The Pennsylvania State University
The Graduate School

UNDERSTANDING AND DETECTING ONLINE MISINFORMATION WITH AUXILIARY DATA

A Dissertation in
Information Sciences and Technology
by
Limeng Cui

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Doctor of Philosophy

May 2022

The dissertation of Limeng Cui was reviewed and approved* by the following:

Dongwon Lee
Professor of College of Information Sciences and Technology
Dissertation Advisor, Chair of Committee

Anna Cinzia Squicciarini
Professor of College of Information Sciences and Technology

Amulya Yadav
Assistant Professor of College of Information Sciences and Technology

Lin Lin
Assistant Professor of Department of Statistics

Mary Beth Rosson
Professor of College of Information Sciences and Technology
Director of Graduate Programs for College of Information Sciences and Technology

Abstract

The Internet provides great convenience for users to access, create, and share diverse information and promotes the spread of misinformation. The cheap to produce, easily accessible fake content online can easily shape public perception and cause detrimental societal effects. Thus, how to effectively detect online misinformation and attenuate its effect has gained much attention in recent years. Recent achievements of misinformation detection methods have shown promising results. However, there still exhibits enormous challenges due to the multi-modality, interpretability, and costs of human annotation in this problem.

In order to address the above-mentioned issues, we can leverage various types of information from different perspectives. For example, user engagements over news articles, including posts and comments, contain justification about the news article. Since these auxiliary data can provide rich contextual information for more accurate and interpretable detection, it is essential to understand and detect misinformation by integrating multiple sources.

This task is challenging because the proposed methods should be able to exploit auxiliary supervision for learning with limited data and effectively detect misinformation. In this regard, three different scenarios related to detecting and understanding online misinformation are discussed in this dissertation. First, the rich information available in user comments on social media suggests that we could investigate whether the latent sentiments hidden in user comments can help distinguish fake news from reliable content. A sentiment-aware fake news detection method is proposed to account for users' latent sentiments. Second, users' lack of sufficient prior knowledge suggests the misinformation detection method to reflect the interpretability of the results than prediction labels. A knowledge-guided model is proposed to solve the challenging limitation on social contexts in domains like healthcare. Third, human labeling is time-consuming and costly. This problem is further exacerbated in misinformation detection scenarios, when the datasets are imbalanced. A novel active learning framework is proposed to improve the model performance at a lower cost in detecting fraudsters in online websites. Finally, this work is closed by future directions on intervening the dissemination of misinformation at an early stage. When the labeled data is limited in a new genre or language, transferring the knowledge from high-resource domains to the new, low-resource domain is a promising solution. The findings of this dissertation significantly expand the boundaries of online misinformation detection and inspire improvements on general machine learning methods.

Table of Contents

List of Figures **viii**

List of Tables **ix**

Acknowledgments **xi**

Chapter 1
Introduction **1**
1.1 Background and Scope 1
1.2 Subproblems in Different Scenarios 2
1.2.1 Sentiment Aware Misinformation Detection 2
1.2.2 Knowledge Graph based Misinformation Detection 2
1.2.3 Active Learning based Fraud Detection 3
1.3 Overview of this Proposal 3

Chapter 2
Sentiment Aware Misinformation Detection **4**
2.1 Introduction 4
2.2 Related Work 6
2.2.1 Fake News Detection 6
2.2.1.1 Single-modal based Methods 6
2.2.1.2 Multi-modal based Methods 7
2.2.2 Sentiment Analysis 8
2.3 Preliminary Data Analysis 8
2.3.1 Datasets 8
2.3.2 User Sentiment Analysis Toward Fake and Real News 9
2.4 Proposed Method 11
2.4.1 Multi-Modal Feature Extractor 12
2.4.2 Adversarial Learning 13
2.4.3 Modeling Sentiment Correlation 14

2.4.4 Fake News Detector 15
2.4.5 The Proposed Method: SAME 15
2.5 Experimental Validation 16
2.5.1 Compared Methods 16
2.5.2 Evaluation Metrics 17
2.5.3 Performance Comparison 20
2.5.4 Ablation Study 20
2.6 Conclusion and Future Work 21

Chapter 3
Knowledge Graph based Misinformation Detection 23
3.1 Introduction 23
3.2 Related Work 26
3.3 Problem Formulation 27
3.4 Methodology 29
3.4.1 Information Propagation Net 30
3.4.2 Knowledge Guided Embedding Layers 32
3.4.3 Model Prediction 33
3.4.4 Training and Inference with DETERRENT 34
3.5 Experiments 34
3.5.1 Datasets 35
3.5.2 Baselines 35
3.5.3 Experimental Setup 38
3.5.3.1 Metrics 38
3.5.3.2 Implementation Details 38
3.5.4 Misinformation Detection (RQ1) 38
3.5.4.1 Overall Comparison 38
3.5.4.2 Performance Comparison w.r.t. Article Types 39
3.5.5 Ablation Analysis (RQ2) 40
3.5.5.1 Effects of Network Depth 40
3.5.5.2 Effects of Attention Mechanisms and Negative Relations 41
3.5.6 Case Study (RQ3) 43
3.6 Conclusion 45

Chapter 4
Fraud Detection with Limited Data 46
4.1 Introduction 46
4.2 Related Work 49
4.2.1 Active Learning on Graphs 49
4.2.2 Graph Coarsening 49

4.2.3 Imbalanced Learning 50
4.3 The Proposed Method: ALLIE 51
4.3.1 Task Description 51
4.3.2 Framework 52
4.3.2.1 Reinforcement Learning Architecture 53
4.3.2.2 Policy Network Design 55
4.3.2.3 Robust Classification 56
4.3.2.4 Training and Inference 57
4.4 Experiments 58
4.4.1 Experimental Setting 59
4.4.1.1 Datasets 59
4.4.1.2 Implementation details 59
4.4.1.3 Baselines 60
4.4.1.4 Evaluation measures 61
4.4.2 EQ1: Performance on Public Datasets 61
4.4.2.1 Balanced Setting 61
4.4.2.2 Imbalanced Setting 61
4.4.3 EQ1: Performance on e-commerce dataset 66
4.4.3.1 Classification on Buyers 66
4.4.3.2 Classification on Reviews 66
4.4.4 EQ2: Ablation Study 68
4.4.4.1 Effect of reward functions 68
4.4.4.2 Effect of graph coarsening and balancing strategies . 68
4.4.5 EQ3: Hyperparameter Sensitivity Analysis 70
4.4.5.1 Performance under different initial training set sizes . 70
4.4.5.2 Performance under different query budgets 70
4.5 Conclusion 70

Chapter 5

Future Work **72**
5.1 Transfer Learning for Misinformation Detection 72
5.2 Thread Recommendation in Online Health Forums 73

Bibliography **74**

List of Figures

2.1 Sentiment polarity distribution of different stances (“Agree”, “Discuss” and “Disagree”) in users’ comments. 10

2.2 Multi-modal Embedding (SAME) accepts input in a triplet of news publisher, user and news, and processes them through a deep network: (1) three different networks to deal with the triplet including news publishers, users, and news; (2) adversarial mechanism to enforce the same distribution between different modalities; and (3) a novel hybrid similarity loss to model the user’s representation and incorporate it into the proposed framework. 12

3.1 Healthcare article examples and related triples from a medical knowledge graph(KG). The triples can either enhance or weaken the augments in the articles. 24

3.2 Illustration of the proposed DETERRENT model. The left subfigure shows the Knowledge Guided Embedding Layers of DETERRENT, and the right subfigure presents the Information Propagation Net of DETERRENT. The Information Propagation Net is performed on the unified graph of Article-Entity Bipartite Graph and Medical Knowledge Graph, which has positive (in black) and negative (in red) relations. . 29

3.3 Performance comparison over the length of article types on two datasets. The background histograms indicate the length of each article; meanwhile, the lines demonstrate the performance w.r.t. F1 score. 41

3.4 The explainable triples captured by DETERRENT. 43

3.5 The visualization with attention wights. 44

3.6 The attention weight analysis indicates that positive relations contribute more to fact, and negative relations contribute more to misinformation. 44

4.1 The overview of ALLIE with two main parts: policy network and fraud detector. 51
4.2 Performance gain comparison over variants on the buyer classification task. The background histograms indicate the F1 gain over AGE of each variant. The lines indicate the running time (in seconds). . . . 69
4.3 Performance gain over Random with different training set sizes and query budgets. 69

List of Tables

2.1 The statistics of the two real-world datasets. 9
2.2 The sentiment polarity distribution under news. 10
2.3 Performance comparison on the two datasets. The best results are listed in bold. 18
2.4 Comparison of variants of SAME on two datasets. The best results are listed in bold. 19

3.1 Statistics of datasets 35
3.2 Performance Comparison on Diabetes and Cancer datasets. DETERRENT outperforms all state-of-the-art baselines including knowledge graph based and article contents based methods. 36
3.3 Effects of the network depth 42
3.4 Ablation study of DETERRENT demonstrated the advantage of the attention mechanisms and modeling both positive and negative relations. 42

4.1 Statistics of citation graph datasets. 59

4.2 Summary of the e-commerce dataset. The dataset is heavily subsampled, and is used to show the efficacy of ALLIE on a real world use case. 59
4.3 Node classification performance (Macro F1 and Micro F1 ±Std) on the balanced setting of citation graph datasets. The best and 2nd best are noted in bold font and underlined, respectively. 62
4.4 Node classification performance (Macro F1 and Micro F1 ±Std) on the imbalanced setting of citation graph datasets. 63
4.5 Buyer classification performance relative change (Precision, Recall and F1 ±Std) on the e-commerce dataset. 64
4.6 Review classification performance relative change (Precision, Recall and F1 ±Std) on the e-commerce dataset. 65
4.7 Effect of reward functions (Macro F1 and Micro F1 ±Std) on imbalanced citation graph datasets. 67

Acknowledgments

My sincerest thanks are extended to my advisor, Dr. Dongwon Lee, for guiding and supporting me over the years. You set an example of excellence as an advisor, researcher and instructor. All other committee members, Dr. Anna Squicciarini, Dr. Amulya Yadav and Dr. Lin Lin, are thanked for their valuable feedback and comments.

I would like to thank all my mentors, collaborators and colleagues who contributed to my research. In particular, I would like to thank Dr. Cao (Danica) Xiao, Dr. Aaron Jaech, Dr. Xianfeng Tang, Dr. Sumeet Katariya and Dr. Nikhil Rao. You have always given me amazing advice. I will forever treasure the insights you shared with me and the skills you taught me.

I would also like to thank my labmates, fellow graduate students and friends. Your support and discussion have been absolutely invaluable.

I would especially like to thank my family for their unconditional love and constant encouragement I have received over the years.

This work was supported in part by NSF #1940076 and #1909702. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Dedication

This dissertation is dedicated to my parents.

Chapter 1 Introduction

In recent years, due to the explosive growth of online contents, misinformation for different political agendas and commercial gains has been coming out in a great amount and widespread online. During the US presidential election in 2016, for instance, misinformation has caused a significant social impact on the election results. For example, “Pizzagate”, a scandal that never was true, quickly went viral on multiple social media platforms. A report on the 2016 election indicates that fake news websites rely on online social media for 48% of traffic, which is a much higher share than that of other sources [1]. Therefore, to mitigate the problems of misinformation, how to detect it effectively has become an important research problem, which will be the main task of this paper.

1.1 Background and Scope

In this section, some related areas are introduced as background and compared with this proposal to clarify the scope here.

How to effectively detect misinformation and prevent its diffusion online has gained much attention in recent years. To provide accurate misinformation detection, it is often useful to take an auxiliary source (e.g., social context and knowledge base) into consideration. There are two aspects of complementary information in misinformation detection.

First, social contexts such as users’ engagements are complementary information that can improve detection performance and derive explanations. We can use the rich information available in user comments to distinguish misinformation from reliable content. For example, user’s comments such as “`I agree..she is a rock star`” or “`No.`

`It’s a fake news story specifically targeting ‘conservative readers’.`”, may potentially add/remove different degrees of credibility to the news.

Second, a knowledge graph can provide additional information to misinformation detection, as social context information is not always available and may not be useful. For example, due to the lack of sufficient professional knowledge, users seldom respond to healthcare information and cannot give accurate comments.

In addition to leveraging multi-source data, we can also exploit the labeled data and pre-trained models to improve the current model performance at a lower cost. For example, as it is often time-consuming, labor-intensive, and expensive to acquire sufficient labeled data for misinformation detection, we can use active learning to select the most relevant example for human labelers.

1.2 Subproblems in Different Scenarios

Based on the general idea of misinformation detection, in this proposal, I propose to consider different aspects as subproblems in different scenarios.

1.2.1 Sentiment Aware Misinformation Detection

In this scenario, I propose to study the problem of fake news detection employing the sentiment analysis idea in user comments. This problem is technically difficult for two reasons. On one hand, the learned features of user’s representation are usually high dimensional and sparse, which cannot be processed by traditional methods. On the other hand, as each modality has an intrinsically different distribution, it is challenging to fuse user’s representation with others. I have a primary study done to solve this problem [2].

1.2.2 Knowledge Graph based Misinformation Detection

This scenario is a natural extension from Sec.1.2.1. Due to the lack of sufficient professional knowledge, users seldom respond to information in specific domains such as healthcare, which makes existing social context-based methods less applicable. In the meanwhile, a knowledge graph, which is constructed from verified sources can be used as an effective auxiliary for misinformation detection. Hence, we study a novel problem of explainable healthcare misinformation detection by leveraging the medical knowledge graph. This is a non-trivial task due to two reasons: On the one hand, healthcare information/texts and medical knowledge graph cannot be directly integrated, as they have different data structures. On the other hand, social network analysis techniques are not applied to the medical knowledge graph. This is one future direction.

1.2.3 Active Learning based Fraud Detection

This scenario is an extension from Sec. 1.2.1 and Sec.1.2.2. Human labeling is time-consuming and costly. This problem is further exacerbated in extremely imbalanced class label scenarios, such as detecting fraudsters in online websites (e.g., Amazon, Walmart). However, existing methods for active learning for graph data often assumes that both data and label distributions are balanced. The challenge of this problem lies in how to select the most relevant example for human labelers to improve the model performance at a lower cost.

1.3 Overview of this Proposal

The next chapter introduces some related works to the general problem proposed. And then each chapter contains research results or plans for each of first two subproblems in Sec.1.2, followed by a chapter describing future research plan as the latter two subproblems in Sec.1.2. Finally, a conclusion is followed.

Chapter 2 Sentiment Aware Misinformation Detection

2.1 Introduction

Fake news is a kind of misinformation that is spread deliberately to manipulate public opinion, through traditional mass media and recent online social media. As verified information about newly emerged and time-critical events can be hardly obtained in a timely manner, it is critical to involve social contexts regarding fake news detection. In this chapter, I will investigate whether the latent sentiment hidden in user comments can potentially help distinguish fake news from reliable content.

Existing methods for detecting fake news can be generally categorized into two categories based on the heterogeneity of the data, i.e., single-modal based and multi-modal based. In single-modal based methods, a single type of, often textual, information such as contents, profiles and descriptions are used. For instance, [3] exploits the linguistic features of misinformation by comparing real news with fake news. Similarly, [4] conducts fake news detection by evaluating the consistency between the body and its claim given a news article. Note that as the content type of news is *not* limited to only text, other data types such as images or videos could also be utilized. In particular, in social media, fake news often comes with multi-modality data including manipulated images, fake videos, or user comments, all of which provide rich information for detecting fake news. As such, multi-modality based fake news detection has gained increased attention. For example, [5] proposes a Recurrent Neural Network (RNN) with an attention mechanism to fuse multi-modal data from tweets for rumor detection. In addition, [6] proposes the Event Adversarial Neural Networks (EANN), which integrates multi-modal features of images and texts and removes event-specific features via discriminator.

In addition to the issue of modality, another important idea is to exploit the latent sentiment in user comments. Although user's viewpoint has been proved to be useful in fake news detection [7], there are few studies on the impact of user's sentiment. User's comments such as "`I agree..she is a rock star`" or "`No. It's a fake news story specifically targeting 'conservative readers'.`", may potentially add/remove different degrees of credibility to the news in question. Therefore, toward the detection of fake news, we propose to explore to employ both the sentiment analysis in user comments as well as multi-modal fake news data.

In an attempt to solve this problem, these are several challenges. As for incorporating user's sentiment into a detection procedure with multi-modal data, the first step is to represent a user. Each user may comment on or "like" a particular type of news. Such a representation can be measured by the correspondence between user's historical interest and type of news. However, this problem is technically difficult for two reasons. On one hand, the learned features of user's representation are usually high dimensional and sparse, which cannot be processed by traditional methods. On the other hand, as each modality has an intrinsically different distribution, it is challenging to fuse user's representation with others. For example, a user's sentiment representation is sparse while the image feature is naturally dense, causing a mismatch.

Overcoming these challenges, in this paper, we present a novel method, named as Sentiment-Aware Multi-modal Embedding (SAME), with the emphasis on both sentiment and multi-modality. We propose to use an end-to-end deep architecture to mitigate the heterogeneity introduced by multi-modal data and capture the representation of user's sentiment better. To be specific, first, we use different networks to deal with the triplet relationship among news publishers, users, and news. Second, an adversarial mechanism is introduced to preserve the semantic similarity and enforces the representation consistency between different modalities. To our best knowledge, this is the first attempt to utilize adversarial learning to find semantic correlations between different modalities in news content. Third, we model a user's sentiment and incorporate it into the proposed framework.

Our main contributions are as follows:

- We propose an end-to-end deep framework to integrate different features of news content for fake news detection. An adversarial mechanism is added to preserve semantic relevance and representation consistency across different modalities.

- We validate the effectiveness of user sentiment through statistical analysis and use users' sentimental polarities to facilitate fake news detection.

- We empirically demonstrate that our proposed method, SAME, significantly outperforms five state-of-the-art baselines in detecting fake news on social media using two real-world benchmark datasets.

2.2 Related Work

In this section, we briefly review two related topics, i.e., fake news detection and sentiment analysis.

2.2.1 Fake News Detection

In recent years, researchers have proposed a number of methods for fake news detection. Interested readers are referred to [8, 9] for further information. From the perspective of information used, fake news detection methods can be roughly divided into two categories: single-modal and multi-modal based methods.

2.2.1.1 Single-modal based Methods

For single-modal based methods, existing works [3, 4, 10, 11] mainly analyze the textual contents of news, including the headline and news content, and extract the characteristics of fake news. Some researchers use methods in linguistics to distinguish the fake news from the real ones. Others check the consistency between the news title and content. For example, Rashkin et al. [3] specifically focus on political coverage verification and fake news detection. They propose to exploit the linguistic features of misinformation by comparing real news with fake news such as satire, hoaxes, and propaganda. Jin et al. [10] assume the images plays a very important role in the news propagation on the microblog. The distribution patterns between images of real and fake news are quite different. Thus, they extract the image features from two aspects, including visual content and statistics. In literature [4], Bhatt et al. conduct fake news detection by evaluating the consistency between the body and its claim given a news article. Statistical and external features are used to build a unified classifier for fake news detection. As the content type of news is not limited to text, the above methods do not fully exploit the multi-modal data such as image, video, and network. Thus, they do not yield satisfying results compared with multi-modal based methods.

2.2.1.2 Multi-modal based Methods

In social media, besides the textual features, news often includes various types of data, which provides more comprehensive features for detecting fake news. Thus, investigating multi-modal data for fake news detection is attracting increasing attention [5–7, 12–15]. A survey on different content types of news and their impacts on readers can be found in [16].

In general, multi-modal based fake news detection focus on extracting features from news content, including news publisher, textual contents and image/video. By using the three types of features mentioned, different kinds of news representations can be built, which capture discriminative aspects of news. In multimedia based methods, researchers usually use deep networks to capture both visual and textual information of news and apply classification models to distinguish fake news from the real ones. In the literature [5], the authors propose an attention based Recurrent Neural Network to fuse the multi-modal data from tweets for rumor detection. An attention mechanism is added to find the correlations between images and texts. The architecture of Event Adversarial Neural Networks (EANN) is proposed in literature [6]. Both text and image in an article are taken into consideration. The authors train an event discriminator in order to eliminate the effects of the event-specific features and maintain the common features among all these studied events.

Despite the success of multi-modal based fake news detection approaches, few of them explicitly model user sentiments towards news for fake news detection; while sentiments are very strong signal which have great potential for improving fake news detection performance. Therefore, in this paper, we investigate a novel problem of exploring user sentiments for fake news detection with multi-modal data.

2.2.2 Sentiment Analysis

Users' opinions or sentiments towards posts or products in social media have been demonstrated to be very effective for many social media mining tasks such as user rating prediction [17,18], recommender system [19] and stock movement prediction [20]. Detecting user sentiments or stances has become a popular task. In literature [21] authors conduct user's belief classification and in literature [22] authors conduct stance detection. Zhang et al. [11] focus on news stance detection. The proposed model takes the headline and body of an article, and generates the probabilities of four news stances including "agree", "disagree", "discuss" and "unrelated". The authors use ranking-based to address the problem brought by classification-based algorithms that a clear distinction exists between any two stances. In literature [23], the authors predict the stance of a set of texts representing facts with respect to a given claim by using end-to-end memory networks.

