is a text classification method based on convolutional neural network. It utilizes convolution filters of various sizes to capture key local features in news contents.
- **LIWC** [24]: LIWC stands for Linguistic Inquiry and Word Count, which is widely used to extract the lexicons falling into psycho-linguistic categories. It learns a feature vector from psychology and deception perspective.

Network embedding methods (NE)

- **Label Propagation (LP)** [50]: LP is merely based on the network structure. The prediction scores will be rounded and cast into labels.
- **DeepWalk** [3]: A random walk based embedding method, which is designed to deal with the homogeneous network. Based on the embedding results, we then train a logistic regression model to perform the classification of news articles.
- **LINE** [40]: LINE optimizes the objective function that preserves the local and the global network structure simultaneously. We also learn a logistic regression model to conduct the classification based on the learned embeddings.

We have also noticed some recently appeared methods for fake news detection [7], [37], [27], but did not compare them. The main consideration is the difference between the scenarios we face. In above works, they all utilize social context like user comments, but AA-HGNN aims at detecting fake news in a relatively early stage with less labeled data. We won't utilize user comments about the news or large amount of training data, because when many users have started to discuss one fake news, the bad influence of fake news has spread.

C. Experimental Results with Analysis

1) Assessing Impact of News-HIN: In order to answer **Question 1**, we first present experiment results in Table II to compare AA-HGNN with three categories of methods introduced in Section V-B3. For text classification methods SVM, LIWC and Text-CNN which use the textual information of news article nodes to do classification, we see that Text-CNN >SVM & LIWC in all metrics. This result shows that Text-CNN can better capture the important textual features in news contents by utilizing multiple convolution filters. For Network embedding methods relying on graph structures, all of them achieve a poor recall. Recall is a pretty critical metric for fake news detection problem. A low recall means we omit lots of fake news so that they will cause bad social influence, which is unexpected. A News-HIN integrates all heterogeneous available data in the form of a graph structure. Intuitively, methods (AA-HGNN, HAN) making full use of New-HIN as training data achieve better results. Through the comparison among GNNs methods, we verify that the heterogeneity of networks should be dealt with in a more effective way. If we simply treat a heterogeneous network as a homogeneous network by ignoring the type, then the results (reported by GAT, GCN) would be very disappointing. We continue to discuss performance concerning heterogeneity in the next section.

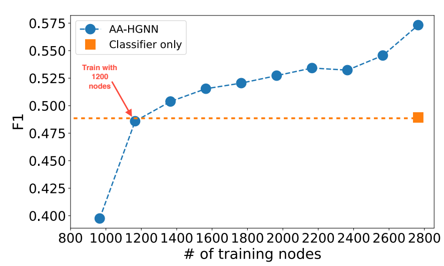


Fig. 6. The advantage in training with less labeled data.

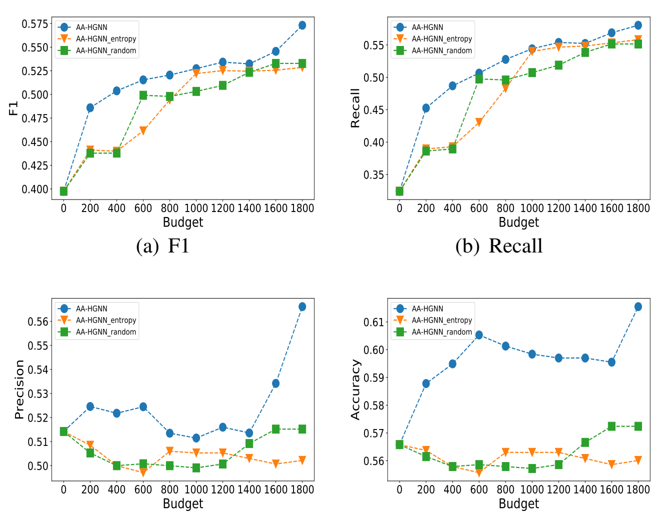
2) Methods performance on Heterogeneous graph: To answer **Question 2**, we further investigate the performance of different GNN methods besides AA-HGNN and its variants. As we utilize a heterogeneous network as source data, the heterogeneity should be handle in an effective manner. In Table II, we observe HGAT achieves the best accuracy, recall and F1. GAT and GCN get high precision but low recall. Particularly for the *PolitiFact* dataset, GCN reach 0.9688 in precision but 0.0246 in recall. This result occurs because they prefer to classify a sample as real news based on News-HIN. They are not powerful methods in fake news problem because they were originally designed for homogeneous networks. Also as a method for heterogeneous graphs, HGAT-based classifier also shows an advantage over HAN. As the basic classifier, HGAT-based classifier can handle the heterogeneity of News-HIN well.

