Heterogeneous Graph Attention Networks for Semi-supervised Short Text Classification

Linmei Hu[1], Tianchi Yang[1], Chuan Shi[*1], Houye Ji[1], Xiaoli Li[2]
[1]Beijing University of Posts and Telecommunications, China
[2]Nanyang Technological University, Singapore
{hulinmei,yangtianchi,shichuan,jhy1993}@bupt.edu.cn
xlli@i2r.a-star.edu.sg

Abstract

Short text classification has found rich and critical applications in news and tweet tagging to help users find relevant information. Due to lack of labeled training data in many practical use cases, there is a pressing need for studying semi-supervised short text classification. Most existing studies focus on long texts and achieve unsatisfactory performance on short texts due to the sparsity and limited labeled data. In this paper, we propose a novel heterogeneous graph neural network based method for semi-supervised short text classification, leveraging full advantage of few labeled data and large unlabeled data through information propagation along the graph. In particular, we first present a flexible HIN (heterogeneous information network) framework for modeling the short texts, which can integrate any type of additional information as well as capture their relations to address the semantic sparsity. Then, we propose **H**eterogeneous **G**raph **AT**tention networks (HGAT) to embed the HIN for short text classification based on a dual-level attention mechanism, including node-level and type-level attentions. The attention mechanism can learn the importance of different neighboring nodes as well as the importance of different node (information) types to a current node. Extensive experimental results have demonstrated that our proposed model outperforms state-of-the-art methods across six benchmark datasets significantly.

1 Introduction

With the rapid development of online social media and e-commerce, short texts, such as online news, queries, reviews, tweets, are increasingly widespread on the Internet (Song et al., 2014). Short text classification can be widely applied in many domains, ranging from sentiment analysis to news tagging/categorization and query intent classification (Aggarwal and Zhai, 2012; Meng et al., 2018). In many practical scenarios, the labeled data is scarce, while human labeling is time-consuming and may require expert knowledge (Aggarwal and Zhai, 2012). As a consequence, there is a pressing need for studying semi-supervised short text classification with a relatively small number of labeled training data.

Nevertheless, semi-supervised short text classification is nontrivial due to the following challenges. Firstly, short texts are usually semantically sparse and ambiguous, lacking contexts (Phan et al., 2008). While some methods have been proposed to incorporate additional information such as entities (Wang et al., 2013, 2017), they are unable to consider the relational data such as the semantic relations among entities. Secondly, the labeled training data is limited, which leads to traditional and neural supervised methods (Wang and Manning, 2012; Kim, 2014; Zhang et al., 2015) ineffective. As such, how to make full use of the limited labeled data and large number of unlabeled data has become a key problem for short text classification (Aggarwal and Zhai, 2012). Finally, we need to capture the importance of different information that is incorporated to address sparsity at multiple granularity levels and reduce the weights of noisy information to achieve more accurate classification results.

In this work, we propose a novel *heterogeneous graph neural network based method* for semi-supervised short text classification, which makes full use of both limited labeled data and large unlabeled data by allowing information propagation through our automatically constructed graph. Particularly, we first present a flexible HIN framework for modeling the short texts, which is able to incorporate any additional information (e.g., entities and topics) as well as capture the rich relations among the texts and the additional information. Then, we propose Heterogeneous Graph Attention

networks (HGAT) to embed the HIN for short text classification based on a new dual-level attention mechanism including node-level and type-level attentions. Our HGAT method considers the heterogeneity of different node types. Additionally, the dual-level attention mechanism captures both the importance of different neighboring nodes (reducing the weights of noisy information) and the importance of different node (information) types to a current node. The main contributions of this paper can be summarized as follows:

1) To the best of our knowledge, *this is the first attempt* to model short texts as well as additional information with an HIN and adapt graph neural networks on the HIN for semi-supervised classification.

2) We propose novel *heterogeneous graph attention networks* (HGAT) for the HIN embedding based on a new dual-level attention mechanism which can learn the importance of different neighboring nodes and the importance of different node (information) types to a current node.

3) Extensive experimental results have demonstrated that our proposed HGAT model significantly outperforms seven state-of-the-art methods across six benchmark datasets.