As sentiment features have shown promising results in improving the performance of news stance detection, we introduce sentiment features into the fake news detection task.

2.3 Preliminary Data Analysis

Users can express their emotions or opinions, through comments such as sensational or skeptical reactions [24]. These features are useful when detecting fake news. In this section, we conduct preliminary data analysis to demonstrate that users' sentiments towards real news and fake news are statistically different, which lays a foundation for integrating sentiments for fake news detection. Next, we first introduce the datasets followed by preliminary data analysis.

2.3.1 Datasets

For preliminary data analysis, we adopt two widely used multi-modal fake news detection datasets, i.e., PolitiFact and GossipCop, which are publicly available from a fake news dataset repository FakeNewsNet[1] [24]. For both datasets, each news entity contains news content, corresponding images, users' retweets/replies and news

[1] https://github.com/KaiDMML/FakeNewsNet

Table 2.1: The statistics of the two real-world datasets.

Dataset	Politifact	GossipCop
# Real News	624	16,817
# Fake News	432	5,323
# User	558,937	1,390,131
# User Replies	552,698	379,996

profiles (source, publisher and keywords). Each news has 0 to 1,000 user comments. Some users didn't leave a comment when they retweet, so we excluded such kind of user engagement data.

- PolitiFact is a fact-checking website that targets on political news. It rates the authenticity of claims by elected officials and others. The two datasets are crawled from Twitter in order to get users' comments. It contains 432/624 (fake/real) news.

- GossipCop is a fact-checking website for celebrity reporting. It investigates the credibility of entertainment stories in magazines, newspapers and social media, to ascertain whether they are real or not. It contains of 5,323/16,817 (fake/real) news.

The statistics of the datasets are summarized in Table 2.1.

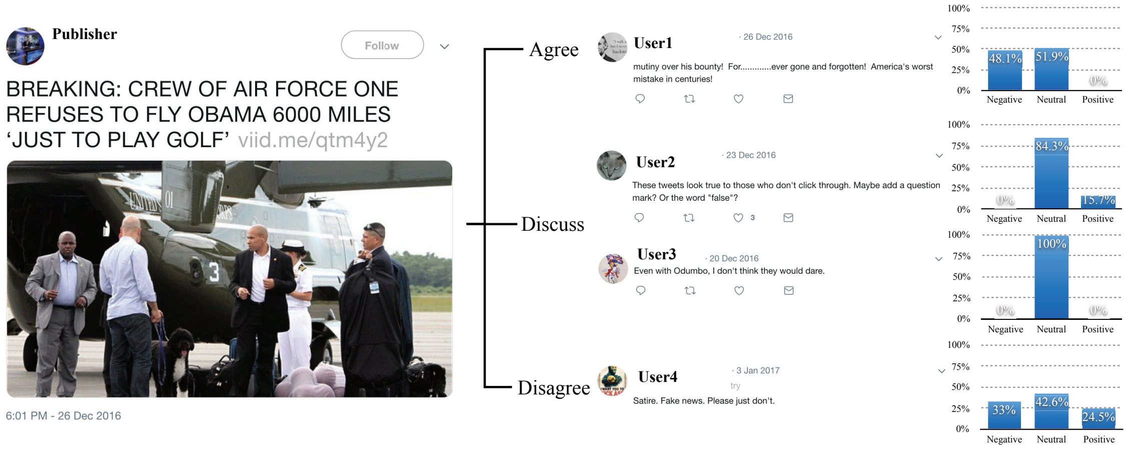
2.3.2 User Sentiment Analysis Toward Fake and Real News

Intuitively, the comments under fake news can be roughly divided into three classes: (1) Agree (from users who believe in the news); (2) Discuss (from users who doubt the authenticity of the news); and (3) Disagree that the original news is false news (from users who do not believe in the news).

Usually, the first and third types of comments contain polarized emotions ("Negative" and "Positive"), which can be seen from **User1** and **User4** in Figure 2.1. The second type of comment does not contain such strong emotions. The sentiment is more neutral in skeptical comments or discussions.

Here we perform the sentiment analysis on the users' comments with VADER[2] [25], which is a lexicon and rule-based tool to predict the sentiment expressed on social

[2] https://github.com/cjhutto/vaderSentiment



Publisher

BREAKING: CREW OF AIR FORCE ONE REFUSES TO FLY OBAMA 6000 MILES 'JUST TO PLAY GOLF' viid.me/qtm4y2

6:01 PM - 26 Dec 2016

User1 · 26 Dec 2016

mutiny over his bounty! For............ever gone and forgotten! America's worst mistake in centuries!

User2 · 23 Dec 2016

These tweets look true to those who don't click through. Maybe add a question mark? Or the word "false"?

User3 · 20 Dec 2016

Even with Odumbo, I don't think they would dare.

User4 · 3 Jan 2017

Satire. Fake news. Please just don't.

Figure 2.1: Sentiment polarity distribution of different stances ("Agree", "Discuss" and "Disagree") in users' comments.

Table 2.2: The sentiment polarity distribution under news.

		Negative	Neutral	Positive
PolitiFact	Fake News	12.6	73.2	14.2
	Real News	9.6	77.9	12.5
GossipCop	Fake News	9.8	69.2	21.0
	Real News	8.9	74.4	16.7

media. For each news piece, we obtain all the replies for this news and apply VADER to predict the sentiment as negative, neutral or positive. As can be seen from Table 2.2, users' comments under fake news often contain more sentiment polarities and are less neutral.

To statistically verify our observation, we conduct hypothesis testing. Positive, neutral and negative sentiment polarities are defined by VADER. For each dataset, two equal-sized collections of tweets are chosen. Each of them contains 50 tweets and each tweet has at least 50 comments. One collection contains the comments randomly selected from fake news, while the other contains comments randomly selected from real news. We use two vectors $\mathbf{s}_f$ and $\mathbf{s}_r$ to denote the sentiment polarities of two groups respectively, where the sentiment polarity is the sum of positive and negative sentiment polarity. A two-sample one-tail t-test is conducted to validate whether there is sufficient statistical correlation to support the hypothesis that the sentiment polarity of the first collection is greater than that of the second.

Let μ_f be the mean of sentiment polarities of the comment in the fake news collection and μ_r the mean of real news. The null hypothesis is H_0, and the alternative hypothesis is H_1. Here the hypothesis of interest can be expressed as:

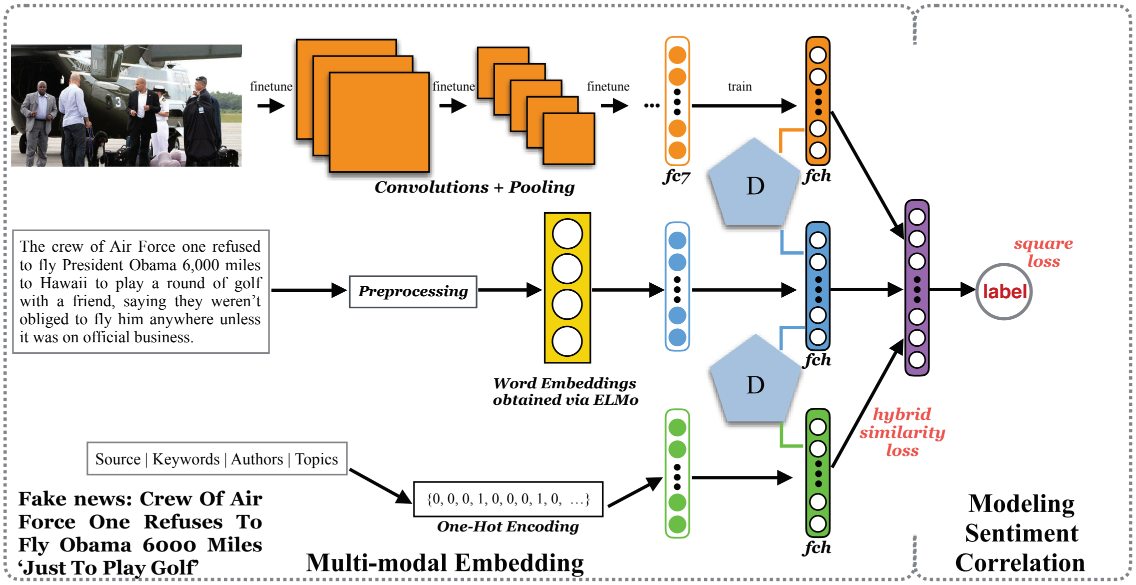
$$\begin{aligned} H_0 &: \mu_f - \mu_r \leq 0 \\ H_1 &: \mu_f - \mu_r > 0 \end{aligned} \tag{2.1}$$

The results show that there is statistical evidence on Politifact dataset, with $t = -1.6927$, $df = 98$, $p - value = 0.04684$ to reject the H_0 hypothesis, which is the evidence that the sentiment polarity of comments under fake news is greater than under real news. And we also find statistical evidence on GossipCop dataset, with $t = -2.1012$, $df = 98$, $p - value = 0.01909$.

2.4 Proposed Method

As we have validated the impact of user's sentiment, in this section, we introduce the proposed multi-modal embedding model by incorporating such information for fake news detection. In multi-modal fake news, we have four objects: news image, content, profile and user comments. The news is multi-modal data which consists of three modalities. Assume that we have N training pairs $\mathbf{D} = \{\mathbf{T}_i, r_i\}_{i=1}^N$ in which $\mathbf{T}_i$ denotes news i and $r_i \in \{0, 1\}$ denotes its ground truth label. Further, let $\mathbf{T}_i = (\mathbf{x}_i, \mathbf{y}_i, \mathbf{z}_i)$, where $\mathbf{x}_i$ denotes the feature vectors of the image modality, $\mathbf{y}_i$ denotes the feature vectors of the text modality and $\mathbf{z}_i$ can be the one-hot code of news profile related to news i. In addition, we are also given a similarity matrix $\mathbf{S}$, where S_{ij} evaluates the similarity of news i and news j. The similarity is defined by the shared user's sentiment. For example, we can say that news i and news j are similar if they share multiple sentiment words.

We first introduce how to learn the latent news presentation from the multi-modal news data by learning a joint embedding function $f(\mathbf{T}_i)$ map the news to space $\mathbb{R}^M$, where different modalities are distributed consistently. To preserve the similarity matrix $\mathbf{S}$, the distance between embedding vectors $\mathbf{h}_i = f(\mathbf{T}_i)$ and $\mathbf{h}_j = f(\mathbf{T}_j)$ should be small if S_{ij} is relatively large. Thus, a hybrid similarity loss is proposed to embed the user's sentiment into the model. The objective is to maximize the similarity between similar news triplets and minimize it between all dissimilar news triplets. Finally, once each data source is mapped to $\mathbb{R}^M$, we use the embedding vector $\mathbf{h}_i$ to predict the news label r_i.


finetune
finetune
finetune
train
Convolutions + Pooling
fc7
fch
D
Preprocessing
Word Embeddings obtained via ELMo
fch
D
square loss
label
hybrid similarity loss
Source | Keywords | Authors | Topics
{0, 0, 0, 1, 0, 0, 0, 1, 0, …}
One-Hot Encoding
fch
Fake news: Crew Of Air Force One Refuses To Fly Obama 6000 Miles 'Just To Play Golf'
Multi-modal Embedding
Modeling Sentiment Correlation

The crew of Air Force one refused to fly President Obama 6,000 miles to Hawaii to play a round of golf with a friend, saying they weren't obliged to fly him anywhere unless it was on official business.

Figure 2.2: Multi-modal Embedding (SAME) accepts input in a triplet of news publisher, user and news, and processes them through a deep network: (1) three different networks to deal with the triplet including news publishers, users, and news; (2) adversarial mechanism to enforce the same distribution between different modalities; and (3) a novel hybrid similarity loss to model the user's representation and incorporate it into the proposed framework.

2.4.1 Multi-Modal Feature Extractor

The hybrid deep architecture for learning multi-modal correlation embedding is shown in Fig. 2.2, which accepts input in a triplet of news image, content and profile, and processes them through a deep network: (1) three different networks to deal with the triplet including news image, content and profile; (2) adversarial mechanism to enforce the same distribution between different modalities; and (3) a novel hybrid similarity loss to model the user's sentiment and incorporate it into the proposed framework.

We built the image network based on VGGNet [26], which is pre-trained on the ImageNet database [27]. To fit CNN into our SAME model, we reserve the first seven layers and replace the eighth layer by a layer with R nodes, $\mathbf{fch}^i$. As for the text network, we use GloVe [28] to process text $\mathbf{y}$, in order to capture the complex characteristics of word use (e.g., syntax and semantics). The obtained text representation is used as the input of the text network. We then adopt the Multi-Layer Perceptron (MLP) comprising two fully connected layers. The second layer

$\mathbf{fch}^t$ has R hidden units, which transforms the network activation to R-dimensional representation. As for profile information, the features are discrete values such as the topic. So we use the one-hot encoding to represent the profile $\mathbf{z}$, and feed it to a two-layer fully-connected MLP, and get the representation $\mathbf{fch}^p$. As for the adversarial networks, we built the discriminator networks by using a three-layer feed-forward neural network.

To integrate the three networks mentioned above, a fully connected layer with M hidden units, which takes the representations of three networks as input, is added on top of the architecture. We denote the multi-modal feature extractor for news i as $f^{(v)}(\mathbf{T}_i^{(v)}; \theta_a) \in \mathbb{R}^M$, which corresponds to the output of the hybrid deep network for multi-modal correlation embedding. Here, θ_a is the network parameter to be learned.

2.4.2 Adversarial Learning

With the above network, however, different modalities are usually distributed inconsistently, which is not beneficial if we use the concatenation for fake news detection. In order to bridge this modality gap, we introduce an adversarial learning mechanism. We use two discriminators for image and profile modalities to investigate their distributions. For the image (profile) discriminator, the inputs are image (profile) features and text features obtained from the feature extractor, and the output is a binary label, either "0" or "1". Specifically, we denote the modality label for the textual feature that has been generated from the text network as "1" and define the modality label for image (profile) semantic modality features generated from image network (profile network) as "0". We feed the outputs of image and text network into one discriminator and feed the outputs of profile and text networks into the other discriminator. The loss functions of the two discriminators can be defined as $\mathcal{L}_a^i$ and $\mathcal{L}_a^p$. The two discriminators act as the two adversaries while we are training the SAME.

The loss function $\mathcal{L}_a^i$ can be written as follows:

$$\min_{\theta_c} \mathcal{L}_a^i = \sum_{j=1}^{2\times N} ||D^{i,t}(\mathbf{fch}_j^*) - \mathbf{d}_j^*||_2^2, \tag{2.2}$$

where $\mathbf{fch}_j^*$ is semantic features obtained from image network or text network, the modality label is $\mathbf{d}_j^*$. Specifically we have $\mathbf{d}_j^i = 0$ denoting the modality label for image and $\mathbf{d}_j^t = 1$ denoting the label for text. The result of Eqn. (2.2) is that the discriminator acts as a binary classifier $D^{i,t}(\mathbf{fch}_j^*; \theta_c)$, classifying the input features into class "1" and class "0". Similarly, we have $\mathcal{L}_a^p$.

The above idea motivates a MinMax game between the feature extractor and the event discriminator. On one hand, the feature extractor tries to fool the modality that the discriminator tries to maximize the discrimination loss. On the other hand, the modality discriminator tries to discover the modality-specific information included in the feature representations to recognize the modality label. In the experiments (section 2.5.4), we demonstrate the effectiveness of adversarial learning in detecting fake news.

2.4.3 Modeling Sentiment Correlation

In order to make the learned joint embeddings maximally preserve the similarity information, we propose a novel hybrid similarity loss by considering such two issues: (1) entity triplets with lower similarity should be separated and have discriminative embeddings; (2) entity triplets should have similar embeddings only if they are similar in the original feature spaces.

To address the first issue, we propose the *Graph Affinity Metric* between news i and news j. The *Graph Affinity Metric* is defined as follows

Definition 1 *Let G_{ij} denotes the similarity of sentiment polarity distribution between the comments of news i and j. We can define the Graph Affinity Metric between two news as G_{ij}.*

Then, we define the *Local Similarity Metric* to ensure the local similarity in each news to ensure the second issue above.

Definition 2 *The Local Similarity Metric $L_{ij,m}(m = 1, 2, 3)$ of each modality involves the local similarity information. On modality $\boldsymbol{x}$, we have*

$$L_{ij,1}^{(v)} = \begin{cases} 1, & \textit{if } \boldsymbol{x}_i \in \boldsymbol{N}_k(\boldsymbol{x}_j) \textit{ or } \boldsymbol{x}_j \in \boldsymbol{N}_k(\boldsymbol{x}_i) \\ 0. & \textit{otherwise} \end{cases}$$

where $\boldsymbol{N}_k(\cdot)$ denotes the set of k-nearest neighbors. Similarly, we have $L_{ij,2}$ and $L_{ij,3}$ defined on modalities $\boldsymbol{y}$ and $\boldsymbol{z}$ respectively.

According to our empirical study, we set the number of nearest neighbors to 5 throughout this paper.

To maintain the similarity between entities and preserve the local structural information in the common embedding space, we propose a hybrid similarity loss loss which ensures the learned embedding space meaningful:

$$\min_{\theta_a} \mathcal{L}_c = \frac{1}{2} \sum_{i,j=1}^{N} S_{ij} \|\mathbf{h}_i - \mathbf{h}_j\|_2^2 \tag{2.3}$$

where $S_{ij} = G_{ij} + L_{ij,1} + L_{ij,2} + L_{ij,3}$.

2.4.4 Fake News Detector

In this section, we introduce how to detect fake news by using the M-dimensional embedding. We use a fully connected layer with softmax, which is shown in Fig. 2.2. Each network takes embedding vectors $\mathbf{h}_i$of news i as input.

We have a training set $\{r_i\}_{i=1}^{N}$, where $r_i \in \{0, 1\}$ denotes the ground truth label of news i. The goal is to find a set of prediction function g, such that the label for any news i can be predicted. We denote the fake news detector as $g^{(v)}(f^{(v)}(\mathbf{T}_i^{(v)}; \theta_a); \theta_b) \in \mathbb{R}$, where θ_b is the network parameter of the network for fake news detector.

Assume the ranking score is modeled as $\hat{r}_i = [\hat{r}_{i,0}, \hat{r}_{i,1}]$, with $\hat{r}_{i,0}$ and $\hat{r}_{i,1}$ indicate the predicted probability of label being 0 (real news) and 1 (fake news) respectively. r_i denotes the ground truth label of news. Thus, for each news, the goal is to minimize the cross-entropy loss function as follows:

$$\min_{\theta_a, \theta_b} \mathcal{L}_q = -r_i \log(\hat{r}_{i,1}) - (1 - r_i) \log(1 - \hat{r}_{i,0}) \tag{2.4}$$

2.4.5 The Proposed Method: SAME

During the training, the feature extractor and the fake news detector work together to minimize the detection loss $\mathcal{L}_q$. Simultaneously, the feature extractor tries to fool the discriminator to get a distribution agreement for different modalities by maximizing the adversarial loss $\mathcal{L}_a^i$ and $\mathcal{L}_a^p$. The

The final objective function of the proposed SAME is:

$$
\begin{aligned}
\mathcal{J}_g &= \mathcal{L}_c + \gamma \mathcal{L}_q \\
\mathcal{J}_a &= \mathcal{L}_a^i + \mathcal{L}_a^p
\end{aligned}
\tag{2.5}
$$

where γ is a penalty parameter for trading off the relative importance of multi-modal correlation and news label. We set $\gamma = 1$ based on empirical study.

If we put them together, we can obtain:

$$
\begin{aligned}
(\theta_a, \theta_b) &= arg \min_{\theta_a, \theta_b} \mathcal{J}_g(\theta_a, \theta_b) - \mathcal{J}_a(\hat{\theta}_c) \\
\theta_c &= arg \max_{\theta_c} \mathcal{J}_g(\hat{\theta}_a, \hat{\theta}_b) - \mathcal{J}_a(\theta_c)
\end{aligned}
\tag{2.6}
$$

All the parameters in the network are learned through RMSprop, which has been widely used among existing methods. It is an adaptive learning rate method which divides the learning rate by an exponentially decaying average of squared gradients.

2.5 Experimental Validation

In this section, we conduct experiments to demonstrate the effectiveness of the proposed method SAME. We first describe experimental settings. We then compare SAME against several state-of-the-art baselines for fake news detection followed by an ablation study to understand the contribution of each component of SAME. The experiments are conducted on two real-world datasets, PolitiFact and GossipCop, introduced in Section 2.3.1.

2.5.1 Compared Methods

We compare SAME with several representative and state-of-the-art fake news detection methods including KNN, SVM, TCNN-URG[3] [13], EANN[4] [6] and CSI[5] [13]. Our implementation of SAME is available here[6].

- **KNN**: This determines the authenticity of the news based on the labels of its neighbors, defined in **Definition 2**.

[3] We implemented the code by ourselves.

[4] https://github.com/yaqingwang/EANN-KDD18

[5] https://github.com/sungyongs/CSI-Code

[6] https://github.com/cuilimeng/SAME

- **SVM**: We concatenate the features including the outputs of VGGNet, GloVe and one-hot encoding, and sentiment polarity distribution vector as the input of Linear SVM. We choose Linear SVM as it is suitable for high-dimensional data.