TABLE II
PERFORMANCE COMPARISON OF DIFFERENT METHODS. THE TRAINING RATIO IS 20%.

		PolitiFact				*BuzzFeed*			
	Methods	Accuracy	Precision	Recall	F1	Accuracy	Precision	Recall	F1
Text	SVM	0.5432	0.4975	0.32	0.3894	0.5398	0.6011	0.5109	0.5523
	LIWC	0.4544	0.4415	0.23	0.3023	0.6137	0.6459	0.5885	0.6175
	Text-CNN	0.5658	0.5873	0.2824	0.3814	0.6317	0.6415	0.6233	0.6322
NE	Label Propagation	0.5796	0.7005	0.1164	0.1996	0.5867	0.6409	0.223	0.3309
	DeepWalk	0.5297	0.4639	0.2881	0.4639	0.3721	0.3083	0.4322	0.3599
	LINE	0.5012	0.4109	0.1215	0.4109	0.5899	0.6123	0.3057	0.4077
GNNs	GAT	0.5765	0.7569	0.0453	0.0854	0.5885	0.654	0.3367	0.4445
	GCN	0.5611	**0.9688**	0.0246	0.048	0.5671	0.6331	0.2674	0.3816
	HAN	0.5867	0.6802	0.2062	0.3165	0.5917	0.7163	0.4677	0.5659
	HGAT-based classifier	0.6154	0.578	0.424	0.4893	0.7022	0.6928	0.6412	0.666
	AA-HGNN$_{random}$	0.5724	0.5152	0.5515	0.5328	0.6843	0.6439	0.6123	0.6277
	AA-HGNN$_{entropy}$	0.5601	0.5022	0.5581	0.5286	0.7161	0.7088	0.6503	0.6783
	AA-HGNN	**0.6155**	0.5661	**0.5804**	**0.5732**	**0.7351**	**0.7211**	**0.6909**	**0.7057**

TABLE III
ADVERSARIAL ACTIVE LEARNING PERFORMANCE OF AA-HGNN IN *PolitiFact*

	Number of training nodes									
Metrics	1000	1200	1400	1600	1800	2000	2200	2440	2600	2800
Accuracy	0.5658	0.5878	0.6049	0.6053	0.6013	0.5984	0.597	0.597	0.5955	0.6155
Precision	0.5142	0.5246	0.5218	0.5245	0.5135	0.5115	0.516	0.5136	0.5342	0.5661
Recall	0.3241	0.4526	0.4869	0.5065	0.5277	0.5441	0.5539	0.5523	0.5688	0.5804
F1	0.3975	0.4859	0.5038	0.5154	0.5205	0.5273	0.5342	0.5323	0.5456	0.5732



(a) F1 (b) Recall

(c) Precision (d) Accuracy

Fig. 7. Performance Analysis of Query Strategy in *PolitiFact*

3) Active learning setting on scarce training data: To answer **Question 3**, we draw Figure 6 to compare the performance of HGAT-based classifier and AA-HGNN. The F1 score of the classifier shown in Figure 6 is achieved with 2800 training nodes. In comparison, AA-HGNN can outperform the classifier when being trained with 1200 labeled nodes. Besides, the score of AA-HGNN applying the active learning setting significantly increased. When the number of training nodes is 2800, the performance of AA-HGNN increase nearly 9% than the model without the active learning setting. From Table II, we can observe that AA-HGNN has the apparent advantage when using 20% training ratio, while other mehtods can not perform well due to the paucity of training data. Also, we see AA-HGNN can reach satisfactory result although the training data is even more scarce in Table III.

4) Adversarial learning impacts on Active Learning: In order to answer **Question 4**, we build two variants AA-HGNN$_{entropy}$ and AA-HGNN$_{random}$ to demonstrate the adversarial learning setting's efforts. These two varients provide different query strategies for active learning. Based on the results of comparative experiments in Figure 7, it is obvious that AA-HGNN outperforms AA-HGNN$_{entropy}$ and AA-HGNN$_{random}$ in every query batch. The adversarial learning between the classifier and the selector indeed provides an effective query strategy for the active learning. The queried candidates are of high value for the classifier, so the performance of the classifier can be significantly improved. Besides, the adversarial learning-based query strategy can consistently provide high-value candidates, as the performance of selectors also improves in adversarial learning.