2 Related Work

2.1 Traditional Text Classification

Traditional text classification methods such as SVM (Drucker et al., 1999) need a feature engineering step for text representation. The most commonly used features are BoW and TF-IDF (Blei et al., 2003). Some recent studies (Rousseau et al., 2015; Wang et al., 2016) model texts as graphs and extract path based features for classification. Despite its initial success on formal and well-edited texts, all these methods fail to achieve satisfactory performance on short text classification, due to the insufficient features incurred by short texts. To address the problem, efforts have been made to enrich the semantics of *short texts*. For example, Phan et al. (2008) extracted the latent topics of the short texts with the help of an external corpus. Wang et al. (2013) introduced external entity information from Knowledge Bases, etc. However, these methods are not able to achieve good performance as the feature engineering step relies on domain knowledge.

2.2 Deep neural networks for Text Classification

Deep neural networks which automatically represent texts as embeddings, have been widely used for text classification. Two representative deep neural models, such as RNNs (Liu et al., 2016; Sinha et al., 2018) and CNNs, (Kim, 2014; Shimura et al., 2018) have shown their power in many NLP tasks, including text classification. To adapt it to *short text classification*, several methods have been proposed. For example, Zhang et al. (2015) designs a character-level CNN which alleviates the sparsity by mining different levels of information within the texts. Wang et al. (2017) incorporates the entities and concepts from KBs to enrich the semantics of short texts. However, these methods cannot capture the semantic relations (e.g., entity relations) and rely heavily on the number of training data. Clearly, lacking of training data is still a key bottleneck that prohibits them from successful practical applications.

2.3 Semi-supervised Text Classification

Considering the cost of human labeling and the fact that unlabeled texts also provide valuable information, semi-supervised methods have peen proposed. They can be categorized into two classes: (1) latent variable models (Lu and Zhai, 2008; Chen et al., 2015); and (2) embedding-based models (Meng et al., 2018). The former mainly extend topic model by user-provided seed information and then infer the documents' labels based on posterior category-topic assignment. The latter use seed information to derive embeddings for documents and label names for text classification. For example, PTE (Tang et al., 2015) models the documents, words and labels with graphs and learns text (node) embeddings for classification. Meng et al. (2018) leveraged seed information to generate pseudo-labeled documents for pre-training. Yin et al. (2015) used a semi-supervised learning method based on SVM to label the unlabeled documents in an iterative way. Recently, graph convolutional networks (GCN) have received wide attention for semi-supervised classification (Kipf and Welling, 2017). TextGCN (Yao et al., 2019) models the whole text corpus as a document-word graph and applies GCN for classification. However, all these methods focus on long texts. In addition, they fail to use attention mechanisms to capture important information.

3 Our Proposed Method

In this paper, we propose a novel heterogeneous graph neural network based method for semi-supervised short text classification, which takes full advantage of both limited labeled data and large unlabeled data by allowing information propagation along the graph. Our method includes two steps. Particularly, to alleviate the sparsity of short texts, we first present a flexible HIN framework for modeling the short texts, which can incorporate any additional information as well as capture the rich relations among the short texts and the added information. Then, we propose a novel model HGAT to embed the HIN for short text classification based on a new dual-level attention mechanism. HGAT considers the heterogeneity of different types of information. In addition, the attention mechanism can learn the importance of different nodes (reducing the weights of noisy information) as well as the importance of different node (information) types.

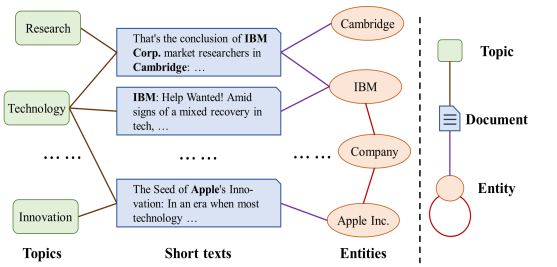
3.1 HIN for Short Texts

We first present the HIN framework for modeling the short texts, which enables integration of any additional information and captures the rich relations among the texts and the added information. In this way, the sparsity of the short texts is alleviated.

Previous studies have exploited latent topics (Zeng et al., 2018) and external knowledge (e.g., entities) from Knowledge Bases to enrich the semantics of the short texts (Wang et al., 2013, 2017). However, they fail to consider the semantic relation information, such as entity relations. Our HIN framework for short texts is flexible for integrating any additional information and modeling their rich relations.

Here, we consider two types of additional information i.e., topics and entities. As shown in Figure 1, we construct the HIN $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ containing the short texts $D = \{d_1, ..., d_m\}$, topics $T = \{t_1, ..., t_K\}$, and entities $E = \{e_1, ..., e_n\}$ as nodes, i.e., $\mathcal{V} = D \cup T \cup E$. The set of edges $\mathcal{E}$ represent their relations. The details of constructing the network are described as follows.