- **TCNN-URG**: this method exploits the user's historical responses to related articles as soft semantic labels. TCNN generates the representation for each article, which is used for further news classification. URG is trained to learn the user's responses to news articles, which can help the classification procedure of TCNN when user's response is not available in early detection.

- **EANN**: In this method, both text and image information are taken into consideration. This method uses an event discriminator in order to eliminate the effects of the event-specific features and maintain the common features among all these studied events. We remove the event discriminator of this method as our datasets do not have event labels.

- **CSI**: This method explores all of the news content, users' responses to the news, and the sources that users promote in detecting fake news. However, as our datasets do not have time interval information in users' comments, we modify the codes accordingly.

For KNN, we set $k = 5$ based on empirical study. We use $C = 1$ in Linear SVM. As for other baseline methods, we use the parameter settings in the paper or in the released source code. For our method, we implemented it using Keras. The news image is re-sized to 128 × 128 pixels. The image network is pre-trained on the ImageNet classification task [27]. We fine-tune CONV1-FC7 initialized from the pre-trained model, and train layer FCH via back-propagation. Each news content is processed through GloVe. For the text network, we use a two-layer neural network, in which the first layer has 4,096 ReLU units with a dropout rate of 0.5. The news profile is represented by one-hot encoding, which is fed into a two-layer neural network as well. We fix the mini-batch size as 128, and set the learning rate as 0.001.

2.5.2 Evaluation Metrics

As the data is imbalanced, following the common way, we use Macro F1 and Micro F1 as evaluation metrics. Macro Precision is the average precision of all classes, similarly,

Table 2.3: Performance comparison on the two datasets. The best results are listed in bold.

		Training Ratio							
		20%		40%		60%		80%	
Datasets	Measure	Macro F1	Micro F1	Macro F1	Micro F1	Macro F1	Micro F1	Macro F1	Micro F1
PolitiFact	KNN	45.30	35.25	56.83	53.87	60.24	55.91	56.53	53.84
	SVM	53.50	51.42	60.74	56.83	64.37	59.39	65.57	60.56
	TCNN-URG	64.53	60.53	68.35	61.50	70.24	67.41	72.35	70.64
	EANN	63.53	59.42	67.93	63.88	70.22	65.65	71.31	69.38
	CSI	65.42	63.42	67.35	65.29	69.64	67.12	74.24	73.24
	SAME	**69.12**	**68.23**	**69.24**	**65.34**	**73.24**	**75.42**	**77.24**	**76.31**
GossipCop	KNN	59.24	56.24	55.46	53.54	54.31	59.32	57.20	53.37
	SVM	56.42	56.58	54.24	57.34	55.24	57.24	61.24	62.34
	TCNN-URG	66.22	62.42	65.33	62.24	67.42	63.42	73.24	68.43
	EANN	65.91	63.62	67.24	65.13	70.23	69.23	71.21	72.24
	CSI	72.35	71.53	74.24	72.24	76.42	74.82	77.24	76.87
	SAME	**76.24**	**76.42**	**78.24**	**75.61**	**77.24**	**78.31**	**80.42**	**81.58**

Table 2.4: Comparison of variants of SAME on two datasets. The best results are listed in bold.

		Training Ratio							
		20%		40%		60%		80%	
Datasets	Measure	Macro F1	Micro F1	Macro F1	Micro F1	Macro F1	Micro F1	Macro F1	Micro F1
PolitiFact	SAME w/o I	63.24	56.24	66.52	65.76	69.38	64.73	73.24	71.68
	SAME w/o D	65.74	63.24	65.13	64.31	68.13	67.91	74.86	72.61
	SAME w/o S	60.37	61.42	63.29	63.88	63.24	62.78	70.85	69.54
	SAME	**69.12**	**68.23**	**69.24**	**65.34**	**73.24**	**75.42**	**77.24**	**76.31**
GossipCop	SAME w/o I	71.53	69.53	73.24	72.24	74.24	72.72	75.24	73.23
	SAME w/o D	70.93	71.84	73.15	72.04	75.14	73.93	77.79	75.37
	SAME w/o S	65.67	64.82	67.71	67.93	73.39	71.01	75.91	73.37
	SAME	**76.24**	**76.42**	**78.24**	**75.61**	**77.24**	**78.31**	**80.42**	**81.58**

Macro Recall is the average recall of all classes. Macro F1 is the harmonic mean of Macro Precision and Macro Recall. Macro F1 calculates metrics for each label, and uses their unweighted mean. It does not take label imbalance into account. However, Micro F1 does not calculate on each class, it calculates metrics by counting the total true positives, false negatives and false positives globally.

2.5.3 Performance Comparison

We predict the score of the authenticity of news on two datasets respectively. We randomly select x% of data for training and the remaining (100-x)% for testing. To fully understand how SAME performs under different data size, we vary x as $\{20, 40, 60, 80\}$. The process is performed for 5 times and the average performance is reported in Table 2.3. From the experimental results, we make the following observations:

- For the SVM method, it concatenates all the features together. However, the results are far from satisfactory. We assume that the features used are highly nonlinear, simple concatenation may cause dense features to dominate the feature space and override the effects of the sparse ones.

- For other baseline methods, the information used is not comprehensive (including visual, textual, profile and sentimental features), so the effects are not as good as SAME.

- Compared against the best baseline, SAME achieves an absolute increase of 2.8%/3.0% on average in terms of Macro F1 and 4.0%/4.1% on average in terms of Micro F1 on two datasets. This clearly demonstrates that SAME is able to leverage heterogeneous data signals while integrating sentiments for effective fake news detection.

2.5.4 Ablation Study

In this section, we conduct an ablation study to fully understand the contribution of each component in SAME. We remove several critical modules in SAME that process images, news profile, and social sentiment (and their corresponding discriminator and loss function) as follows:

- SAME without image data (SAME w/o I): this method removes the images network.

- SAME without discriminators (SAME w/o D): this methods removes the two discriminators.

- SAME without users' sentiment information (SAME w/o S): social sentiment is removed from the proposed model.

- SAME: this method is the proposed method, which incorporates not only the three multi-modal networks, but also the sentiment information from users' comments.

We report the Macro F1 and Micro F1 on both datasets in Table 2.4. We can observe that all the components: visual and textual features, social context features and adversarial mechanisms are indispensable for achieving the best performance of SAME. Different components can provide complementary information, which also verifies the effectiveness of our proposed framework.

2.6 Conclusion and Future Work

In this paper, we investigate a novel problem of exploring sentiment for fake news detection with multi-modal data. We first use statistical analysis to test the hypothesis in order to validate the effectiveness of user's sentiment. Then, we propose a new deep multi-modal embedding architecture for fake news detection, which unifies multi-modal data with adversarial learning and incorporates user's sentiment. The experimental results demonstrate the effectiveness of our method as well as the roles of user's sentiment in fake news detection. In addition, we also examine the necessity of each module in the proposed method and thus test the fusion network proposed. The outcome of this work not only has a significant contribution to building a machine-based solution for detecting fake news, but also has a far-reaching impact on society by helping improve the quality of information.

There are several interesting directions that need further investigation. First, to mitigate the problem of fake news better, extending SAME to be able to do the early detection is important yet challenging (due to the lack of important signals). Second, most of the current fake news detection methods solely focus on the detection.

However, in addition to detecting fake news, being able to “explain" why one is fake news is equally important.

Chapter 3 Knowledge Graph based Misinformation Detection

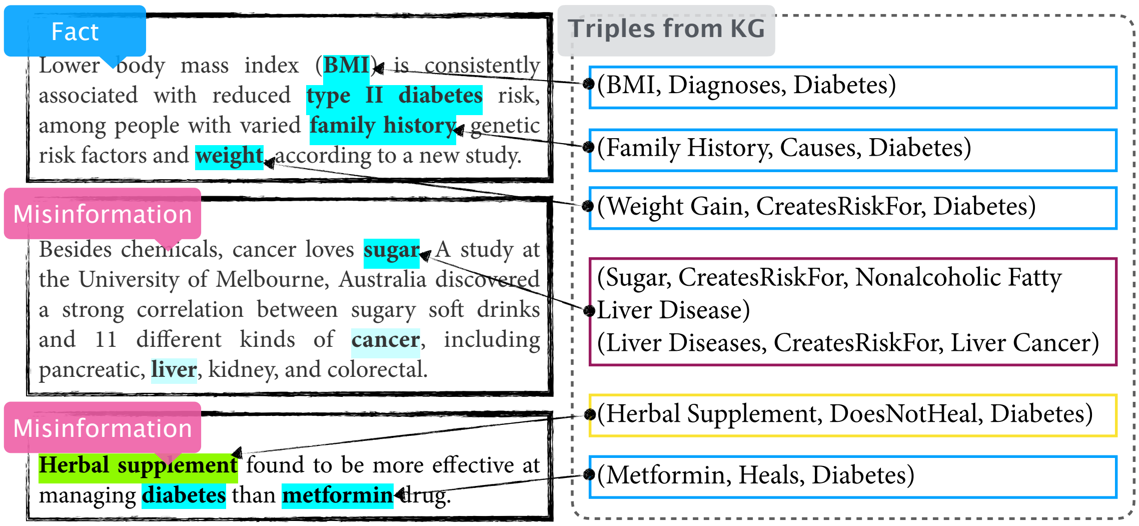
3.1 Introduction

The popularity of online social networks has promoted the growth of various applications and information, which also enables users to browse and publish such information more freely. In the healthcare domain, patients often browse the Internet looking for information about illnesses and symptoms. For example, nearly 65% of Internet users use the Internet to search for related topics in healthcare [29]. However, the quality of online healthcare information is questionable. Many studies [30,31] have confirmed the existence and the spread of healthcare misinformation. For example, a study of three health social networking websites found that 54% of posts contained medical claims that are inaccurate or incomplete [32].

Healthcare misinformation has detrimental societal effects. First, the community's trust and support for public health agencies are undermined by misinformation, which could hinder public health control. For example, the rapid spread of misinformation is undermining trust in vaccines crucial to public health[1]. Second, health rumors that circulate on social media could directly threaten public health. During the 2014 Ebola outbreak, the World Health Organization (WHO) noted that some misinformation on social media about certain products that could prevent or cure the Ebola virus disease has led to deaths[2]. Thus, detecting healthcare misinformation is critically important.

[1] `https://www.nature.com/articles/d41586-018-07034-4`

[2] `https://www.who.int/mediacentre/news/ebola/15-august-2014/en/`



Lower body mass index (BMI) is consistently associated with reduced type II diabetes risk, among people with varied family history, genetic risk factors and weight, according to a new study.

Besides chemicals, cancer loves sugar. A study at the University of Melbourne, Australia discovered a strong correlation between sugary soft drinks and 11 different kinds of cancer, including pancreatic, liver, kidney, and colorectal.

Herbal supplement found to be more effective at managing diabetes than metformin drug.

(BMI, Diagnoses, Diabetes)

(Family History, Causes, Diabetes)

(Weight Gain, CreatesRiskFor, Diabetes)

(Sugar, CreatesRiskFor, Nonalcoholic Fatty Liver Disease)
(Liver Diseases, CreatesRiskFor, Liver Cancer)

(Herbal Supplement, DoesNotHeal, Diabetes)

(Metformin, Heals, Diabetes)

Figure 3.1: Healthcare article examples and related triples from a medical knowledge graph(KG). The triples can either enhance or weaken the augments in the articles.

Though misinformation detection in other domains such as politics and gossips have been extensively studied [1, 3, 16], healthcare misinformation detection has its unique properties and challenges. *First*, as non-health professionals can easily rely on given health information, it is difficult for them to discern information correctly, especially when the misinformation was intentionally made to target such people. Existing misinformation detection for domains such as politics and gossips usually adopt social contexts such as user comments to provide auxiliary information for detection [7, 14, 33, 34]. However, in the healthcare domain, social context information is not always available and may not be useful because users without professional knowledge seldom respond to healthcare information and cannot give accurate comments. *Second*, despite the good performance of existing misinformation detection methods [6], the majority of them cannot explain why a piece of information is classified as misinformation. Without proper explanation, users who have no health expertise might not be able to accept the result of the detection. To convince them, it is necessary to offer an understandable explanation of why certain information is unreliable. Therefore, we need some auxiliary information that can (1) help detect healthcare misinformation; and (2) provide easy to understand professional knowledge for an explanation.

The medical knowledge graph, which is constructed from research papers and reports can be used as an effective auxiliary for healthcare misinformation detection, to find the inherent relations between entities in texts to improve detection performance and provide explanations. In particular, we take the article-entity bipartite graph and medical knowledge graph as complementary information, into consideration to facilitate a detection model (See Figure 3.1). First, article contents contain linguistic features that could be used to verify the truthfulness of an article. Misinformation (including hoaxes, rumors and fake news) is intentionally written to mislead readers by using exaggeration and sensationalization verbally. For example, we can infer from a medical knowledge graph that *Sugar* is not directly linked to *Liver Cancer*, however, the misinformation indicates that there is a “strong correlation” between the two entities. Second, the relation triples from a medical knowledge graph can add/remove the credibility of certain information, and provide explanations to the detection results. For example, in Figure 3.1, we can see that the triple (*BMI*, *Diagnoses*, *Diabetes*) and two more triples can directly verify that the article is real, while the triple (*Herbal Supplement*, *DoesNotHeal*, *Diabetes*) can prove that the saying in an article is wrong. Above all, it is beneficial to explore the medical graph for healthcare misinformation detection. And to our best knowledge, there is no prior attempt to detect healthcare misinformation by exploiting the knowledge graph.

Therefore in this paper, we study a novel problem of explainable healthcare misinformation detection by leveraging the medical knowledge graph. Modeling the medical knowledge graph with healthcare articles is a non-trivial task. On the one hand, healthcare information/texts and medical knowledge graph cannot be directly integrated, as they have different data structures. On the other hand, social network analysis techniques are not applied to the medical knowledge graph. For example, recommendation systems would recommend movies to users who watched a similar set of movies. However, in the healthcare domain, two medications are not necessarily related even if they can heal the same disease. To address the above two issues, we propose a knowledge guided graph attention network that can better capture the crucial entities in news articles and guide the article embedding. We incorporate the *Article-Entity Bipartite Graph* and a *Medical Knowledge Graph* into a unified relational graph and compute node embeddings along with the graph. We use the *Node-level Attention* and *BPR loss* [35] to tackle the positive and negative relations in the graph. The main contributions of the paper include:

- We study a novel problem of explainable healthcare misinformation detection by leveraging medical knowledge graph to better capture the high-order relations between entities;

- We propose a novel method DETERRENT (knowleDgE guided graph aTtention nEtwoRks foR hEalthcare misiNformation deTection), which characterizes multiple positive and negative relations in the medical knowledge graph under a relational graph attention network; and

- We manually build two healthcare misinformation datasets on diabetes and cancer. Extensive experiments have demonstrated the effectiveness of DETERRENT. The reported results show that DETERRENT achieves a relative improvement of 1.05%, 4.78% on the Diabetes dataset and 6.30%, 12.79% on the Cancer dataset comparing with the best results in terms of Accuracy and F1 Score. The case study shows the interpretability of DETERRENT.

3.2 Related Work

In this section, we briefly review two related topics: misinformation detection and graph neural networks.

Misinformation Detection. Misinformation detection methods generally focus on using article contents and external sources. Article contents contain linguistic clues and visual factors that can differentiate the fake and real information. Linguistic features based methods check the consistency between the headlines and contents [4], or capture specific writing styles and sensational headlines that commonly occur in fake content [36]. Visual-based features can work with linguistic features to identify fake images [6], and help to detect misinformation collectively [2, 33].

For external sources based approaches, the features are mainly context-based. Context-based features represent the information of users' engagements from online social media. Users' responses in terms of credibility [12], viewpoints [34] and emotional signals [2] are beneficial to detect misinformation. The diffusion network constructed from users' posts can evaluate the differences in the spread of truth and falsity [37]. However, users' engagements are not always available when a news article is just released, or users lack professional knowledge of relevant fields such as medicine. Knowledge graph (KG) can address the disadvantages of current methods relying on social context and derive explanations to the detection results. Some researchers use knowledge graph based methods to decide and explain whether a

(Subject, Predicate, Object) triple is fake or not [38–40]. These methods use the score function to measure the relevance of the vector embedding of the subject and vector embedding of the object with the embedding representation of predicate. For example, KG-Miner exploits frequent predicate paths between a pair of entities [41]. Other researchers use news streams to update the knowledge graph [42].

Hence in this paper, we study the novel problem of knowledge guided misinformation detection, aiming to improve misinformation detection performance in healthcare, and provide a possible interpretation of the result of detection simultaneously.

Graph Neural Networks. Graph Neural Networks (GNNs) refer to the neural network models that are applied to graph-structured data and aim to learn node embeddings by aggregating local neighborhood information. Several variants of GNN have been proposed to improve its representation capability and efficiency. GCN [43] tries to learn node embeddings in a semi-supervised fashion using per-neighbor normalization, instead of simply averaging all the neighborhood information. GAT [44] extends GNN by incorporating the attention mechanism; thus, each neighboring node can have a different level of contribution to the central node. R-GCN [45] is also an extension of GCN which is suitable for large-scale relational data. It is an entity encoder model that uses a new propagation model in the forward-pass update of entities to be able to handle relational data. RGAT [46] takes advantage of both the attention mechanism and R-GCN to build an efficient graph classification model suitable for relational input data. Signed Networks [47–50] are variants of GCNs applicable to the signed graph domain, in which each edge has a positive or negative sign. These methods benefit from the balance theory in social psychology to be able to correctly captures negative and positive links in the aggregating process and propagate information across layers.

However, existing methods are not suitable for modeling the positive and negative relations in the medical knowledge graph, as mentioned in the introduction. In this work, we model the medical knowledge graph under a relational graph attention network, and use BPR loss to capture positive and negative relations.

3.3 Problem Formulation

In this section, we describe the notations and formulate medical knowledge graph guided misinformation detection problem. The medical knowledge graph describes the entities collected from the medical literature, as well as positive/negative relations (e.g., $Heals/DoesNotHeal$) among entities. For example, (*Calcium Chloride*, $Heals$, *Hypocalcemia*) contains a positive relationship, but (*Actonel*, $DoesNotHeal$, *Hypocalcemia*) has a negative relationship.

Definition 3 ***Medical Knowledge Graph****: Let $\mathcal{G}_m = \{\mathcal{E}, \mathcal{R}, \mathcal{R}^-, \mathcal{T}, \mathcal{T}^-\}$ be a knowledge graph, where $\mathcal{E}$, $\mathcal{R}$, $\mathcal{R}^-$, $\mathcal{T}$ and $\mathcal{T}^-$ are the entity set, positive relation set, negative relation set, positive subject-relation-object triple set and negative triple set, respectively. The positive triples are presented as $\{(e_i, r, e_j)|e_i, e_j \in \mathcal{E}, r \in \mathcal{R}\}$, which describes a relationship r from the head node e_i to the tail node e_j. Similarly, negative triples are represented as $\{(e_i, r, e_j)|e_i, e_j \in \mathcal{E}, r \in \mathcal{R}^-\}$.*

We denote $\mathcal{D}$ as the health-related article set. Each article $S \in \mathcal{D}$ contains $|S|$ words, $S = \{w_1, w_2, \ldots, w_{|S|}\}$. We perform entity linking to build the word-entity alignment set $\{(w, e)|w \in \mathcal{V}, e \in \mathcal{E}\}$, where (w, e) means that word w in the vocabulary $\mathcal{V}$ can be linked with an entity e in the entity set. To capture the co-relationships of articles and entities in a medical knowledge graph, we define the article-entity bipartite graphs as follow.

Definition 4 ***Article-Entity Bipartite Graph****: The article-entity bipartite graph is denoted as $\mathcal{G}_{ae} = (\mathcal{D} \cup \mathcal{E}, \mathcal{L})$, where $\mathcal{L}$ is the set of links. The link is denoted as $\{(S, Has, e)|S \in \mathcal{D}, e \in \mathcal{E}\}$. If an article S contains a word that can be linked to entity e, there will be a link "Has" between them, otherwise none.*

Exploiting the knowledge path between entities is of great importance. Here we formally define the knowledge path.

Definition 5 ***Knowledge Path****: A knowledge path between entity e_1 and e_k is denoted as $e_1, r_1, e_2, r_2 \ldots, r_{k-1}, e_k$, where $e_k \in \mathcal{E}$, $r_k \in \mathcal{R}$ and $(e_{k-1}, r_{k-1}, e_k) \in \mathcal{T}$.*

Consider such a knowledge path: e_1, r_1, e_2, r_2, e_3, of which the two relations are $(Diabetes, CreatesRiskFor, Kidney\ Disease)$ and $(Kidney\ Disease, Causes, Edema)$. The two relations build a path between "diabetes" and "edema", which implies a potential link between the two disorders. Such a knowledge path can add credibility to the article mentioning these two disorders. Conversely, if two words are not reachable in a knowledge graph, such two words are largely irrelevant, which reduces the credibility of related articles. For example, although "bipolar disorder" and "fenofibrate" may be the causes of "diabetes", there is no strong connection between the two entities themselves from a medical perspective. However, existing text classification methods regard 'bipolar disorder" and "fenofibrate" as related as they both co-occur with "diabetes" a lot. Hence, we argue that considering knowledge paths between words through a knowledge graph can provide medical evidence in healthcare misinformation detection, which yields higher detection accuracy.