VI. CONCLUSION

In this paper, we study the HIN-based fake news detection problem and propose a novel adversarial active learning-based graph neural network AA-HGNN to solve it. AA-HGNN employs a novel hierarchical attention mechanism to deal with the heterogeneity of News-HIN and learns textual and structural information simultaneously. An active learning framework is applied in AA-HGNN to enhance the learning performance, especially when facing the paucity of labeled data. A selector is trained in an adversarial manner to query high-value candidates for the active learning setting. Experiments with real-world fake news data show that our model can outperform text-based models and other graph-based models when using less labeled data. Experiments also verify the effectiveness of adversarial learning-based query strategy, which consistently queries high-value candidates to improve the performance. As an adversarial active learning-based model, AA-HGNN is ideal for detecting fake news in the early stages when lacking training data. Finally, due to the good generalizability of AA-HGNN, it has the ability to be widely used in other node classification-related applications on heterogeneous graphs, where there will be no obstacles to the transfer.

VII. Acknowledgement

This work is partially supported by NSF through grant IIS-1763365 and by FSU.

References

[1] Charu C Aggarwal, Xiangnan Kong, Quanquan Gu, Jiawei Han, and S Yu Philip. Active learning: A survey. In *Data Classification*, pages 599–634. Chapman and Hall/CRC, 2014.

[2] H. Allcott and M. Gentzkow. Social media and fake news in the 2016 election. *Journal of Economic Perspectives*, 2017.

[3] Rami Al-Rfou B. Perozzi and Steven Skiena. Deepwalk: Online learning of social representations. In *KDD*, 2014.

[4] Marco T Bastos and Dan Mercea. The brexit botnet and user-generated hyperpartisan news. *Social Science Computer Review*, 37(1):38–54, 2019.

[5] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. In *arXiv preprint arXiv:1312.6203*, 2013.

[6] Giovanni Luca Ciampaglia, Prashant Shiralkar, Luis M Rocha, Johan Bollen, Filippo Menczer, and Alessandro Flammini. Computational fact checking from knowledge networks. *PloS one*, 2015.

[7] Limeng Cui, Kai Shu, Suhang Wang, Dongwon Lee, and Huan Liu. defend: A system for explainable fake news detection. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 2961–2964. ACM, 2019.

[8] Quanyu Dai, Qiang Li, Jian Tang, and Dan Wang. Adversarial network embedding. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[9] Sanjoy Dasgupta, Adam Tauman Kalai, and Claire Monteleoni. Analysis of perceptron-based active learning. In *International Conference on Computational Learning Theory*, pages 249–263. Springer, 2005.

[10] Xin Dong, Evgeniy Gabrilovich, Geremy Heitz, Wilko Horn, NiLao, Kevin Murphy, Thomas Strohmann, Shaohua Sun, and Wei Zhang. Knowledge vault: A web-scale approach to probabilistic knowledge fusion. In *KDD*, 2014.

[11] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[12] Marco Gori, Gabriele Monfardini, and Franco Scarselli. A new model for learning in graph domains. In *IJCNN*, 2005.

[13] Han Guo, Juan Cao, Yazi Zhang, Junbo Guo, and Jintao Li. Rumor detection with hierarchical social attention network. In *CIKM*, 2018.

[14] Aditi Gupta, Hemank Lamba, Ponnurangam Kumaraguru, and Anupam Joshi. Faking sandy: characterizing and identifying fake images on twitter during hurricane sandy. In *Proceedings of the 22nd international conference on World Wide Web*, pages 729–736. ACM, 2013.

[15] Shashank Gupta, Raghuveer Thirukovalluru, Manjira Sinha, and Sandya Mannarswamy. Cimtdetect: A community infused matrix-tensor coupled factor- ization based method for fake news detection. In *arXiv preprint arXiv:1809.05252*, 2018.

[16] William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *NIPS*, 2017.

[17] Binbin Hu, Yuan Fang, and Chuan Shi. Adversarial learning on heterogeneous information networks. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019.

[18] Zhiwei Jin, Juan Cao, Yongdong Zhang, and Jiebo Luo. News verification by exploiting conflicting social viewpoints in microblogs. In *AAAI*, 2016.

[19] Yoon Kim. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882*, 2014.

[20] Thomas N. Kipf and Max Welling. Semi-supervised classi cation with graph convolutional networks. In *ICLR*, 2017.

[21] Yan Li and Jieping Ye. Learning adversarial networks for semi-supervised text classification via policy gradient. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1715–1723. ACM, 2018.