First, we mine the latent topics T to enrich the semantics of short texts using LDA (Blei et al., 2003). Each topic $t_i = (\theta_1, ..., \theta_w)$ (w denotes the vocabulary size) is represented by a probability distribution over the words. We assign each



(a) HIN for short texts (b) HIN schema

Figure 1: An example of HIN for short texts on AGNews.

document to the top P topics with the largest probabilities. Thus, the edge between a document and a topic is built if the document is assigned to the topic.

Second, we recognize the entities E in the documents D and map them to Wikipedia with the entity linking tool TAGME[1]. The edge between a document and an entity is built if the document contains the entity. We take an entity as a whole word and learn the entity embeddings using word2vec[2] based on the Wikipedia corpus. To further enrich the semantics of short texts and advance the information propagation, we consider the relations between entities. Particularly, if the similarity score (cosine similarity) between two entities, computed based on their embeddings, is above a predefined threshold δ, we build an edge between them.

By incorporating the topics, entities and the relations, we enrich the semantics of the short texts and thus greatly benefit the following classification task. For example, as shown in Figure 1, the short text "the seed of Apple's Innovation: In an era when most technology..." is semantically enriched by the relations with the entities "Apple Inc." and "company", as well as the topic "technology". Thus, it can be correctly classified into the category of "business" with high confidence.

3.2 HGAT

We then propose HGAT model (shown in Figure 2) to embed the HIN for short text classification based on a new dual-level attention mechanism including node level and type level. HGAT considers the heterogeneity of different types of information with heterogeneous graph convolution. In addition, the dual-level attention mechanism

[1]https://sobigdata.d4science.org/group/tagme/
[2]https://code.google.com/archive/p/word2vec/

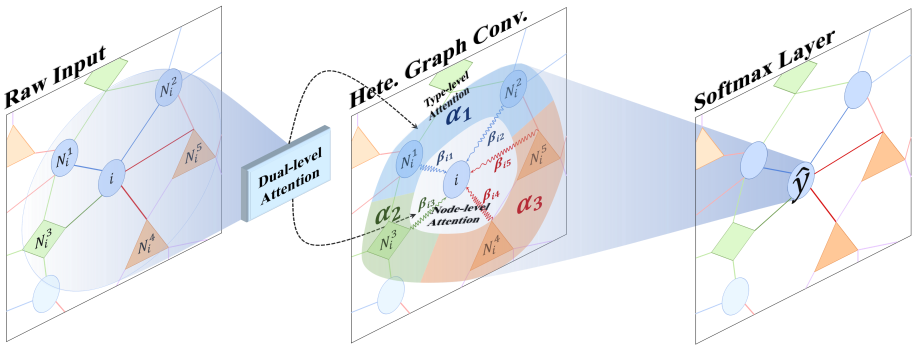


Figure 2: Illustration of our model HGAT.

captures the importance of different neighboring nodes (reducing the weights of noisy information) and the importance of different node (information) types to a specific node. Finally, it predicts the labels of documents through a softmax layer.

3.2.1 Heterogeneous Graph Convolution

We first describe the heterogeneous graph convolution in HGAT, considering the heterogeneous types of nodes (information).

As known, GCN (Kipf and Welling, 2017) is a multi-layer neural network that operates directly on a homogeneous graph and induces the embedding vectors of nodes based on the properties of their neighborhoods. Formally, consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ where $\mathcal{V}$ and $\mathcal{E}$ represent the set of nodes and edges respectively. Let $X \in \mathbb{R}^{|\mathcal{V}| \times q}$ be a matrix containing the nodes with their features $x_v \in \mathbb{R}^q$ (each row x_v is a feature vector for a node v). For the graph $\mathcal{G}$, we introduce its adjacency matrix $A' = A + I$ with added self-connections and degree matrix M, where $M_{ii} = \sum_j A'_{ij}$. Then the layer-wise propagation rule is as follows:

$$H^{(l+1)} = \sigma(\tilde{A} \cdot H^{(l)} \cdot W^{(l)}). \quad (1)$$

Here, $\tilde{A} = M^{-\frac{1}{2}} A' M^{-\frac{1}{2}}$ represents the symmetric normalized adjacency matrix. $W^{(l)}$ is a layer-specific trainable transformation matrix. $\sigma(\cdot)$ denotes an activation function such as ReLU. $H^{(l)} \in \mathbb{R}^{|\mathcal{V}| \times q}$ denotes the hidden representations of nodes in the l^{th} layer. Initially, $H^{(0)} = X$.