With the above notations and definitions, we formulate the knowledge guided misinformation detection task as follows:

Problem 1 (Medical Knowledge Graph Guided Misinformation Detection) *Given a set of healthcare articles $\mathcal{D}$, their corresponding label set $\mathcal{Y}$, and the medical knowledge graph $\mathcal{G}$, the goal is to learn a prediction function f for distinguishing whether an article is fake.*

3.4 Methodology

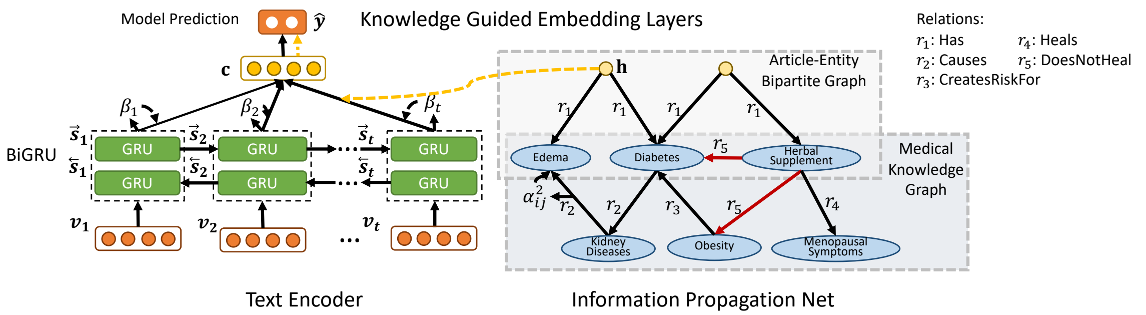


Figure 3.2: Illustration of the proposed DETERRENT model. The left subfigure shows the Knowledge Guided Embedding Layers of DETERRENT, and the right subfigure presents the Information Propagation Net of DETERRENT. The Information Propagation Net is performed on the unified graph of Article-Entity Bipartite Graph and Medical Knowledge Graph, which has positive (in black) and negative (in red) relations.

Our proposed framework consists of three components, which is shown in Figure 3.2: 1) an information propagation net, which propagates the knowledge between articles and nodes by preserving the structure of KG; 2) knowledge aware attention, which learns the weights of a node's neighbors in KG and aggregates the information from the neighbors and an article's contextual information to update its representation; 3) a prediction layer, which takes an article's representation as input and outputs a predicted label. Next, we introduce the details of each component.

3.4.1 Information Propagation Net

The medical knowledge graph can provide medical evidence in healthcare misinformation detection. To fully utilize the medical knowledge graph for healthcare misinformation detection, motivated by previous work [42, 45], we leverage the inherent directional structure of the medical database to learn the entity embedding. To propagate the information from the knowledge graph to the article, we incorporate the Article-Entity Bipartite Graph and Medical Knowledge Graph into a unified relational graph, and add a set of self-loops (edge type 0) denoted as $\mathcal{A} = \{(e_i, 0, e_i) | e_i \in \mathcal{E}\}$, which allows the state of a node to be kept. Hence, the new graph is defined as $\mathcal{G} = \{\mathcal{E}', \mathcal{R}', \mathcal{R}^-, \mathcal{T}', \mathcal{T}^-\}$, where $\mathcal{E}' = \mathcal{E} \cup \mathcal{D}$, $\mathcal{R}' = \mathcal{R} \cup \mathcal{R}^- \cup \{Has, 0\}$ and $\mathcal{T}' = \mathcal{T} \cup \mathcal{T}^- \cup \mathcal{L} \cup \mathcal{A}$.

Information Propagation: As there are multiple relations in a graph, we use R-GCN [45] to model the relational data, which is very effective in modeling multi-relational graph data. In R-GCN, each node is assigned to an initial representation $\mathbf{h}_i^{(0)}$. The layer-wise propagation rule updates the node representation using the representations of its neighbors in the graph in the $(l+1)$-th layer, yielding the representation $\mathbf{h}_i^{(l+1)}$ as follows:

$$\mathbf{h}_i^{(l+1)} = \sigma \left(\sum_{r \in \mathcal{R}'} \sum_{(j,r,i) \in \mathcal{T}'} \frac{1}{c_{i,r}} \mathbf{W}^r \mathbf{h}_j^{(l)} \right), \tag{3.1}$$

where $c_{i,r}$ is a normalization factor which is usually set to the number of neighbors of node $i \in \mathcal{E}'$ under relation $r \in \mathcal{R}'$, $\mathbf{W}^r$ is a learnable edge-type-dependent weight parameter and $\sigma(\cdot)$ denotes an activation function (we use LeakyReLU in this paper).

Node-level Attention: Each entity has relations with multiple entities. Not all relations are equally important for the healthcare misinformation detection problem. However, each neighbor has different importance to the node representation. Thus, we introduce the attention mechanism into the Information Propagation in Eq. (3.1) to assign more weights to important neighboring nodes, and the node representation is computed as the weighted sum of neighbors':

$$\mathbf{h}_i^{(l+1)} = \sigma\left(\sum_{r\in\mathcal{R}'}\sum_{(j,r,i)\in\mathcal{T}'}\alpha_{ij}^r\mathbf{W}^r\mathbf{h}_j^{(l)}\right) \tag{3.2}$$

where α_{ij}^r measures the importance of node i for a neighbor j, which is calculated as follows:

$$\begin{aligned}\mathbf{u}_{ij}^r &= \mathbf{W}^r(\mathbf{h}_i^{(l)} \parallel \mathbf{h}_j^{(l)}),\\ \alpha_{ij}^r &= \frac{\exp(\mathbf{a}^r\mathbf{u}_{ij}^r)}{\sum_{(k,r,i)\in\mathcal{T}'}\exp(\mathbf{a}^r\mathbf{u}_{ik}^r)}\end{aligned} \tag{3.3}$$

where $\mathbf{a}^r$ is the learnable parameter that weighs different feature dimensions of the node representation.

An issue of Eq. 3.2 is that, with the increasing number of relation types, the model will be quickly over-parameterized. To alleviate this problem, we apply Basis Decomposition [45] for regularization. This approach decomposes the weight matrix into a linear combination of several basic matrices, which largely decreases the number of model parameters.

Modeling Negative Relations: Since negative relations have different effects on the target node compared with positive relations, they should be treated separately. For example, the following three positive triples between four entities in a medical knowledge graph: 1) *Calcitriol* can heal *Calcium Deficiency*; 2) *Actonel* can heal *Calcium Deficiency*; and 3) *Calcitriol* can alleviate *Hypocalcemia*. Intuitively, we can infer that *Actonel* is a potential treatment for *Hypocalcemia*. However, a negative triple in a medical knowledge graph indicates that *Actonel* does not heal *Hypocalcemia*. Although the fact overrides our guess, it is explainable medically: Both *Calcitriol* and *Actonel* can treat *Calcium Deficiency*. However, the active ingredients in them are Vitamin D and Risedronate, respectively. Furthermore, Vitamin D in *Calcitriol* can alleviate *Hypocalcemia* while Risedronate cannot. Thus, when we are modeling the graph, we hope the discrepancy between two entities in a negative triple is larger than in a positive triple. To achieve this goal, we choose BPR loss [35]. It is commonly used in recommendation systems, to maximize the difference between the scores of the positive and negative samples. Hence, we first conduct inner product of entity representations as the matching score:

$$m_{ij} = (\mathbf{W}^r\mathbf{h}_j)^{\mathrm{T}}\tanh\left(\mathbf{W}^r\mathbf{h}_i\right) \tag{3.4}$$

where $\mathbf{h}_i$ and $\mathbf{h}_j$ are the representations for entity e_i and e_j under relation r in each layer. Then we use BPR loss to penalize the scores of two entities in a negative triple:

$$\mathcal{L}_k = \sum_{\substack{(e_j,r,e_i)\in\mathcal{T}' \\ (e_k,r,e_i)\in\mathcal{T}^-}} -\ln\sigma\left(m_{ij} - m_{ik}\right) \tag{3.5}$$

where $\sigma(\cdot)$ is the Sigmoid function.

It is worth noting that the signed GCNs [49, 50] use balance theory [51] in social psychology to deal with the negative relations in GCN. The balance theory suggests a positive relationship between two nodes, if there exists a knowledge path between the nodes that have an even number of negative relations (e.g., "The enemy of my enemy is my friend"). However, these methods cannot be used in modeling the medical knowledge graph due to the complexity of entities (medications and diagnoses). Distinct from the existing methods, our model uses a soft assumption on the negative relations, which does not require the graph to be balanced.

3.4.2 Knowledge Guided Embedding Layers

After going through the Information Propagation Net, we can get the neighboring attention weights of nodes (including articles). In this section, we propose Knowledge Guided Embedding Layers to use the relevance scores of entities to an article to guide the embedding of the article.

Text Encoder: To fully capture the contextual information of an article, we use BiGRU [52] to encode word sequences from both directions of words. To be specific, given the word embeddings $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{|S|}\}$ of an article S, the article embedding is computed as below:

$$\begin{aligned} \overrightarrow{\mathbf{s}}_t &= \mathrm{GRU}(\overrightarrow{\mathbf{s}}_{t-1}, \mathbf{v}_t) \\ \overleftarrow{\mathbf{s}}_t &= \mathrm{GRU}(\overleftarrow{\mathbf{s}}_{t-1}, \mathbf{v}_t) \end{aligned} \tag{3.6}$$

We concatenate the forward hidden state $\overrightarrow{\mathbf{s}}_t$ and the backward hidden state $\overleftarrow{\mathbf{s}}_t$ as $\mathbf{s}_t = [\overrightarrow{\mathbf{s}}_t, \overleftarrow{\mathbf{s}}_t]$, which captures the contextual information of the article centered around word $\mathbf{v}_t$.

Since not all words equally contribute to the semantic representation of the article, we leverage the attention mechanism to learn the weights to measure the importance of each word, and compute the article representation vector as follows:

$$\mathbf{c} = \sum_{t=1}^{|S|} \beta_t \mathbf{s}_t \tag{3.7}$$

where β_t measures the importance of the t-th word for the article, which is calculated as follows:

$$\begin{aligned} \mathbf{u}_t &= \tanh\left(\mathbf{W}_c \mathbf{s}_t + \mathbf{b}_c\right) \\ \beta_t &= \frac{\exp(\mathbf{u}_t^{\mathrm{T}} \mathbf{g})}{\sum_{k=1}^{|S|} \exp(\mathbf{u}_k^{\mathrm{T}} \mathbf{g})} \end{aligned} \tag{3.8}$$

where $\mathbf{u}_t$ is a hidden representation of $\mathbf{v}_t$ obtained by feeding the hidden state $\mathbf{v}_t$ to a fully embedding layer, and $\mathbf{g}$ is a trainable parameter to guide the extraction of the context.

Knowledge Guided Attention: To incorporate the knowledge guidance into the textual information, we update the $\mathbf{g}$ in Eq. 3.8 by $\mathbf{g}'$ to get the final attention function:

$$\mathbf{g}' = \gamma \mathbf{g} + (1 - \gamma) \mathbf{W}_k \mathbf{h}^s \tag{3.9}$$

where $\mathbf{h}^s$ is the node embedding of the article S obtained from the Information Propagation Net, $\mathbf{W}_k$ is a learnable transformation matrix and $\gamma \in [0, 1]$ is a trade-off parameter that controls the relative importance of the two terms. If we set $\gamma = 1$, then $\mathbf{g}'$ degenerates to $\mathbf{g}$ and our framework degenerates to a text classifier without the information from the medical knowledge graph. It makes it easy to pre-train the model to get good word embeddings for misinformation detection. The updated context vector $\mathbf{g}'$ takes both linguistic features from BiGRU and knowledge guidance into consideration. The Information Propagation Net propagates more information among similar entities and articles through the knowledge paths. We further use the attention score β_t to compute the articles representation vector $\mathbf{c}$ by Eq. 3.7.

3.4.3 Model Prediction

We have introduced how we can encode article contents through knowledge guidance. We further feed the embeddings to a softmax layer for misinformation classification as follows:

$$\hat{y} = \mathrm{Softmax}(\mathbf{W}_f \mathbf{c} + \mathbf{b}_f) \tag{3.10}$$

where $\hat{y}$ is the predicted value which indicates the probability of the article being fake. For each article, our goal is to minimize the cross-entropy loss:

$$\mathcal{L}_d = -y \log \hat{y} - (1-y) \log(1-\hat{y}) \tag{3.11}$$

where $y \in \{0, 1\}$ is the ground truth label being 0 (fact) and 1 (misinformation), respectively.

3.4.4 Training and Inference with DETERRENT

Finally, we combine the detection goal with BPR loss to form the final objective function as follows:

$$\mathcal{L}_{final} = \mathcal{L}_k + \mathcal{L}_d + \eta \left\|\Theta\right\|_2^2 \tag{3.12}$$

where Θ is the model parameters, and η is a regularization factor.

During the training, we optimize $\mathcal{L}_k$ and $\mathcal{L}_d$ alternatively. We use Adam [53] to optimize the embedding loss and the prediction loss. Adam is a widely used optimizer, which can compute individual adaptive learning rates for different parameters w.r.t. the absolute value of gradient.

3.5 Experiments

In this section, we present the experiments to evaluate the effectiveness of DETERRENT. Specifically, we aim to answer the following evaluation questions:

- **RQ1**: Is DETERRENT able to improve misinformation classification performance by incorporating the medical knowledge graph?
- **RQ2**: How effective are knowledge graph and knowledge aware attention, respectively, in improving the misinformation detection performance of DETERRENT?
- **RQ3**: Can DETERRENT provide reasonable explanations about misinformation detection results?

Next, we first introduce the datasets and baselines, followed by experiments to answer these questions.

Table 3.1: Statistics of datasets

Disease	Diabetes	Cancer
# Misinformation	608	1,476
# Fact	1,661	4,623
# Entities	1,932	2,873
# Relations	22,685	28,391

3.5.1 Datasets

As the medical knowledge graph, we use a public medical knowledge graph KnowLife[3] [54] with 25,334 entity names and 591,171 triples. We extract six positive relations including $Causes$, $Heals$, $CreatesRiskFor$, $ReducesRiskFor$, $Alleviates$, $Aggravates$ and four negative relations including $DoesNotCause$, $DoesNotHeal$, $DoesNotCreateRiskFor$, $DoesNotReduceRiskFor$.

To evaluate the performance of DETERRENT, we need a reasonably sized collection of health-related articles of several diseases with labels. Unfortunately, there is no available dataset of adequate size. For this reason, we have collected a health-related article dataset whose years range from 2014 to 2019.

To gather real articles, we crawled from 7 reliable media outlets that have been cross-checked as reliable, e.g., Healthline, ScienceDaily, NIH (National Institutes of Health), MNT (Medical News Today), Mayo Clinic, Cleveland Clinic, WebMD. For misinformation, we crawled verified health misinformation from Snopes.com and Hoaxy API, which are a popular hoax-debunking site and a web tool respectively. The detailed statistics of the datasets are shown in Table 3.1.

3.5.2 Baselines

We compare DETERRENTwith representative and state-of-the-art misinformation detection algorithms, which are listed as follows:

- KG-Miner [41]: KG-Miner is a fast discriminative path mining algorithm that can predict the truthfulness of a statement. We first use OpenIE [55] to extract the relation triple of each sentence in the article. Then we compute the score of

[3] http://knowlife.mpi-inf.mpg.de/

Table 3.2: Performance Comparison on Diabetes and Cancer datasets. DETERRENT outperforms all state-of-the-art baselines including knowledge graph based and article contents based methods.

Datasets	Metric	KG-Miner	TransE	text-CNN	CSI\c	dEFEND\c	GUpdater	HGAT	DETERRENT
Diabetes	Accuracy	0.7601	0.7671	0.7566	0.8359	0.9101	0.9012	0.8888	**0.9206**
	Precision	0.5398	0.5963	0.5563	0.6847	**0.9793**	0.9687	0.7730	0.8445
	Recall	0.6333	0.4248	0.4836	0.7826	0.6597	0.6369	0.8289	**0.8503**
	F1 Score	0.5828	0.4961	0.5174	0.7304	0.7883	0.7685	0.7996	**0.8474**
Cancer	Accuracy	0.8051	0.8536	0.8812	0.8982	0.8969	0.9022	0.8608	**0.9652**
	Precision	0.5790	0.6455	0.8531	0.7900	0.8847	0.7868	0.7226	**0.9469**
	Recall	0.7365	0.8125	0.5988	0.8165	0.6538	0.8147	0.7338	**0.9153**
	F1 Score	0.6485	0.7195	0.7037	0.8030	0.7519	0.8005	0.7282	**0.9309**

each triple when the subject, predicate, object are all in the KG, and average all the score as output label.

- TransE [56]: TranE is a knowledge graph embedding method, which embeds entities and relations into latent vectors and completes KGs based on these vectors. We use TransE on the unified relational graph. The article embeddings are used for misinformation detection.

- text-CNN [57]: text-CNN is a text classification model that utilizes convolutional neural networks to model article contents, which can capture different granularity of text features with multiple convolution filters.

- CSI\c [12]: CSI is a hybrid deep learning-based misinformation detection model that utilizes information from article content and user response. The article representation is modeled via an LSTM model with the article embedding via Doc2Vec [58] and user response. For a fair comparison, the user features are ignored. This abbreviated model is termed as CSI\c.

- dEFEND\c [59]: dEFEND utilizes a hierarchical attention neural network framework on article content and co-attention mechanism between article content and user comment for misinformation detection. For a fair comparison, the user comments are ignored. This abbreviated model is termed as dEFEND\c.

- HGAT [60]: HGAT is a flexible heterogeneous information network framework for classifying short texts, which can integrate any type of additional information. We add *Semantic Group* to the entities as side information, such as *Procedures* and *Disorders*.

- GUpdater [42]: GUpdater can update KGs by using the news. It is built upon GNNs with a text-based attention mechanism to guide the updating message passing through KG structures. Similar to TransE, we use article embeddings for misinformation detection.

Note that for a fair comparison, we choose above contrasting methods that use features from the following aspects: (1) only knowledge graph, such as TransE, KG-Miner; (2) only article contents, such as text-CNN, CSI\c, dEFEND\c and (3) both knowledge graph and article contents, such as HGAT and GUpdater. For knowledge graph methods, we feed output article embeddings into several traditional machine learning methods and choose the one that achieves the best performance. The methods include Logistic Regression, Multilayer Perceptron and Random Forest. We run these methods by using scikit-learn [61] with default parameter settings.

3.5.3 Experimental Setup

3.5.3.1 Metrics

To evaluate the performance of misinformation detection algorithms, we use the following metrics, which are commonly used to evaluate classifiers in related areas: Accuracy, Precision, Recall, and F1 score.

3.5.3.2 Implementation Details

We implement all models with Keras. We randomly use the labels of 75% news pieces for training and predict the remaining 25%. We set the hidden dimension of our model and other neural models to 128. The word embeddings are initialized by GloVe [28] and the dimension of pre-trained word embeddings is 100. For DETERRENT, the entity embeddings and relation embeddings are pre-trained using Information Propagation Net. We tested the depth of DETERRENT $L = \{1, 2, 3, 4\}$ and learning rate $lr = \{10^{-2}, 10^{-3}, 10^{-4}\}$. We tried $\gamma = \{0.01, 0.05, 0.1, 0.5\}$ and $\gamma = 0.05$ works best. We set $\eta = 0.05$. For other methods, we follow the network architectures as shown in the papers. For all models, we use Adam with a minibatch of 50 articles on the Diabetes dataset and 100 on the Cancer dataset, and the training epoch is set as 10. For a fair comparison, we use cross-entropy loss.

3.5.4 Misinformation Detection (RQ1)

To answer **RQ1**, we first compare DETERRENT with the representative misinformation detection algorithms introduced in Section 3.5.2, and then investigate the performance of DETERRENT when dealing with different types of articles.

3.5.4.1 Overall Comparison

Table 3.2 summarized the detection performance of all competing methods (reporting the average of 5 runs). From the table, we make the following observations:

- For knowledge graph-based methods, TransE and KG-Miner, the performance is less satisfactory. Although they are designed for KG triple checking and they do not incorporate linguistic features in news information. TransE can capture article-entity relations to differentiate fake and real news. When detecting fake articles, KG-Miner is dependent on OpenIE to extract the relation triple from the contents, and the performance of OpenIE tends to decrease as the sentence gets longer.

- In addition, article content-based methods, text-CNN, CSI\c and dEFEND\c perform better than those methods purely based on a knowledge graph. This indicates that the methods can utilize the semantic and syntactic clues in texts. dEFEND\c can better capture important words and sentences that can contribute to the prediction through a hierarchical attention structure.

- Moreover, methods using both article contents and knowledge graph, DETERRENT, GUpdater, and HGAT, perform comparable or better than those methods using either one of them, and those only based on the knowledge graph. This indicates that a knowledge graph can provide complementary information to the linguistic features, and thus improving the detection results thereby.

- Generally, for methods based on both article contents and knowledge graph, we can see that DETERRENT consistently outperforms other methods in terms of Accuracy and F1 Score on both two datasets. DETERRENT achieves a relative improvement of 1.05%, 4.78% on the Diabetes dataset and 6.30%, 12.79% on the Cancer dataset, comparing against the best results in terms of Accuracy and F1 Score.