[22] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. Gated graph sequence neural networks. In *ICLR*, 2016.

[23] Marcelo Mendoza, Barbara Poblete, and Carlos Castillo. Twitter under crisis: Can we trust what we rt? In *Proceedings of the first workshop on social media analytics*, pages 71–79. ACM, 2010.

[24] J. Pennebaker, R. Boyd, K. Jordan, and K. Blackburn. The development and psychometric properties of liwc. *Technical Report*, 2015.

[25] Veronica Perez-Rosas, Bennett Kleinberg, Alexandra Lefevre, and Rada Mihalcea. Automatic detection of fake news. In *arXiv preprint arXiv:1708.07104*, 2017.

[26] Martin Potthast, Johannes Kiesel, Kevin Reinartz, Janek Bevendor, and Benno Stein. A stylometric inquiry into hyperpartisan and fake news. In *arXiv preprint arXiv:1702.05638*, 2017.

[27] Feng Qian, Chengyue Gong, Karishma Sharma, and Yan Liu. Neural user response generator: Fake news detection with collective user intelligence. In *IJCAI*, pages 3834–3840, 2018.

[28] K. Rapoza. Can 'fake news' impact the stock market? *Forbes News*, 2017.

[29] Yuxiang Ren, Charu Aggarwal, and Jiawei Zhang. Activeiter: Meta diagram based active learning in social networks alignment. *IEEE Transactions on Knowledge and Data Engineering*, 2019.

[30] Neil Rubens, Mehdi Elahi, Masashi Sugiyama, and Dain Kaplan. Active learning in recommender systems. In *Recommender systems handbook*, pages 809–846. Springer, 2015.

[31] Victoria L Rubin and Tatiana Lukoianova. Truth and deception at the rhetorical structure level. *Journal of the Association for Information Science and Technology*, 2015.

[32] Natali Ruchansky, Sungyong Seo, and Yan Liu. Csi: A hybrid deep model for fake news detection. In *CIKM*, 2017.

[33] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks 20, 1*, 2009.

[34] Francesc Serratosa and Xavier Cortés. Interactive graph-matching using active query strategies. *Pattern Recognition*, 48(4):1364–1373, 2015.

[35] Baoxu Shi and Tim Weninger. Discriminative predicate path mining for fact checking in knowledge graphs. *Knowledge-Based Systems*, 2014.

[36] C. Shi, Y. Li, J. Zhang, Y. Sun, and P. Yu. A survey of heterogeneous information network analysis. *TKDE*, 2017.

[37] Kai Shu, Suhang Wang, and Huan Liu. Beyond news contents: The role of social context for fake news detection. In *WSDM*, 2019.

[38] Kai Shu, Suhang Wang, and Huan Liu. Beyond news contents: The role of social context for fake news detection. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*, pages 312–320, 2019.

[39] Y. Sun and J. Han. Mining heterogeneous information networks: a structural analysis approach. *KDD Explorations*, 2012.

[40] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *WWW*, 2015.

[41] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio. Graph attention networks. In *ICLR*, 2018.

[42] Soroush Vosoughi, Deb Roy, and Sinan Aral. The spread of true and false news online. *Science*, 359(6380):1146–1151, 2018.

[43] Jun Wang, Lantao Yu, Weinan Zhang, Yu Gong, Yinghui Xu, Benyou Wang, Peng Zhang, and Dell Zhang. Irgan: A minimax game for unifying generative and discriminative information retrieval models. In *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*, pages 515–524. ACM, 2017.

[44] Keze Wang, Dongyu Zhang, Ya Li, Ruimao Zhang, and Liang Lin. Cost-effective active learning for deep image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 27(12):2591–2600, 2016.

[45] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Peng Cui, P. Yu, and Yanfang Ye. Heterogeneous graph attention network. In *WWW*, 2019.

[46] K. Xu, W. Hu, J. Leskovec, and S. Jegelka. How powerful are graph neural networks? In *ICLR*, 2019.

[47] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy. Hierarchical attention networks for document classification. In *NAACL*, 2016.

[48] Jiawei Zhang, Limeng Cui, Yanjie Fu, and Fisher B Gouza. Fake news detection with deep diffusive network model. In *arXiv preprint arXiv:1805.08751*, 2018.

[49] Xinyi Zhou and Reza Zafarani. Fakenews: A survey of research, detection methods, and opportunities. In *arXiv preprint arXiv:1812.00315*, 2018.

[50] Xiaojin Zhu and Zoubin Ghahramani. Learning from labeled and unlabeled data with label propagation. Technical report, 2002.