- It is worthwhile to point out that dEFEND\c and CSI\c have a relatively high Precision and low Recall, which indicates that the methods predict positive samples (misinformation) wrongly as negative (fact). Hence we can see the necessity of modeling the relations between entities, as only linguistic information is not enough to distinguish fake and real information.

3.5.4.2 Performance Comparison w.r.t. Article Types

Besides fake articles, misinformation also includes shorter formats such as clickbait and fake posts which can easily be posted and quickly go viral on social media. The important motivation of misinformation detection is to build a general framework to detect various types of misinformation.

Hence we investigate the performance of DETERRENT when dealing with different types of articles, including title and abstract. We evaluate DETERRENT by using articles' titles and abstracts respectively. The results in terms of the F1 score on both datasets are shown in Figure 3.3. The bars show the word lengths of different news types in log base 10. From the results, we observe that:

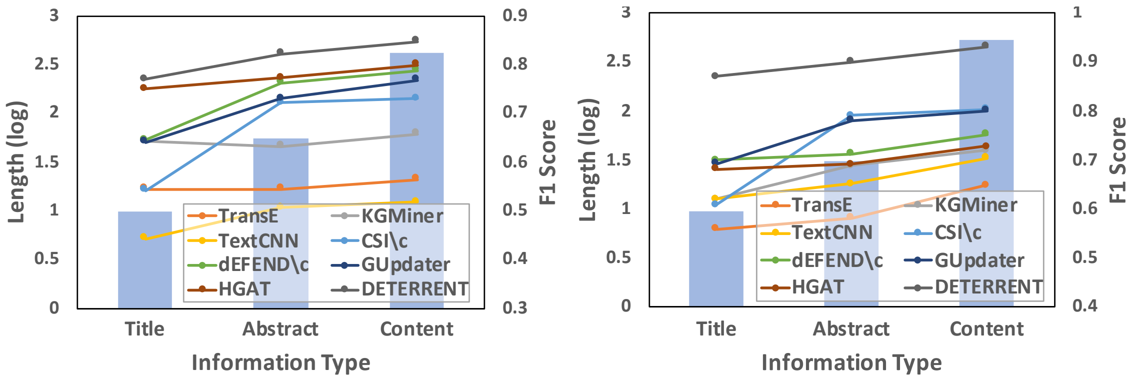
- DETERRENT consistently outperforms the other models. It demonstrates the effectiveness of DETERRENT on different types of misinformation regardless of the length. It again verifies the significance of knowledge graph and knowledge guided text embedding.
- The performance of article contents based methods like CSI\c and dEFEND\c do not perform very well when the length of the information is short. This suggests that those methods rely on the linguistic features of contents and cannot avoid the disadvantages brought by limited data. Although DETERRENT leverages article contents, it also exploits the additional information of entities to address the above issue. The performance of DETERRENT only slightly decreases when dealing with titles (the shortest text).
- The performance of knowledge graph-based methods, KG-Miner and TransE, is relatively stable with all types of information on the two datasets.

3.5.5 Ablation Analysis (RQ2)

In order to answer **RQ2**, we explore each component of DETERRENT. We first investigate the layer number of the model, then we examine the components of knowledge graph embedding and the attention mechanisms by deriving several variants.

3.5.5.1 Effects of Network Depth

We vary the depth L of DETERRENT to investigate the efficiency of the usage of multiple embedding propagation layers of a knowledge graph. The larger L allows further information to propagate through the information propagation layer. In particular, we search the layer number in the set of $\{1, 2, 3, 4\}$. For $L > 3$, we did not



(a) Diabetes

(b) Cancer

Figure 3.3: Performance comparison over the length of article types on two datasets. The background histograms indicate the length of each article; meanwhile, the lines demonstrate the performance w.r.t. F1 score.

get satisfying results on both datasets, which suggests that forth- and higher-order knowledge paths contribute little information. The results are summarized in Table 3.3. From this, we make the following observations:

- Increasing the depth of DETERRENT can improve the performance of DETERRENT, which demonstrates the effectiveness of modeling high-order knowledge paths.

- By analyzing Table 3.2 and Table 3.3, we can see that DETERRENT is slightly better than the article contents based methods, which indicates the effectiveness of leveraging the relations between entities.

- Besides first-order knowledge paths, high-order knowledge paths can discover inherent relations that are overlooked by traditional methods.

3.5.5.2 Effects of Attention Mechanisms and Negative Relations

In addition to article contents, we also apply knowledge graph information and integrate it with article contents with knowledge guided attention. We further investigate the effects of these components by defining three variants of DETERRENT:

- w/o Rel: w/o Rel is a variant of DETERRENT, which does not consider the relations in the medical knowledge graph. The Information Propagation Net is replaced by a GNN model.

Table 3.3: Effects of the network depth

Datasets	Metric	1	2	3
Diabetes	Accuracy	0.8853	0.9171	0.9206
	Precision	0.7500	0.9217	0.8445
	Recall	0.8543	0.7361	0.8503
	F1 Score	0.7987	0.8185	0.8474
Cancer	Accuracy	0.9580	0.9599	0.9652
	Precision	0.9108	0.9507	0.9469
	Recall	0.9157	0.8817	0.9153
	F1 Score	0.9132	0.9149	0.9309

Table 3.4: Ablation study of DETERRENT demonstrated the advantage of the attention mechanisms and modeling both positive and negative relations.

Datasets	Metric	w/o Rel	w/o K-Att	w/o Neg
Diabetes	Accuracy	0.8412	0.9012	0.9118
	Precision	0.7164	0.8870	0.9565
	Recall	0.7988	0.7236	0.7096
	F1 Score	0.7554	0.7971	0.8148
Cancer	Accuracy	0.9022	0.9291	0.9586
	Precision	0.9291	0.9385	0.9462
	Recall	0.6569	0.7651	0.8756
	F1 Score	0.7697	0.8430	0.9096

- w/o K-Att: w/o K-Att is a variant of DETERRENT, which excludes the knowledge-guided attention module. Each article is represented by the concatenation of the text embedding from the text encoder and node embedding from the Information Propagation Net, and fed into the prediction module.

- w/o Neg: w/o Neg is a variant of DETERRENT, which does not specifically model the negative relations in the medical knowledge graph. The BPR loss is excluded from this variant.

When one removes a medical knowledge graph, leaving only a BiGRU text encoder, the results are far from satisfactory, and thus are omitted. We summarize the experimental results in Table 3.4 and have the following findings:

- When we solely use a medical knowledge graph without considering relations, the performance of DETERRENT largely degrades, which suggests the necessity of modeling relations.

- Removing knowledge guided embedding attention degrades the model's performance, as the attention mechanism will assign importance weights for words, based on the semantic clues in differentiating misinformation from fact without considering knowledge paths.

- When we do not specifically model negative relations, some entities may be embedded close in a relation wrongly through information propagation. Thus, some misinformation (label 1) may be predicted as fact (label 0), which leads to relatively high Precision and low Recall.

Through the ablation study of DETERRENT, we conclude that (1) knowledge-guided article embedding can contribute to the misinformation detection performance; (2) both positive and negative relations are necessary for effective misinformation detection.

3.5.6 Case Study (RQ3)

In order to illustrate the importance of knowledge graph for explaining healthcare misinformation detection results, we use an example to show the triples captured by DETERRENT in Figure 3.4 and the corresponding attention weight in Figure 3.5.

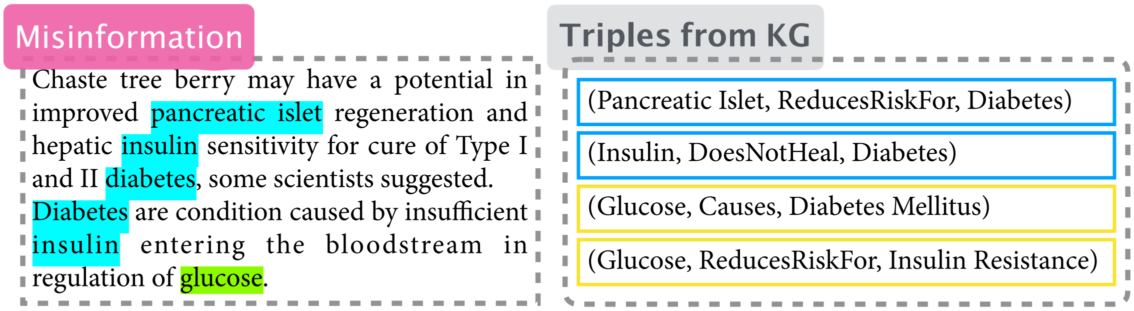

Misinformation
Chaste tree berry may have a potential in improved pancreatic islet regeneration and hepatic insulin sensitivity for cure of Type I and II diabetes, some scientists suggested.
Diabetes are condition caused by insufficient insulin entering the bloodstream in regulation of glucose.
Triples from KG
(Pancreatic Islet, ReducesRiskFor, Diabetes)
(Insulin, DoesNotHeal, Diabetes)
(Glucose, Causes, Diabetes Mellitus)
(Glucose, ReducesRiskFor, Insulin Resistance)

Figure 3.4: The explainable triples captured by DETERRENT.

In Figure5, $Diabetes$ has higher attention weights to the texts. The related triples ($PancreaticIslet$, $ReducesRiskFor$, $Diabetes$) and ($Insulin$, $DoesNotHeal$, $Diabetes$) can provide explanations about why the information is false, as the texts

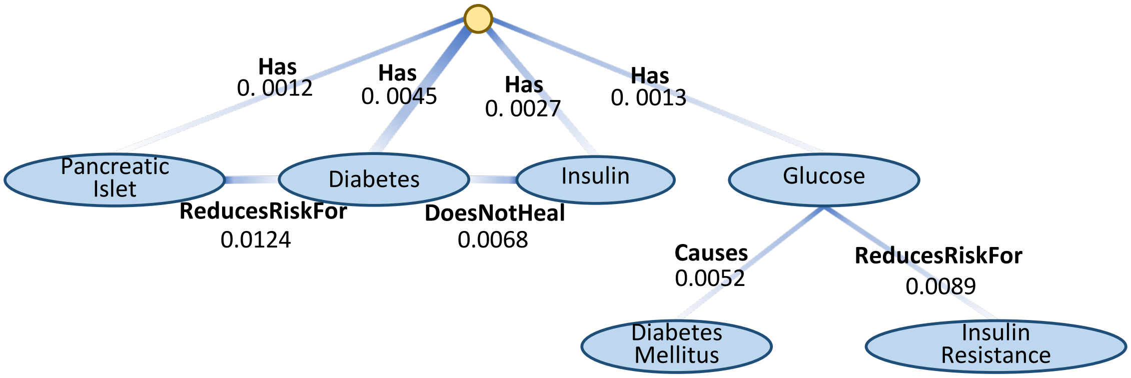
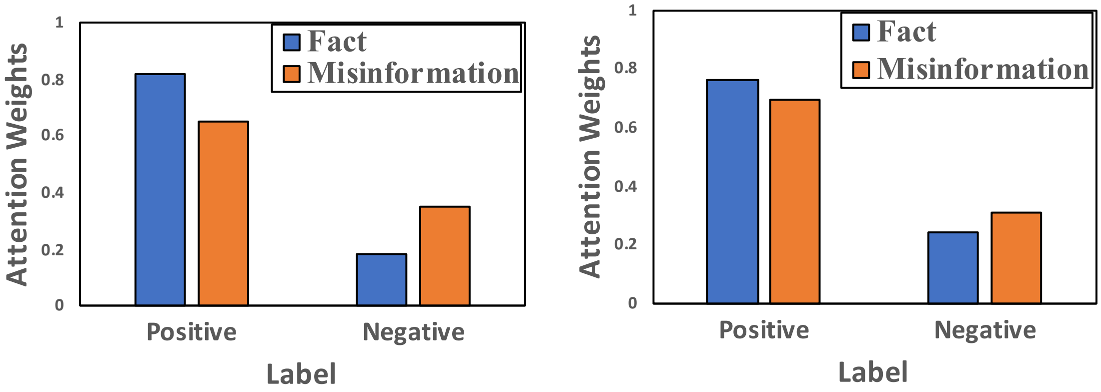


Figure 3.5: The visualization with attention wights.



(a) Diabetes

(b) Cancer

Figure 3.6: The attention weight analysis indicates that positive relations contribute more to fact, and negative relations contribute more to misinformation.

exaggerated the effects of *PancreaticIslet* and *Insulin*. In contrast, *Glucose* has a smaller attention weight than the above two entities. We can see that DETERRENT can not only detect the given information as fake but also yields the explanations of the detection results.

We calculate the average attention weights of positive and negative relations to both misinformation and fact on two datasets. The results are shown in Figure 3.6. Note that positive relations have higher attention weights to fact than misinformation. On the contrary, negative relations have higher attention weights to misinformation than fact. Hence, the attention weight analysis indicates that positive relations contribute more to the fact, and negative relations contribute more to misinformation.

3.6 Conclusion

In this paper, we proposed DETERRENT, a knowledge guided graph attention network for misinformation detection in healthcare. DETERRENT leverages additional information from a medical knowledge graph, to guide the article embedding with a graph attention network. The network can capture both positive and negative relations, and automatically assign more weights to important relations in differentiating misinformation from fact. The node embedding is used for guiding text encoder. Experiments on two real-world datasets demonstrate the strong performance of DETERRENT.

DETERRENT has two limitations: DETERRENT mainly targets on checking the truthfulness of an article be leveraging a knowledge graph, instead of other complementary information. In addition, DETERRENT does not consider the publishing time of an article. In the future, first, we can incorporate the data from medical forums to automatically find questionable user comments. Second, other data sources, such as doctors' remarks/comments can be considered for complementary information. Third, time intervals between posts can be considered for modeling the diffusion of healthcare information.

Chapter 4 Fraud Detection with Limited Data

4.1 Introduction

Human labeling is time-consuming and costly. This problem is further exacerbated in extremely imbalanced class label scenarios, such as detecting fraudsters in online websites. **Active Learning** (AL) selects the most relevant example for human labelers to improve the model performance at a lower cost. However, existing methods for active learning for graph data often assumes that both data and label distributions are balanced. These assumptions fail in extreme rare-class classification scenarios, such as classifying abusive reviews in an e-commerce website.

Graph structured data are ubiquitous and are widely used in social network analysis [62], financial fraud detection [63], molecular design [64], search engines [65] and recommender systems [66, 67]. Recently, Graph Neural Networks (GNNs) have emerged as state-of-the-art models on these types of datasets, due to their ability to learn and aggregate complex interactions between (K-hop) neighborhoods, as opposed to traditional pointwise or pairwise models [68]. Despite their appealing advantages, GNNs, like other deep learning models, require a large amount of labeled data for training in supervised settings. It is often time-consuming, labor-intensive, and expensive to acquire sufficient labeled data for training in many domains, hindering the application of GNNs.

Active Learning is a promising solution to obtain labels faster, cheaper, and train models efficiently. AL dynamically queries candidate samples[1] for labeling to maximize the performance of the machine learned model with limited budget. The recent developments in AL on graphs [64, 69–76] have proven to be effective on several benchmark datasets, such as citation graphs and gene networks. However, AL methods for large-scale *imbalanced* scenarios (e.g., finding a small fraction of fraudulent reviews on an e-commerce website) is less explored. This motivates us to study how to query the most "informative" samples so as to ameliorate the effect of imbalance and to reduce the training cost of GNNs.

Training GNNs with AL algorithm on imbalanced graphs is non-trivial. The **low prevalence rate** of positive samples[2] prevents traditional AL methods from learning the whole data distribution, because under-represented positive samples are less likely to be selected by traditional AL methods. For example, finding abusive reviews on a shopping website can be formulated as a binary classification problem, where positive samples (i.e., abusive reviews) are a very small portion of the labeled data. Training an AL model to sample reviews for labeling will mostly yield non-abusive reviews, resulting in limited model performance improvement. Most of the AL sampling methods proposed in natural language processing and computer vision [77–79] to balance class distribution assume independent and identically distributed (i.i.d.) data. These approaches are not directly applicable to graph structured data due to the heterogeneous relational structure and dense connections. Moreover, existing AL methods tend to reinforce or even worsen the prediction bias on minority classes when querying unlabeled data [79].

It is challenging to build an AL approach for **large-scale graph data**. For example, popular social network platforms (e.g., Facebook, Snapchat) have hundreds of millions of monthly active users; online e-commerce websites (e.g., Amazon, Walmart) host millions of products and conduct billions of transactions. Searching over all the unlabeled samples in the graph at this scale is impractical, as the computational complexity of AL methods grows exponentially with the size of the unlabeled set. Therefore, it is critical to reduce the search space for AL algorithms on large-scale graphs.

To tackle the aforementioned two challenges, we propose an Active Learning based method for Large-scale ImbalancEd graphs (ALLIE), which combines the idea of AL

[1] Samples in the case of node classification in a GNN will be nodes.

[2] We assume positive samples are the rare class in the imbalanced setting

on graphs with reinforcement learning for accurate and efficient node classification. ALLIE can effectively select informative unlabeled samples for labeling, using multiple uncertainty measures as its criteria. Moreover, our approach gives labeling priority to less confident and "under-represented" samples. To scale our approach to large graphs, we further introduce a graph coarsening strategy for ALLIE that categorizes similar nodes into clusters. With a better representation of nodes in each cluster, the search space for the AL algorithm is reduced. To the best of our knowledge, this work is the first to jointly model the imbalance issue on large-scale graphs and active learning. Our contributions are as follows:

- **Imbalance-aware reinforcement learning based graph policy network**. We apply a reinforcement learning strategy by maximizing the performance of the classifier to find a representative subset of the unlabeled dataset. The queried nodes will be more representative for the minority class (Section. 4.3.2.1).

- **Graph coarsening strategy to handle large-scale graph data**. Existing methods seldom pay attention to scalability, making them less efficient when applied to real-world applications. To reduce running time, we apply a graph coarsening strategy to reduce the action space in the policy network (Section. 4.3.2.2).

- **Robust learning for more accurate node classification**. Unlike conventional methods that do not distinguish the majority and minority classes when optimizing the objective function, we construct a node classifier with focal loss that down-weights the well-classified examples (Section. 4.3.2.3).

We evaluated ALLIE on both balanced and imbalanced datasets. Our balanced datasets use public citation graphs (Section. 4.4.2) and the imbalanced dataset is from a proprietary e-commerce website (Section. 4.4.3). We report the performance on node classification on both datasets. The reported results show that on balanced graph datasets, ALLIE improved an average of 2.39% in Macro F1 and 2.71% in Micro F1 over the best baseline. On the e-commerce website dataset, ALLIE achieved an average increase of 4.75% in Precision, 1.96% in Recall and 3.45% in F1 (with 10.54%, 3.7% and 7.71% relative improvement respectively) on the positive classes (i.e., the abusive users and reviews) over the best baseline. We also conduct a comprehensive ablation study to demonstrate the necessity of each component of ALLIE. Additional experiments show ALLIE performs well over baselines with various initial training set sizes and query budgets.

4.2 Related Work

4.2.1 Active Learning on Graphs

Active learning [80,81] has been widely studied in different domains such as computer vision [82,83] and natural language processing [84,85]. More recently, some pioneering works explore AL for graph structured data [64,75,76]. For example, AGE [69] selects the clustering center of node features. It combines several measurements together, including information entropy [86], density and centrality to find the best candidate(s) from all unlabeled nodes. FeatProp [70] extends AGE and also uses cluster centers as selected candidates. The authors proved an upper bound on the classification loss, and discussed why they chose K-Medoids instead of K-Center as the clustering method. $\mathrm{ANR}_{\mathrm{MAB}}$ [71] uses Multi-Armed Bandit to select one metric from the measurements in AGE. Chen et al. [72] propose ActiveHNE to further extend $\mathrm{ANR}_{\mathrm{MAB}}$ to cover heterogeneous graphs. GPA [73] uses a policy network to perform AL on graphs. The goal is to select a sequence of nodes by using reinforcement learning which maximizes the performance of the GNN. Different from the heuristics-based AL methods, MetAL [74] uses meta-gradients to evaluate the importance of labeling each unlabeled instance. Despite these achievements, existing work mainly focus on balanced datasets, and perform poorly when the datasets are imbalanced. In addition, the measurements that are commonly used to estimate the representativeness of the samples are centrality and density. Though these criteria can help characterize data distribution, they do not favor the most "underrepresented" samples at the borders between classes. We explicitly focus on functions that can be adapted to imbalanced datasets.

4.2.2 Graph Coarsening

Learning on graphs is too time-consuming for large-scale graph data that model the dense connections among millions of nodes. The computational cost grows exponentially with the number of nodes [87]. In addition to using state-of-the-art models, we can compress graphs to reduce the running time of AL on them. Sparsification and reduction are two common ways of simplifying graphs. Sparsification reduces the number of edges, such as spanners, edge cut, and spectral sparsifiers [88, 89]. Such methods have been previously used for recommendation systems [90]. Reduction is conducted on the number of vertices as well as the number of edges. Related methods include graph coarsening [91, 92] and Kron reduction [93].

Graph coarsening is the merging of vertices in a graph to obtain a coarser version of the original graph with similar spectral properties [94]. We can use the same algorithm to process the coarser graph as with the original. Graph coarsening can be repeated several times until we get a sufficiently coarse graph [95, 96]. DiffPool [91] uses an assignment matrix to transform the original graph to a coarser one. It pools nodes given an assignment matrix at each layer. SAGPool [92] generalizes convolution operations to graphs. Researchers have incorporated graph coarsening into GNNs as a way to implement efficient pooling [97, 98]. For example, GraphSAGE [99] with DiffPool is 12 times faster than the original model.

4.2.3 Imbalanced Learning

Many real-world applications in computer vision [100], medical diagnosis [101] and fraud detection [102] suffer from class imbalance. Learning from an imbalanced dataset may result in a prediction model that favors the majority class over the minority class [103]. A comprehensive review of class imbalance problems in deep learning can be found in [104].

Methods for dealing with imbalance can be roughly divided into two categories: data-level and algorithmic-level methods. Oversampling [105, 106] and undersampling [107] are two data-level methods that are commonly used in deep learning. Oversampling replicates selected samples from minority classes, while undersampling removes samples from majority classes. Algorithm-level methods keep the data unchanged while adjusting the training or inference process. Focal loss is a scaled cross-entropy loss, where the scaling factor goes to zero for well-classified samples [108]. Cost-sensitive learning assigns different costs to the misclassified samples from different classes [109].

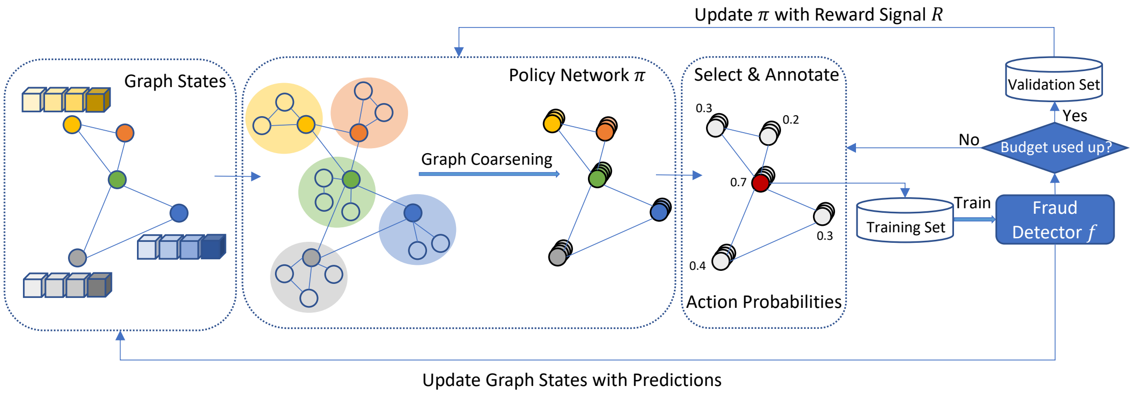


Figure 4.1: The overview of ALLIE with two main parts: policy network and fraud detector.

4.3 The Proposed Method: ALLIE

4.3.1 Task Description

Let $G = (V, E)$ denote a graph, where V is a set of nodes and E is a set of edges. We consider a classification setting where each node $v \in V$ has a label $y \in \mathcal{Y} = \{1, \dots, C\}$ (C is the number of classes). The node set is divided into three subsets including V_{train}, V_{valid} and V_{test}, with corresponding label sets Y_{train}, Y_{valid} and Y_{test}. In traditional supervised learning, the goal is to learn a classifier $f(G, V_{\text{train}}; \theta_d)$ parameterized by θ_d with the graph G and labels V_{train} to predict the labels of the nodes in V_{test}.

In AL setting, a query budget B is given, which allows us to query the labels of B samples from V_{train} ($B \ll |V_{\text{train}}|$) in total. Suppose the initial label set is denoted as V^0_{query}. At each step t, we select an unlabeled node v^t using an AL policy π from the remaining candidate nodes $V_{\text{train}} \backslash V^{t-1}_{\text{query}}$ that have not been queried, and query the label of the node v^t. Then we update V^t_{query} by $V^{t-1}_{\text{query}} \cup \{v^t\}$ and train the classifier $f(G, V^t_{\text{query}}; \theta_d)$ for one epoch. After the query budget is used up, we continue training $f(G, V^B_{\text{query}}; \theta_d)$ with V_{query} until convergence.

The learning process of policy π can be naturally formulated as a Markov Decision Process (MDP), in which the AL network is sequentially querying unlabeled nodes into a sequence over time. Formally, the MDP is defined as follows.

- **State space $\mathcal{S}$:** A state matrix $S^t \in \mathcal{S}$ is defined as the state of graph G at time t where each row $\mathbf{s}^t$ is the state representation of a node. More specifically, a state $\mathbf{s}^t$ consists of a node's degree, entropy, average KL divergence and reverse KL divergence between its predicted label distribution and its neighbor's.

- **Action space $\mathcal{A}$**: At time t, the action $a^t \in \mathcal{A}$ is to determine which node should be queried next. The AL network will append the node to the node sequence. The number of actions taken should satisfy the given budget constraint.

- **Reward $\mathcal{R}$**: After the network has selected a sequence of nodes, we evaluate the performance of the classifier on the validation set V_{valid} as the final reward. Since the size of the initial training set $|V_{\text{train}|}$ is limited in AL, calculating the immediate reward after each action will change the policy estimation a lot. Hence, in order to measure the policy's quality more accurately, we choose to calculate only the final reward, which provides a more stable estimation.

- **Transition probability $\mathcal{P}$**: Transition probability $p(S^{t+1}|S^t, a^t)$ defines the state transition from S^t to S^{t+1} after taking action a^t at time t.

We parameterize the policy network using a deep neural network, which is defined as follows:

Definition 6 (Policy Network) *A policy network $\pi(\cdot;\theta_p)$ parameterized by θ_p is used to select a node sequence from the candidate training nodes to query, which yields a probability score for each node in the unlabeled set. We learn the optimal parameter θ_p^* by maximizing the performance of the classifier f on the validation set: $\mathcal{M}(f(G, V_{valid}), Y_{valid})$, where $\mathcal{M}$ is an evaluation metric, f is trained on V_{query}^B chosen by θ_p^* and Y_{valid} is the labels of the validation set.*

Hence we can formally define the MDP based AL problem on graphs as follows:

Problem 2 (MDP based Active Learning Problem) *Given a graph $G = (V, E)$, with a query budget B, our goal is to learn a policy π to select the best node sequence to query, in order to optimize the prediction performance throughout the query process.*

4.3.2 Framework

Figure 4.1 illustrates the proposed framework. First, we alternately use a policy network to query the label of a candidate node and train the GNN classifier to update the current state of the graph, until the query budget is reached. In what follows, we evaluate the GNN classifier on the validation set to update the policy.

4.3.2.1 Reinforcement Learning Architecture

The AL algorithm takes an action by selecting the next node to query. In addition to the heuristic metrics, we can choose the nodes that can maximize the performance of the GNN classifier on the validation set. As this problem can be naturally formalized as a reinforcement learning architecture, we followed the GPA framework [70] in our paper.

We denote the state of graph G at step t as a matrix S^t, where each row $\mathbf{s}_v^t$ is the state representation of node v. In order to represent the state representation, we adopt degree as representativeness measure and entropy and KL divergence as the uncertainty measures in the policy network:

- Degree: We use the degree of a node to represent its representativeness. The higher the degree of the nodes, the more important the nodes are. Thus their labels are more likely to be informative. The degree is denoted by

  $$\mathbf{s}_{v,1}^t = \min(\text{degree}(v)/\delta, 1), \tag{4.1}$$

  where δ is a scaling hyperparameter.

- Entropy: The entropy of the label distribution is to predict the uncertainty of each node. In other words, if the classifier has low confidence about a node's predicted label, then the node's label is more likely to be useful. We divide the entropy by $\log(C)$ to normalize it into range $[0, 1]$:

  $$\mathbf{s}_{v,2}^t = -\frac{1}{\log(C)} \sum_{i=1}^{C} \hat{y}_i(v^t) \log(\hat{y}_i(v^t)), \tag{4.2}$$

  where $\hat{y}_i(v^t)$ is the class probability of node v belonging to the i-th class predicted by the classifier at step t.

- Divergence: The divergence is calculated based on a node's label prediction distribution and its neighbor's. It measures how different the node and its neighbors are, which can better identify the decision boundaries in the graph:

  $$\mathbf{s}_{v,3}^t = \frac{1}{|N_v|} \sum_{u \in N_v} \text{KL}(\hat{y}(v^t) \| \hat{y}(u^t)), \ \mathbf{s}_{v,4}^t = \frac{1}{|N_v|} \sum_{u \in N_v} \text{KL}(\hat{y}(u^t) \| \hat{y}(v^t)). \tag{4.3}$$

We use an indicator to represent whether the node has been labeled or not, and concatenate it with the above metrics to form the feature vector s_v^t for each node v. The graph state matrix S^t will be passed into the policy network to generate the action probabilities.

Imbalance-aware Reward Function Design: In order to fix the imbalanced data distribution issue and boost the model's performance on the minority classes, we introduce a balancing strategy on the reward in AL to make the method query more nodes that can represent the minority class better. Specifically, for the reward signal, we use a performance metric that treats each class equally instead of each sample and thus assigns more weights to the minority samples.

Below we detail how we calculate the reward signal and what metrics we choose. The policy network is rewarded by the performance gain of the GNN classifier f trained with the updated set of labeled nodes. The reward of the selected node sequence is calculated based on the performance of f on the validation set:

$$R(V_{\text{query}}^B) = \mathcal{M}(f(G, V_{\text{valid}}), Y_{\text{valid}}), \tag{4.4}$$

where $\mathcal{M}$ is the evaluation metric, f trained on graph G and labels of V_{query}^B, V_{valid} and Y_{valid} are the nodes and labels of the validation set.

We implement $\mathcal{M}$ using the following metrics:

- Weighted reward: When the sample belongs to the minority class, the reward is $+1$ if the prediction $\hat{y}$ is correct; -1 if not. When the sample belongs to the majority class, the reward is multiplied by the imbalanced ratio ρ, which is the number of samples in the minority class divided by the number of samples in the majority class [110].

- Micro-1 calculates metrics by counting the total true positives, false negatives and false positives globally, which favors the majority classes, e.g., the benign buyer.

- Macro-F1 averages the F1 score per class, which can get a sense of effectiveness on the small classes (e.g., the abusive buyer).

We empirically compare the three reward functions in Section 4.4.4.1 and find ALLIE with Macro-F1 achieves the best results.

Reinforcement Learning Algorithm: The training framework is shown in Figure 4.1. At every step, we first update the graph state matrix S_G^t. The policy network selects a node v^t from $V_{\text{train}} \backslash V_{\text{query}}^{t-1}$ based on the probability of each action $\pi(\cdot|S^t)$, gets its label, and puts it into the label set V_{query}^t. Then the GNN classifier f is trained for one epoch on graph state matrix S_G^t and the label set V_{query}^t. After that, we can get the new label prediction of each node and update the heuristic metrics such as $\mathbf{s}_{v,2}^t$, $\mathbf{s}_{v,3}^t$ and $\mathbf{s}_{v,4}^t$. The heuristic metrics are used to generate the graph state matrix S^{t+1} for the next step. When the query budget B is used up, we train the GNN classifier f until convergence without querying more nodes.

4.3.2.2 Policy Network Design

The policy network takes graph state as an input and produces the probability distribution of each action (where an action is querying a node's label). GNNs can better characterize the graph's topology and help find the most informative nodes in the graph. Hence we set up the policy network architecture as a GNN. We use GCN [68] to implement the policy network. In GCN, the nodes are assigned to an initial feature matrix $H^{(0)} \in \mathbb{R}^{N \times F}$, where N is the number of nodes and F is the feature dimension size. Here we use the initial state of graph $H^{(0)} = S^t$ as the initial input feature. The layer-wise propagation rule updates the node representations using the representations of its neighbors in the graph in the $(l+1)$-th layer, yielding the feature matrix:

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} W), \tag{4.5}$$

where $\tilde{A} \in \mathbb{R}^{N \times N}$ is the adjacency matrix with self-connections $(A + I)$, $\tilde{D}$ is the degree matrix of $\tilde{A}$, $W \in \mathbb{R}^{N \times F}$ is the weight matrix and $\sigma(\cdot)$ denotes an activation function (we use ReLU in this paper). We apply a linear layer to map the final output to a probability score indicating whether this node should be queried:

$$\pi(\cdot|S^t) = \text{Softmax}(WH^{(l)} + b). \tag{4.6}$$

Graph Coarsening: The computational cost will grow exponentially as the number of GCN layers increases. In addition, as the search space covers all the candidate nodes for annotation, the large number of discrete actions makes reinforcement learning methods difficult to apply for large graphs. Thus, we introduce the graph coarsening strategy SAGPool [92] into the GCN policy network in Eq. (4.5) to distinguish between the nodes that should be dropped and the nodes that should be retained, which will reduce the running time and shrink the action space at the same time.

The self-attention score matrix $Z \in \mathbb{R}^{N \times 1}$ is calculated as follows:

$$Z^{(l)} = \sigma(\tilde{D}^{-\frac{1}{2}} \tilde{A} \tilde{D}^{-\frac{1}{2}} H^{(l)} \Theta), \tag{4.7}$$

where $\Theta \in \mathbb{R}^{N \times 1}$ is the parameter matrix to be learned. With the attention score matrix Z, we can select the top k percent nodes to keep in each layer, yielding a list of top $\lceil kN \rceil$ nodes' indices:

$$idx = \text{top}(Z, \lceil kN \rceil). \tag{4.8}$$

The output feature matrix and the corresponding adjacency matrix of each layer are calculated as:

$$H^{(l+1)} = H^{(l)}_{idx,:} \odot Z^{(l)}_{idx}, \ A = A_{idx,idx}, \tag{4.9}$$

where $_{idx,:}$ represents the row-wise (i.e. node-wise) index notation, $\odot$ is the broadcasted elementwise product, and $_{idx,idx}$ represents the row-wise and col-wise index notation.

4.3.2.3 Robust Classification

The GCN fraud detector can classify both labeled and unlabeled nodes. On the top of the policy network, we apply a linear layer, taking the final output embedding $H^{(L)}$ as input. The output of the fraud detector is the probability of a node being positive.

The goal of the fraud detector is to determine whether a node is positive (abusive buyer) or not (benign buyer). On the shopping website, the positive class makes up only a very small portion (<5%). This data imbalanced issue causes two learning problems: (1) the easy negative samples do not contain much information to facilitate the training; (2) the easy negative samples may degenerate the model. To efficiently train on all samples, we employ focal loss [108]. Denote positive nodes as $v^+ \sim p_{\mathbb{R}^+}(v)$, and negative nodes $v^- \sim p_{\mathbb{R}^-}(v)$, where $\mathbb{R}^+$ and $\mathbb{R}^-$ represent the positive samples' and negative samples' spaces respectively. The loss function is denoted as follows:

$$\begin{aligned}\mathcal{J}_c(\theta_c) = &- \mathbb{E}_{v^+ \sim p_{\mathbb{R}^+}}[\alpha(1 - f(v;\theta_c))^\gamma \log f(v;\theta_c)] \\ &- \mathbb{E}_{v^- \sim p_{\mathbb{R}^-}}[(1-\alpha) f(v;\theta_c)^\gamma \log(1 - f(v;\theta_c))]\end{aligned}$$

where γ is a focusing parameter, which focuses more on hard and easily misclassified examples, and α is the weight assigned to the rare class. $\gamma = 2$ and $\alpha = 0.25$ work best based on the rule of thumb [108].

As for the multi-class scenario in general, we exclude α as it is not applicable for multiple classes. We still set γ as 2 based on rule of thumb. The multi-class focal loss is calculated as follows:

$$\mathcal{J}_c(\theta_c) = -(1 - f(v;\theta_c))^\gamma \log f(v;\theta_c) \tag{4.10}$$

4.3.2.4 Training and Inference

For training the classifier, we minimize the focal loss $\mathcal{J}_c$ in Eq. (4.10). The objective function of the policy network is:

$$\mathcal{J}_p(\theta_p) = \mathbb{E}_{\pi(V^B_{\text{query}};\theta_p)}[R(V^B_{\text{query}})], \tag{4.11}$$

where B is the query budget, and R is the reward on graph G. We use a classic policy gradient method REINFORCE [111] to train the policy network π.

In order to train the policy network $\pi(\cdot;\theta_p)$ parametered by θ_p, we alternately update θ_c by optimizing the focal loss $\mathcal{J}_c$ on the training data queried by policy $\pi(\cdot;\hat{\theta}_p)$, and update θ_p by maximizing the sum of expected rewards obtained from the classifier $f(\cdot;\hat{\theta}_c)$ on the validation set:

$$\theta_p^* = \operatorname*{argmax}_{\theta_p} \mathcal{J}_p(\theta_p), \quad \theta_c^* = \operatorname*{argmin}_{\theta_c} \mathcal{J}_c(\theta_c). \tag{4.12}$$

The training process is divided into two stages. In the first stage, we train the classifier $f(\cdot;\theta_c)$ to minimize the loss function $\mathcal{J}_c(\theta_c)$, while actively querying the unlabeled nodes. When the query budget is used up, we train the classifier f until convergence. In the second stage, we evaluate the trained classifier f on the validation set to get the reward signal and use that to update θ_p) together with the policy gradient. The detailed training steps are summarized in Algorithm 1.

Algorithm 1: ALLIE for AL on Graphs.

```
Input: Graph $G$, validation set $V_{\text{valid}}$ and corresponding label set $Y_{\text{valid}}$,
       initial query set $V^0_{\text{query}}$, query budget $B$ and training epochs $N$
Output: Well-trained node classifier $f$ and AL policy $\pi$
for $e = 1, \dots, N$ do
    for $t = 1, \dots, B$ do
        Update the graph state $S^t_G$;
        Use policy $\pi$ to sample a node based on $S^t_G$ for annotation, and add it
          to the query set $V^t_{\text{query}}$;
        Minimize the detection loss $\mathcal{J}_c(\theta_c)$ in Eq. (4.10) with the updated
          $V^t_{\text{query}}$ for one epoch;
    end
    while not converged do
        Minimize the detection loss $\mathcal{J}_c(\theta_c)$ in Eq. (4.10) with $V^B_{\text{query}}$;
    end
    Evaluate classifier $f$ on the validation set $V_{\text{valid}}$ and $Y_{\text{valid}}$ to get the
      reward signal $R(V^B_{\text{query}})$ in Eq. (4.4);
    Use the sum of expected rewards to learn the optimal policy $\pi^*$ in
      Eq. (4.11);
end
```

4.4 Experiments

In this section, we compare the performance of ALLIE with state-of-the-art AL methods on graphs. We aim to answer the following evaluation questions (EQ):

- **EQ1**: Is ALLIE able to improve the node classification performance on both benchmark dataset and real-world e-commerce dataset?
- **EQ2**: How effective are graph coarsening, focal loss, and reward function adaptation method in ALLIE?
- **EQ3**: How robust is ALLIE with respect to its hyperparameter values?

To this end, we introduce the datasets used and baselines, followed by experiments to answer these questions.

Table 4.1: Statistics of citation graph datasets.

	Cora	Citeseer	Pubmed
# nodes	2,485	2,110	19,717
# edges	5,068	3,668	44,338
# classes	7	6	3

Table 4.2: Summary of the e-commerce dataset. The dataset is heavily subsampled, and is used to show the efficacy of ALLIE on a real world use case.

Data property	Value
Node types	{buyer, seller, review, product}
# nodes (post sampling)	∼50K
% abusive buyers	5.2
# edges (post sampling)	∼61K
% abusive reviews	1.7

4.4.1 Experimental Setting

4.4.1.1 Datasets

We use several benchmark citation graph datasets (Cora, Citeseer, Pubmed [112]). The statistics of the citation graph datasets are presented in Table 4.1. We also use datasets created from sampled, anonymized logs from an e-commerce website. We construct a graph consisting of sellers, buyers, reviews and products. Table 4.2 shows the approximate numbers of the nodes and edges that we sampled. This dataset is heavily sampled, and is not reflective of production traffic. We merely use it here to highlight the utility of ALLIE. We randomly initialized the attributes of each node when training the graph neural network. Sampling is done by randomly picking 10K buyers, performing Breadth First Search (BFS), and add the additional nodes to our dataset. A similar sampling method is used previously in [113].

4.4.1.2 Implementation details

We implement ALLIE with PyTorch. We vary the learning rate in $\{10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}$ and found learning rate 10^{-2} worked best. For baselines, we follow the exact network architecture detailed in the corresponding works. We outline the baselines used in Section 4.4.1.3. For all models, we use Adam [114] with 100 epochs. On citation graph datasets, we use 5 samples from each class to construct the initial training set, and set the query budget as 20 for each class. On the e-commerce dataset, we use 200 samples from each class to construct the initial training set, and set query budget as 250 for each class. We repeat all experiments five times and report the averages and the standard deviations of the metrics measured.

4.4.1.3 Baselines

We compare ALLIE with the following representative and state-of-the-art AL on graph algorithms:

- Random: Random selects several candidate nodes uniformly at random to annotate in each epoch, and uses GNN to re-train the classifier using these nodes.
- AGE[3] [69]: AGE uses the weighted sum of entropy, density and centrality to find the best candidate(s) from all unlabeled nodes.
- FeatProp[4] [70]: FeatProp uses cluster centers as selected candidates through k-medoids clustering.
- GPA[5] [73]: GPA formalizes AL on graphs as an MDP (Markov Decision Process) and learns the optimal query strategy with reinforcement learning.
- MetAL[6] [74]: MetAL uses an AL algorithm that selects a set of unlabeled instances based on an informative metric, gets their labels, and updates the labeled dataset.

We choose the above methods that based on the following aspects: (1) only heuristic metrics, such as AGE and FeatProp; (2) heuristic metrics and reinforcement learning, such as GPA; and (3) heuristic metrics and meta-learning, such as MetAL. This allows us to compare ALLIE to multiple kinds of methods.

[3] https://github.com/vwz/AGE

[4] https://github.com/CrickWu/active_graph

[5] https://github.com/ShengdingHu/GraphPolicyNetworkActiveLearning

[6] https://github.com/Kaushalya/metal

4.4.1.4 Evaluation measures

We use Micro F1 and Macro F1 to evaluate the performance of all methods on the citation datasets. We report per-class precision, recall and F1 score on the e-commerce dataset. The latter dataset lends itself to a highly imbalanced classification problem.

4.4.2 EQ1: Performance on Public Datasets

To answer **EQ1**, we first compare ALLIE with the state-of-the-art AL algorithms introduced in Section 4.4.1.3 on benchmark graph datasets. We conduct experiments in both balanced and imbalanced settings.

4.4.2.1 Balanced Setting

We use the original datasets as-is to conduct the experiments in this setting. The problem is that of multi-class node classification. Table 4.3 summarizes the node classification performance of all competing methods (reporting the average of 5 runs). From the table, we can make the following observations:

- For the metric-based methods AGE and FeatProp, the performance is unsatisfactory. Though they use several heuristic metrics to capture the representativeness of nodes, they do not leverage node interactions to better measure node informativeness.

- The meta-learning based method MetAL performs better than metric-based methods, demonstrating the effectiveness of using the classifier's performance as feedback. MetAL is inferior to ALLIE. We hypothesize that MetAL needs a moderate- to large-sized initial training set to learn accurate model weights.

- ALLIE outperforms other methods in terms of Macro F1 and Micro F1 on three datasets. This shows that ALLIE effectively leverages both graph information as well as feedback.

4.4.2.2 Imbalanced Setting

In this setting, we manually adapt the datasets into binary classes to make the data distribution imbalanced. Following [115], we treat the smallest class in Cora, Citeseer and PubMed as the positive class and the rest as the negative class. The positive class

Table 4.3: Node classification performance (Macro F1 and Micro F1 ±Std) on the balanced setting of citation graph datasets. The best and 2nd best are noted in bold font and underlined, respectively.

	Cora		Citeseer		PubMed	
	Macro F1	Micro F1	Macro F1	Micro F1	Macro F1	Micro F1
Random	0.6819±0.041	0.7031±0.071	0.5438±0.093	0.5762±0.079	0.7033±0.051	0.7296±0.118
AGE	0.7322±0.046	0.7725±0.031	0.6152±0.013	0.6722±0.098	0.7735±0.019	0.7737±0.019
FeatProp	0.7826±0.018	0.7645±0.009	0.6417±0.041	0.6097±0.087	0.7392±0.068	0.7321±0.124
GPA	0.7677±0.063	0.8105±0.012	0.6635±0.039	0.7130±0.102	0.7912±0.091	0.7996±0.043
MetAL	0.7455±0.012	0.7985±0.023	0.6184±0.016	0.7018±0.065	0.7711±0.086	0.7764±0.075
ALLIE	**0.8025±0.071**	**0.8242±0.027**	**0.6838±0.020**	**0.7425±0.082**	**0.8228±0.038**	**0.8376±0.049**

Table 4.4: Node classification performance (Macro F1 and Micro F1 ±Std) on the imbalanced setting of citation graph datasets.

	Cora		Citeseer		PubMed	
	Macro F1	Micro F1	Macro F1	Micro F1	Macro F1	Micro F1
Random	0.1781±0.048	0.5645±0.042	0.2755±0.063	0.5695±0.072	0.1735±0.036	0.5691±0.193
AGE	0.1631±0.084	0.6424±0.038	0.3716±0.093	0.6293±0.049	0.1871±0.048	0.6266±0.051
FeatProp	0.3582±0.071	0.6427±0.193	0.4295±0.104	0.6817±0.094	0.2594+0.088	0.6692±0.038
GPA	0.4892±0.042	0.7384±0.098	0.4239±0.078	0.7084±0.053	0.3813±0.085	0.7442±0.158
MetAL	0.4583±0.035	0.6979±0.025	0.4328±0.062	0.7035±0.067	0.3602±0.105	**0.7791±0.084**
ALLIE	**0.5391±0.027**	**0.7692±0.015**	**0.4894±0.041**	**0.7684±0.074**	**0.4391±0.037**	0.7694±0.056

Table 4.5: Buyer classification performance relative change (Precision, Recall and F1 ±Std) on the e-commerce dataset.

	Benign Buyer			Abusive Buyer		
	Precision	Recall	F1	Precision	Recall	F1
Random	-	-	-	-	-	-
AGE	+0.1650±0.013	+0.0640±0.053	+0.1408±0.028	+0.1119±0.025	+0.0337±0.024	+0.0685±0.017
FeatProp	+0.1547±0.042	+0.0522±0.041	+0.0876±0.031	+0.0987±0.016	+0.0028±0.036	+0.0388±0.028
GPA	+0.1872±0.042	**+0.0950±0.032**	+0.1647±0.035	+0.1519±0.013	+0.0435±0.042	+0.0891±0.042
MetAL	+0.0546±0.031	+0.0401±0.018	+0.0743±0.039	+0.0941±0.031	+0.0482±0.013	+0.0771±0.031
ALLIE	**+0.1936±0.025**	+0.0834±0.018	**+0.1671±0.024**	**+0.2024±0.013**	**+0.0521±0.041**	**+0.1184±0.027**

Table 4.6: Review classification performance relative change (Precision, Recall and F1 ±Std) on the e-commerce dataset.

		Benign Review			Abusive Review	
	Precision	Recall	F1	Precision	Recall	F1
Random	-	-	-	-	-	-
AGE	+0.1864±0.013	+0.1968±0.019	+0.2037±0.173	+0.1544±0.012	+0.1299±0.695	+0.1187±0.031
FeatProp	+0.1432±0.022	+0.0612±0.017	+0.1286±0.022	+0.1858±0.023	+0.1259±0.031	+0.1409±0.032
GPA	**+0.3934±0.032**	+0.3167±0.009	+0.3885±0.025	+0.4515±0.013	+0.3815±0.058	+0.3979±0.023
MetAL	+0.1168±0.017	+0.1968±0.041	+0.2054±0.026	+0.4139±0.052	+0.2758±0.032	+0.3800±0.077
ALLIE	+0.3864±0.013	**+0.3968±0.019**	**+0.4207±0.173**	**+0.4960±0.056**	**+0.4409±0.013**	**+0.4375±0.042**

ratios are 7%, 8% and 21% respectively. Results are shown in Table 4.4 (reporting the average of 5 runs). From the table, we can see that:

- All the model performances on Macro-F1 and Micro-F1 degrade. This reinforces our hypothesis that when the data distribution becomes imbalanced, the classifier tends to predict most samples as belonging to the majority class.
- ALLIE outperforms the other models. It demonstrates the effectiveness of the balancing strategies, including the imbalance-aware reinforcement learning framework and focal loss.

4.4.3 EQ1: Performance on e-commerce dataset

In order to test **EQ1** in a real-world setting with large-scale imbalanced graphs, we define two node classification tasks on the e-commerce dataset. The tasks are detecting abusive users behavior and abusive reviews. Both these tasks are important in e-commerce to ensure high customer trust. Because the dataset is proprietary, we report relative changes in each metric with respect to a baseline.

4.4.3.1 Classification on Buyers

Here we investigate the performance of ALLIE when distinguishing abusive buyers from benign buyers, and make the following observations.

- ALLIE outperforms the other models. It again shows the importance of applying reinforcement learning to query nodes from the unlabeled data, which directly optimizes the performance of the GNN classifier.
- It is worthwhile to point out that ALLIE has a higher performance improvement with the abusive buyer class compared with the benign buyer class. This indicates the effectiveness of adapting the reward function to better capture the minority class (abusive buyer) and using focal loss to down-weight the well-classified samples (benign buyers far from the classification boundary).

4.4.3.2 Classification on Reviews

We summarize the observations of ALLIE in classifying abusive reviews and benign reviews.

Table 4.7: Effect of reward functions (Macro F1 and Micro F1 ±Std) on imbalanced citation graph datasets.

	Cora		Citeseer		PubMed	
Reward Function	Macro F1	Micro F1	Macro F1	Micro F1	Macro F1	Micro F1
Weighted Reward	0.5274±0.034	0.7524±0.080	0.4789±0.016	0.7639±0.011	0.4323±0.141	0.7543±0.187
Micro-F1	0.5245±0.138	0.7534±0.046	0.4716±0.022	0.7563±0.048	0.4287±0.069	0.7685±0.098
Macro-F1	**0.5391±0.027**	**0.7692±0.015**	**0.4894±0.041**	**0.7684±0.074**	**0.4391±0.037**	**0.7694±0.056**

- ALLIE still outperforms baselines, especially in the abusive review class, which indicates that ALLIE is suitable for imbalanced graphs.
- The scores of metric-based methods, AGE and FeatProp, on review classification task is generally much lower than their results on the buyer classification task. This indicates that their performance worsens when the data is more imbalanced.

4.4.4 EQ2: Ablation Study

In order to answer **EQ2**, we explore each component of ALLIE separately. We first study the influence of different reward function designs. Then we examine the influence of graph coarsening and balancing strategies.

4.4.4.1 Effect of reward functions

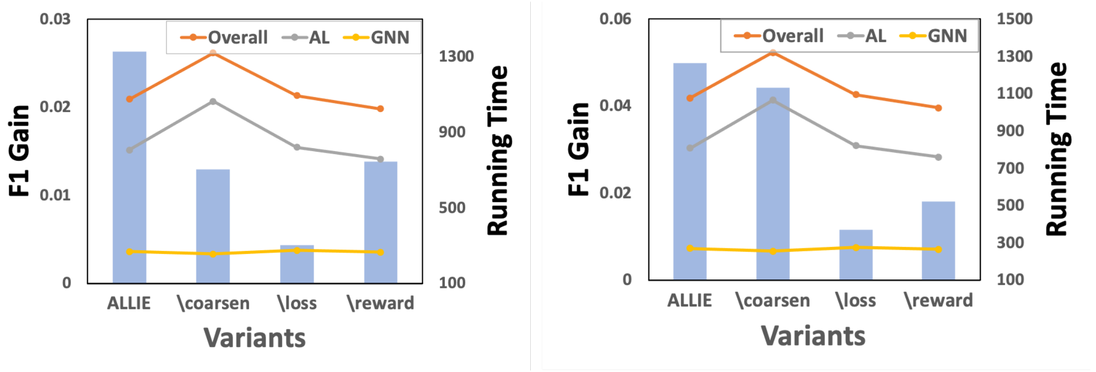
To explore the impact of various reward function designs in Section 4.3.2.1, we consider several variants of ALLIE that use different reward functions: weighted reward, Micro-F1 and Macro-F1. We term these methods $\text{ALLIE}_{\text{weighted reward}}$, $\text{ALLIE}_{\text{Micro-F1}}$ and $\text{ALLIE}_{\text{Macro-F1}}$ respectively.

Table 4.7 summarizes the results on the imbalanced setting of the public datasets in Section 4.4.2.2. We find that $\text{ALLIE}_{\text{Macro-F1}}$ is superior than $\text{ALLIE}_{\text{weighted reward}}$ and $\text{ALLIE}_{\text{Micro-F1}}$. This verifies that incorporating sample balancing into the reward function design can address the class imbalance issue.

4.4.4.2 Effect of graph coarsening and balancing strategies

We define several variants of ALLIE to study the effects of graph coarsening, focal loss and reward function adaptation:

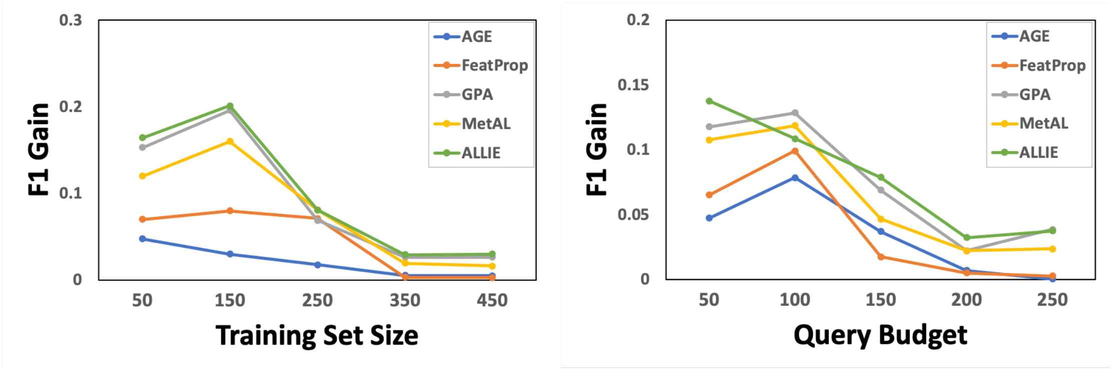
- **\coarsen**: This is a variant of ALLIE which does not integrate the graph coarsening module.
- **\loss**: This is a variant which does not specifically down-weight the well-classified samples. The focal loss function is replaced by the standard cross entropy loss function.
- **\reward**: This is a variant which uses Micro-F1 as its reward metric.



(a) Benign Buyer

(b) Abusive Buyer

Figure 4.2: Performance gain comparison over variants on the buyer classification task. The background histograms indicate the F1 gain over AGE of each variant. The lines indicate the running time (in seconds).



(a)

(b)

Figure 4.3: Performance gain over Random with different training set sizes and query budgets.

We also record the running time (in seconds) of each variant and summarize the experimental results in Figure 4.2. We have the following findings:

- Removing the graph coarsening module slightly degrades the model's performance, as SAGPool has an attention mechanism that can improve the performance of GNN. Furthermore, this variant takes the longest time compared to all methods in the AL part.

- Changing the reward function to Macro-F1 is more effective for improving the F1 of abusive buyers, as Micro-F1 favors large classes (benign buyer) while

Macro-F1 averages F1 per class.

- When we do not use focal loss, the false positive rate increases, which results in lower Precision and F1. This variant performs the worst, which indicates that the focal loss function contributes the most in ALLIE.

4.4.5 EQ3: Hyperparameter Sensitivity Analysis

We vary the initial training set sizes and query budget to test how ALLIE varies along these dimensions. The buyer classification task on the e-commerce dataset is used as the example task here.

4.4.5.1 Performance under different initial training set sizes

We start ALLIE with $\{50, 150, 250, 350, 450\}$ initial training samples. We run each method five times and report the averaged F1 score in Figure 4.3a. From the results, we see that ALLIE outperforms all the baselines regardless of the initial training set size. Importantly, when the training set sizes are small, ALLIE significantly outperforms the baseline methods.

4.4.5.2 Performance under different query budgets

We train ALLIE with $\{50, 100, 150, 200, 250\}$ budgets, and then evaluate the learned model. All the methods are tested using the same initial training set. We run each method five times and report the averaged F1 score in Figure 4.3b. Again, we see that ALLIE outperforms baselines significantly when the budgets are small.

The above experiments show that ALLIE is indeed well suited to the problem of active learning on graphs when the labeled data is highly imbalanced.

4.5 Conclusion

In this paper, we propose ALLIE, a novel active learning framework designed for large-scale imbalanced graphs. ALLIE leverages a graph policy network to query the candidate nodes to label by optimizing the long-term performance of the GNN classifier. With two balancing strategies, ALLIE can better deal with an imbalanced data distribution compared with several state-of-the-art methods. Moreover, ALLIE has a graph coarsening module which makes it scalable on large-scale applications. Experiments on three benchmark datasets and a real-world shopping website dataset demonstrate the strong performance of ALLIE.

Chapter 5 | Future Work

In this chapter, the motivation, background and main challenge of two directions of future works are introduced.

5.1 Transfer Learning for Misinformation Detection

In recent years, the Data Mining and Machine Learning community has studied and proposed an array of successful methods to accurately detect misinformation using various features. Despite their effectiveness, however, these methods largely depend on the availability of training samples (of fake and real news) with reasonable sizes and qualities. However, the effectiveness of such successful methods deteriorates when applied to new domain, genre, or language (i.e., an English fake news detector may not work well to detect fake news written in Tagalog).

To mitigate these problems, reusing data from a set of relevant tasks becomes a feasible solution. Specifically, we can exploit labeled fake news from other relevant high-resource domains as the teacher and design the learning framework to transfer the knowledge to a low-resource domain as the student. For instance, thousands of crowdsourcing-labeled misinformation tweets in English could be considered as a *high-resource domain* while hundreds of hand-labeled misinformation tweets in Tagalog could be viewed as a *low-resource domain*.

5.2 Thread Recommendation in Online Health Forums

It is necessary to intervene in the misinformation at an early stage, which calls for conveying the evidence-based information to swing users. For example, if a user asks whether herbal tea can treat diabetes on a medical forum, instead of providing a binary prediction label (real/false), I want to provide him/her with related verified threads or answers in the forum, or scientific papers or reports, which is a better way for the user to accept the high-quality information.

One challenge of this problem is to identify the evidence-based information among all the answers. For example, a user asked whether his/her CPAP machine is related to his/her frequently reoccurring stomach issues on a medical forum, and the top-rated answer is "`What's a CPAP machine?`". The difference in meaning may ask for a new method to select the most credible as well as informative answers.

The other challenge of this problem is to align the existing verified information to a newly proposed question. For example, as users may ask a question that is similar to an existing one, we can recommend the existing threads/answers and scientific papers to the users, to prevent users from falling into the myths at an early stage.

Bibliography

[1] Allcott, H. and M. Gentzkow (2017) "Social media and fake news in the 2016 election," *Journal of Economic Perspectives*, **31**(2), pp. 211–36.

[2] Cui, L., S. Wang, and D. Lee (2019) "SAME: sentiment-aware multi-modal embedding for detecting fake news," in *Proceedings of the 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pp. 41–48.

[3] Rashkin, H., E. Choi, J. Y. Jang, S. Volkova, and Y. Choi (2017) "Truth of varying shades: Analyzing language in fake news and political fact-checking," in *EMNLP*, pp. 2931–2937.

[4] Bhatt, G., A. Sharma, S. Sharma, A. Nagpal, B. Raman, and A. Mittal (2018) "Combining Neural, Statistical and External Features for Fake News Stance Identification," in *Companion of the The Web Conference 2018 on The Web Conference 2018*, International World Wide Web Conferences Steering Committee, pp. 1353–1357.

[5] Jin, Z., J. Cao, H. Guo, Y. Zhang, and J. Luo (2017) "Multimodal fusion with recurrent neural networks for rumor detection on microblogs," in *Proceedings of the 2017 ACM on Multimedia Conference*, ACM, pp. 795–816.

[6] Wang, Y., F. Ma, Z. Jin, Y. Yuan, G. Xun, K. Jha, L. Su, and J. Gao (2018) "EANN: Event Adversarial Neural Networks for Multi-Modal Fake News Detection," in *KDD*, ACM, pp. 849–857.

[7] Jin, Z., J. Cao, Y. Zhang, and J. Luo (2016) "News Verification by Exploiting Conflicting Social Viewpoints in Microblogs." in *AAAI*, pp. 2972–2978.

[8] Conroy, N. J., V. L. Rubin, and Y. Chen (2015) "Automatic deception detection: Methods for finding fake news," in *Proceedings of the 78th ASIS&T Annual Meeting: Information Science with Impact: Research in and for the Community*, American Society for Information Science, p. 82.

[9] Shu, K., A. Sliva, S. Wang, J. Tang, and H. Liu (2017) "Fake news detection on social media: A data mining perspective," *ACM SIGKDD Explorations Newsletter*, **19**(1), pp. 22–36.

[10] Jin, Z., J. Cao, Y. Zhang, J. Zhou, and Q. Tian (2017) "Novel visual and statistical image features for microblogs news verification," *IEEE transactions on multimedia*, **19**(3), pp. 598–608.

[11] Zhang, Q., E. Yilmaz, and S. Liang (2018) "Ranking-based Method for News Stance Detection," in *Companion of the The Web Conference 2018 on The Web Conference 2018*, International World Wide Web Conferences Steering Committee, pp. 41–42.

[12] Ruchansky, N., S. Seo, and Y. Liu (2017) "Csi: A hybrid deep model for fake news detection," in *CIKM*, ACM, pp. 797–806.

[13] Qian, F., C. Gong, K. Sharma, and Y. Liu (2018) "Neural User Response Generator: Fake News Detection with Collective User Intelligence." in *IJCAI*, pp. 3834–3840.

[14] Tschiatschek, S., A. Singla, M. Gomez Rodriguez, A. Merchant, and A. Krause (2018) "Fake News Detection in Social Networks via Crowd Signals," in *Companion of the The Web Conference 2018 on The Web Conference 2018*, International World Wide Web Conferences Steering Committee, pp. 517–524.

[15] Wu, L. and H. Liu (2018) "Tracing fake-news footprints: Characterizing social media messages by how they propagate," in *WSDM*, ACM, pp. 637–645.

[16] Parikh, S. B. and P. K. Atrey (2018) "Media-Rich Fake News Detection: A Survey," in *MIPR*, IEEE, pp. 436–441.

[17] Lei, X., X. Qian, and G. Zhao (2016) "Rating prediction based on social sentiment from textual reviews," *IEEE Transactions on Multimedia*, **18**(9), pp. 1910–1921.

[18] Tang, D., B. Qin, T. Liu, and Y. Yang (2015) "User Modeling with Neural Network for Review Rating Prediction." in *IJCAI*, pp. 1340–1346.

[19] Gao, H., J. Tang, X. Hu, and H. Liu (2015) "Content-Aware Point of Interest Recommendation on Location-Based Social Networks." in *AAAI*, pp. 1721–1727.

[20] Nguyen, T. H., K. Shirai, and J. Velcin (2015) "Sentiment analysis on social media for stock movement prediction," *Expert Systems with Applications*, **42**(24), pp. 9603–9611.

[21] Qazvinian, V., E. Rosengren, D. R. Radev, and Q. Mei (2011) "Rumor has it: Identifying misinformation in microblogs," in *EMNLP*, Association for Computational Linguistics, pp. 1589–1599.

[22] Sobhani, P., S. Mohammad, and S. Kiritchenko (2016) "Detecting stance in tweets and analyzing its interaction with sentiment," in *Proceedings of the Fifth Joint Conference on Lexical and Computational Semantics*, pp. 159–169.

[23] Mohtarami, M., R. Baly, J. Glass, P. Nakov, L. Màrquez, and A. Moschitti (2018) "Automatic Stance Detection Using End-to-End Memory Networks," in *NAACL-HLT*, vol. 1, pp. 767–776.

[24] Shu, K., D. Mahudeswaran, S. Wang, D. Lee, and H. Liu (2018) "Fakenewsnet: A data repository with news content, social context and dynamic information for studying fake news on social media," *arXiv preprint arXiv:1809.01286*.

[25] Gilbert, C. H. E. (2014) "Vader: A parsimonious rule-based model for sentiment analysis of social media text," in *ICWSM*.

[26] Simonyan, K. and A. Zisserman (2014) "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*.

[27] Russakovsky, O., J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. (2015) "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, **115**(3), pp. 211–252.

[28] Pennington, J., R. Socher, and C. Manning (2014) "Glove: Global vectors for word representation," in *EMNLP*, pp. 1532–1543.

[29] Nguyen, A., S. Mosadeghi, and C. V. Almario (2017) "Persistent digital divide in access to and use of the Internet as a resource for health information: Results from a California population-based study," *International journal of medical informatics*, **103**, pp. 49–54.

[30] Eysenbach, G., J. Powell, O. Kuss, and E.-R. Sa (2002) "Empirical studies assessing the quality of health information for consumers on the world wide web: a systematic review," *Jama*, **287**(20), pp. 2691–2700.

[31] Scanfeld, D., V. Scanfeld, and E. L. Larson (2010) "Dissemination of health information through social networks: Twitter and antibiotics," *American journal of infection control*, **38**(3), pp. 182–188.

[32] Tsai, C. C., S. Tsai, Q. Zeng-Treitler, and B. Liang (2007) "Patient-centered consumer health social network websites: a pilot study of quality of user-generated health information," in *AMIA Annu Symp Proc*, vol. 1137.

[33] Guo, H., J. Cao, Y. Zhang, J. Guo, and J. Li (2018) “Rumor detection with hierarchical social attention network,” in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pp. 943–951.

[34] Shu, K., L. Cui, S. Wang, D. Lee, and H. Liu (2019) “defend: Explainable fake news detection,” in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 395–405.

[35] Rendle, S., C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme (2009) “BPR: Bayesian personalized ranking from implicit feedback,” in *Proceedings of the twenty-fifth conference on uncertainty in artificial intelligence*, AUAI Press, pp. 452–461.

[36] Potthast, M., J. Kiesel, K. Reinartz, J. Bevendorff, and B. Stein (2017) “A stylometric inquiry into hyperpartisan and fake news,” *arXiv preprint arXiv:1702.05638*.

[37] Vosoughi, S., D. Roy, and S. Aral (2018) “The spread of true and false news online,” *Science*, **359**(6380), pp. 1146–1151.

[38] Ciampaglia, G. L., P. Shiralkar, L. M. Rocha, J. Bollen, F. Menczer, and A. Flammini (2015) “Computational fact checking from knowledge networks,” *PloS one*, **10**(6).

[39] Karagiannis, G., I. Trummer, S. Jo, S. Khandelwal, X. Wang, and C. Yu (2019) “Mining an" anti-knowledge base" from Wikipedia updates with applications to fact checking and beyond,” *Proceedings of the VLDB Endowment*, **13**(4), pp. 561–573.

[40] Huynh, V.-P. and P. Papotti (2019) “A Benchmark for Fact Checking Algorithms Built on Knowledge Bases,” in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 689–698.

[41] Shi, B. and T. Weninger (2016) “Discriminative predicate path mining for fact checking in knowledge graphs,” *Knowledge-based systems*, **104**, pp. 123–133.

[42] Tang, J., Y. Feng, and D. Zhao (2019) “Learning to Update Knowledge Graphs by Reading News,” in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 2632–2641.

[43] Kipf, T. N. and M. Welling (2016) “Semi-supervised classification with graph convolutional networks,” *arXiv preprint arXiv:1609.02907*.

[44] Veličković, P., G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio (2018) "Graph Attention Networks," *International Conference on Learning Representations*, accepted as poster.
URL `https://openreview.net/forum?id=rJXMpikCZ`

[45] Schlichtkrull, M., T. N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, and M. Welling (2018) "Modeling relational data with graph convolutional networks," in *European Semantic Web Conference*, Springer, pp. 593–607.

[46] Busbridge, D., D. Sherburn, P. Cavallo, and N. Y. Hammerla (2019) "Relational Graph Attention Networks," *arXiv preprint arXiv:1904.05811*.

[47] Leskovec, J., D. Huttenlocher, and J. Kleinberg (2010) "Predicting positive and negative links in online social networks," in *Proceedings of the 19th international conference on World wide web*, pp. 641–650.

[48] ——— (2010) "Signed networks in social media," in *Proceedings of the SIGCHI conference on human factors in computing systems*, pp. 1361–1370.

[49] Derr, T., Y. Ma, and J. Tang (2018) "Signed graph convolutional networks," in *2018 IEEE International Conference on Data Mining (ICDM)*, IEEE, pp. 929–934.

[50] Huang, J., H. Shen, L. Hou, and X. Cheng (2019) "Signed Graph Attention Networks," *arXiv preprint arXiv:1906.10958*.

[51] Heider, F. (1946) "Attitudes and cognitive organization," *The Journal of psychology*, **21**(1), pp. 107–112.

[52] Bahdanau, D., K. Cho, and Y. Bengio (2014) "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*.

[53] Kingma, D. P. and J. Ba (2014) "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*.

[54] Ernst, P., A. Siu, and G. Weikum (2015) "KnowLife: a versatile approach for constructing a large knowledge graph for biomedical sciences," *BMC bioinformatics*, **16**(1), p. 157.

[55] Angeli, G., M. J. J. Premkumar, and C. D. Manning (2015) "Leveraging linguistic structure for open domain information extraction," in *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 344–354.

[56] Bordes, A., N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko (2013) "Translating embeddings for modeling multi-relational data," in *Advances in neural information processing systems*, pp. 2787–2795.

[57] Kim, Y. (2014) "Convolutional Neural Networks for Sentence Classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1746–1751.

[58] Le, Q. and T. Mikolov (2014) "Distributed representations of sentences and documents," in *ICML*, pp. 1188–1196.

[59] Shu, K., L. Cui, S. Wang, D. Lee, and H. Liu (2019) "DEFEND: Explainable Fake News Detection," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, Association for Computing Machinery, New York, NY, USA, p. 395–405.
URL `https://doi.org/10.1145/3292500.3330935`

[60] Linmei, H., T. Yang, C. Shi, H. Ji, and X. Li (2019) "Heterogeneous graph attention networks for semi-supervised short text classification," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4823–4832.

[61] Pedregosa, F., G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, et al. (2011) "Scikit-learn: Machine learning in Python," *Journal of machine learning research*, **12**(Oct), pp. 2825–2830.

[62] Li, R., X. Wu, X. Wu, and W. Wang (2020) "Few-shot learning for new user recommendation in location-based social networks," in *Proceedings of The Web Conference 2020*, pp. 2472–2478.

[63] Wang, D., J. Lin, P. Cui, Q. Jia, Z. Wang, Y. Fang, Q. Yu, J. Zhou, S. Yang, and Y. Qi (2019) "A semi-supervised graph attentive network for financial fraud detection," in *2019 IEEE International Conference on Data Mining (ICDM)*, IEEE, pp. 598–607.

[64] Hao, Z., C. Lu, Z. Huang, H. Wang, Z. Hu, Q. Liu, E. Chen, and C. Lee (2020) "ASGN: An active semi-supervised graph neural network for molecular property prediction," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 731–752.

[65] Srinivasan, S., N. S. Rao, K. Subbian, and L. Getoor (2019) "Identifying facet mismatches in search via micrographs," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 1663–1672.

[66] Monti, F., M. M. Bronstein, and X. Bresson (2017) "Geometric matrix completion with recurrent multi-graph neural networks," *arXiv preprint arXiv:1704.06803*.

[67] Rao, N., H.-F. Yu, P. Ravikumar, and I. S. Dhillon (2015) "Collaborative Filtering with Graph Information: Consistency and Scalable Methods." in *NIPS*, vol. 2, Citeseer, p. 7.

[68] Kipf, T. N. and M. Welling (2017) "Semi-Supervised Classification with Graph Convolutional Networks," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, OpenReview.net.
URL `https://openreview.net/forum?id=SJU4ayYgl`

[69] Cai, H., V. W. Zheng, and K. C.-C. Chang (2017) "Active learning for graph embedding," *arXiv preprint arXiv:1705.05085*.

[70] Wu, Y., Y. Xu, A. Singh, Y. Yang, and A. Dubrawski (2019) "Active learning for graph neural networks via node feature propagation," *arXiv preprint arXiv:1910.07567*.

[71] Gao, L., H. Yang, C. Zhou, J. Wu, S. Pan, and Y. Hu (2018) "Active discriminative network representation learning," in *IJCAI International Joint Conference on Artificial Intelligence*.

[72] Chen, X., G. Yu, J. Wang, C. Domeniconi, Z. Li, and X. Zhang (2019) "Activehne: Active heterogeneous network embedding," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, International Joint Conferences on Artificial Intelligence Organization, pp. 2123–2129.

[73] Hu, S., Z. Xiong, M. Qu, X. Yuan, M.-A. Côté, Z. Liu, and J. Tang (2020) "Graph Policy Network for Transferable Active Learning on Graphs," in *NeurIPS*.

[74] Madhawa, K. and T. Murata (2020) "MetAL: Active Semi-Supervised Learning on Graphs via Meta-Learning," in *Asian Conference on Machine Learning*, PMLR, pp. 561–576.

[75] Li, Y., J. Yin, and L. Chen (2020) "SEAL: Semisupervised Adversarial Active Learning on Attributed Graphs," *IEEE Transactions on Neural Networks and Learning Systems*.

[76] Zhu, Y., W. Xu, Q. Liu, and S. Wu (2020) "When Contrastive Learning Meets Active Learning: A Novel Graph Active Learning Paradigm with Self-Supervision," *arXiv preprint arXiv:2010.16091*.

[77] Zhang, R., L. Li, Y. Zhang, and C. Bu (2018) "Imbalanced networked multi-label classification with active learning," in *2018 IEEE International Conference on Big Knowledge (ICBK)*, IEEE, pp. 290–297.

[78] Zhang, Y., P. Zhao, J. Cao, W. Ma, J. Huang, Q. Wu, and M. Tan (2018) "Online adaptive asymmetric active learning for budgeted imbalanced data," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2768–2777.

[79] Aggarwal, U., A. Popescu, and C. Hudelot (2020) "Active learning for imbalanced datasets," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1428–1437.

[80] Aggarwal, C. C., X. Kong, Q. Gu, J. Han, and S. Y. Philip (2014) "Active learning: A survey," in *Data Classification: Algorithms and Applications*, CRC Press, pp. 571–605.

[81] Bachman, P., A. Sordoni, and A. Trischler (2017) "Learning algorithms for active learning," in *international conference on machine learning*, PMLR, pp. 301–310.

[82] Li, X. and Y. Guo (2013) "Adaptive active learning for image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 859–866.

[83] Konyushkova, K., R. Sznitman, and P. Fua (2015) "Introducing geometry in active learning for image segmentation," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2974–2982.

[84] Xu, Y., Y. Hong, H. Ruan, J. Yao, M. Zhang, and G. Zhou (2018) "Using active learning to expand training data for implicit discourse relation recognition," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 725–731.

[85] Dor, L. E., A. Halfon, A. Gera, E. Shnarch, L. Dankin, L. Choshen, M. Danilevsky, R. Aharonov, Y. Katz, and N. Slonim (2020) "Active learning for BERT: An empirical study," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7949–7962.

[86] Shannon, C. E. (2001) "A mathematical theory of communication," *ACM SIGMOBILE mobile computing and communications review*, **5**(1), pp. 3–55.

[87] Kalofolias, V. and N. Perraudin (2018) "Large Scale Graph Learning From Smooth Signals," in *International Conference on Learning Representations*.

[88] Spielman, D. A. and S.-H. Teng (2011) "Spectral sparsification of graphs," *SIAM Journal on Computing*, **40**(4), pp. 981–1025.

[89] Koutis, I., G. L. Miller, and R. Peng (2011) "A nearly-m log n time solver for sdd linear systems," in *2011 IEEE 52nd Annual Symposium on Foundations of Computer Science*, IEEE, pp. 590–598.

[90] Dasarathy, G., N. Rao, and R. Baraniuk (2017) "On computational and statistical tradeoffs in matrix completion with graph information," in *Signal Processing with Adaptive Sparse Structured Representations Workshop SPARS*.

[91] Ying, R., J. You, C. Morris, X. Ren, W. L. Hamilton, and J. Leskovec (2018) "Hierarchical graph representation learning with differentiable pooling," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 4805–4815.

[92] Lee, J., I. Lee, and J. Kang (2019) "Self-attention graph pooling," in *International Conference on Machine Learning*, PMLR, pp. 3734–3743.

[93] Caliskan, S. Y. and P. Tabuada (2014) "Towards Kron reduction of generalized electrical networks," *Automatica*, **50**(10), pp. 2586–2590.

[94] Lafon, S. and A. B. Lee (2006) "Diffusion maps and coarse-graining: A unified framework for dimensionality reduction, graph partitioning, and data set parameterization," *IEEE transactions on pattern analysis and machine intelligence*, **28**(9), pp. 1393–1403.

[95] Hendrickson, B. and R. Leland (1995) "A Multi-Level Algorithm For Partitioning Graphs," in *Supercomputing'95: Proceedings of the 1995 ACM/IEEE Conference on Supercomputing*, IEEE, pp. 28–28.

[96] Wang, L., Y. Xiao, B. Shao, and H. Wang (2014) "How to partition a billion-node graph," in *2014 IEEE 30th International Conference on Data Engineering*, IEEE, pp. 568–579.

[97] Defferrard, M., X. Bresson, and P. Vandergheynst (2016) "Convolutional neural networks on graphs with fast localized spectral filtering," *Advances in neural information processing systems*, **29**, pp. 3844–3852.

[98] Bronstein, M. M., J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst (2017) "Geometric deep learning: going beyond euclidean data," *IEEE Signal Processing Magazine*, **34**(4), pp. 18–42.

[99] Veličković, P., G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio (2018) "Graph Attention Networks," in *International Conference on Learning Representations*.

[100] Oksuz, K., B. C. Cam, S. Kalkan, and E. Akbas (2020) "Imbalance problems in object detection: A review," *IEEE transactions on pattern analysis and machine intelligence*.

[101] Cui, L., S. Biswal, L. M. Glass, G. Lever, J. Sun, and C. Xiao (2020) "Conan: Complementary pattern augmentation for rare disease detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, pp. 614–621.

[102] Liu, Y., X. Ao, Z. Qin, J. Chi, J. Feng, H. Yang, and Q. He (2021) "Pick and Choose: A GNN-based Imbalanced Learning Approach for Fraud Detection," in *Proceedings of the Web Conference 2021*, pp. 3168–3177.

[103] He, H. and E. A. Garcia (2009) "Learning from imbalanced data," *IEEE Transactions on knowledge and data engineering*, **21**(9), pp. 1263–1284.

[104] Buda, M., A. Maki, and M. A. Mazurowski (2018) "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Networks*, **106**, pp. 249–259.

[105] Chawla, N. V., K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer (2002) "SMOTE: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, **16**, pp. 321–357.

[106] He, H., Y. Bai, E. A. Garcia, and S. Li (2008) "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *2008 IEEE international joint conference on neural networks (IEEE world congress on computational intelligence)*, IEEE, pp. 1322–1328.

[107] Seiffert, C., T. M. Khoshgoftaar, J. Van Hulse, and A. Napolitano (2009) "RUSBoost: A hybrid approach to alleviating class imbalance," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, **40**(1), pp. 185–197.

[108] Lin, T.-Y., P. Goyal, R. Girshick, K. He, and P. Dollár (2017) "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, pp. 2980–2988.

[109] Elkan, C. (2001) "The foundations of cost-sensitive learning," in *International joint conference on artificial intelligence*, vol. 17, Lawrence Erlbaum Associates Ltd, pp. 973–978.

[110] Lin, E., Q. Chen, and X. Qi (2020) "Deep reinforcement learning for imbalanced classification," *Applied Intelligence*, **50**(8), pp. 2488–2502.

[111] Williams, R. J. (1992) "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine learning*, **8**(3), pp. 229–256.

[112] Sen, P., G. Namata, M. Bilgic, L. Getoor, B. Galligher, and T. Eliassi-Rad (2008) "Collective classification in network data," *AI magazine*, **29**(3), pp. 93–93.

[113] Adhikari, B., L. Li, N. Rao, and K. Subbian (2021) "Finding Needles in Heterogeneous Haystacks," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, pp. 15232–15239.

[114] Kingma, D. P. and J. Ba (2015) "Adam: A Method for Stochastic Optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings* (Y. Bengio and Y. LeCun, eds.).
URL `http://arxiv.org/abs/1412.6980`

[115] Shi, M., Y. Tang, X. Zhu, D. Wilson, and J. Liu (2020) "Multi-class imbalanced graph convolutional network learning," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20)*.

Vita

Limeng Cui

Education

The Pennsylvania State University *Aug. 2018 – Present*
Ph.D. in Information Sciences and Technology
University of Chinese Academy of Sciences *Sep. 2013 – Jun. 2018*
Ph.D. in Computer Applied Technology
Beijing Institution of Technology *Sep. 2009 – Jun. 2013*
B.Eng. in Software Engineering

Experiences

The Pennsylvania State University *Aug. 2018 – Present*
Research Assistant
Amazon *May. 2021 – Aug. 2021*
Applied Scientist Intern
Facebook *Jun. 2020 – Sep. 2020*
Research Intern
IQVIA *May. 2019 – Aug. 2019*
Machine Learning Intern
Chinese Academy of Sciences *Sep. 2013 – Jun. 2018*
Research Assistant
University of Illinois at Chicago *Oct. 2016 – Oct. 2017*
Student Intern

Publications

[1] **Limeng Cui**, Xianfeng Tang, Sumeet Katariya, Nikhil Rao, Pallav Agrawal, Karthik Subbian, Dongwon Lee. ALLIE: Active Learning on Large-scale Imbalanced Graphs. *The Web Conference (WWW)*, 2022

[2] **Limeng Cui**, Haeseung Seo, Maryam Tabar, Fenglong Ma, Suhang Wang, Dongwon Lee. DETERRENT: Knowledge Guided Graph Attention Network for Detecting Healthcare Misinformation. *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2020

[3] **Limeng Cui**, Siddharth Biswal, Lucas Glass, Greg Lever, Jimeng Sun, Cao Xiao. CONAN: Complementary Pattern Augmentation for Rare Disease Detection. *AAAI Conference on Artificial Intelligence (AAAI)*, 2020

[4] **Limeng Cui**, Kai Shu, Suhang Wang, Dongwon Lee, Huan Liu. dEFEND: A System for Explainable Fake News Detection. *ACM International Conference on Information and Knowledge Management (CIKM)*, 2019 (demo)

[5] **Limeng Cui**, Suhang Wang, Dongwon Lee. SAME: Sentiment-Aware Multi-modal Embedding for Detecting Fake News. *IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, 2019

[6] Kai Shu, **Limeng Cui**, Suhang Wang, Dongwon Lee, Huan Liu. dEFEND: Explainable Fake News Detection. *ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2019