Unfortunately, GCN cannot be directly applied to the HIN for short texts due to the node heterogeneity issue. Specifically, in the HIN, we have three types of nodes: documents, topics and entities with different feature spaces. For a document $d \in D$, we use the TF-IDF vector as its feature vector x_d. For the topic $t \in T$, the word distribution is used to represent the topic $x_t = \{\theta_i\}_{i=[1,w]}$. For each entity, to make full use of relevant information, we represent the entity x_v by concatenating its embedding and TF-IDF vector of its Wikipedia description text.

A straightforward way to adapt GCN for the HIN containing different types of nodes $\mathcal{T} = \{\tau_1, \tau_2, \tau_3\}$ is to construct a new large feature space by concatenating together the feature spaces of different types of nodes. For example, each node is denoted as a feature vector with 0 values for the irrelevant dimensions for other types. We name this basic method for adapting GCN to HIN as *GCN-HIN*. However, it suffers from reduced performance since it ignores the heterogeneity of different information types.

To address the issue, we propose the heterogeneous graph convolution, which considers the difference of various types of information and projects them into an implicit common space with their respective transformation matrices.

$$H^{(l+1)} = \sigma(\sum_{\tau \in \mathcal{T}} \tilde{A}_\tau \cdot H_\tau^{(l)} \cdot W_\tau^{(l)}), \quad (2)$$

where $\tilde{A}_\tau \in \mathbb{R}^{|\mathcal{V}| \times |\mathcal{V}_\tau|}$ is the submatrix of $\tilde{A}$, whose rows represent all the nodes and columns represent their neighboring nodes with the type τ. The representation of the nodes $H^{(l+1)}$ is obtained by aggregating information from the features of their neighboring nodes $H_\tau^{(l)}$ with different types τ using different transformation matrix $W_\tau^{(l)} \in \mathbb{R}^{q^{(l)} \times q^{(l+1)}}$. The transformation matrix $W_\tau^{(l)}$ considers the difference of different feature spaces and projects them into an implicit common space $\mathbb{R}^{q^{(l+1)}}$. Initially, $H_\tau^{(0)} = X_\tau$.

3.2.2 Dual-level Attention Mechanism

Typically, given a specific node, different types of neighboring nodes may have different impacts on it. For example, the neighboring nodes of the same

type may carry more useful information. Additionally, different neighboring nodes of the same type could also have different importance. To capture both the different importance at both node level and type level, we design a new dual-level attention mechanism.

Type-level Attention. Given a specific node v, the type-level attention learns the weights of different types of neighboring nodes. Specifically, we first represent the embedding of the type τ as $h_\tau = \sum_{v'} \tilde{A}_{vv'} h_{v'}$, which is the sum of the neighboring node features $h_{v'}$ where the nodes $v' \in \mathcal{N}_v$ and are with the type τ. Then, we calculate the type-level attention scores based on the current node embedding h_v and the type embedding h_τ:

$$a_\tau = \sigma(\mu_\tau^T \cdot [h_v || h_\tau]), \tag{3}$$

where μ_τ is the attention vector for the type τ, $||$ means "concatenate", and $\sigma(\cdot)$ denotes the activation function, such as Leaky ReLU.

Then we obtain the type-level attention weights by normalizing the attention scores across all the types with the softmax function:

$$\alpha_\tau = \frac{\exp(a_\tau)}{\sum_{\tau' \in \mathcal{T}} \exp(a_{\tau'})}. \tag{4}$$

Node-level Attention. We design the node-level attention to capture the importance of different neighboring nodes and reduce the weights of noisy nodes. Formally, given a specific node v with the type τ and its neighboring node $v' \in \mathcal{N}_v$ with the type τ', we compute the node-level attention scores based on the node embeddings h_v and $h_{v'}$ with the type-level attention weight $\alpha_{\tau'}$ for the node v':

$$b_{vv'} = \sigma(\nu^T \cdot \alpha_{\tau'} [h_v || h_{v'}]), \tag{5}$$

where ν is the attention vector. Then we normalize the node-level attention scores with the softmax function:

$$\beta_{vv'} = \frac{\exp(b_{vv'})}{\sum_{i \in \mathcal{N}_v} \exp(b_{vi})}. \tag{6}$$

Finally, we incorporate the dual-level attention mechanism including type-level and node-level attentions into the heterogeneous graph convolution by replacing Eq. 2 with the following layer-wise propagation rule:

$$H^{(l+1)} = \sigma(\sum_{\tau \in \mathcal{T}} \mathcal{B}_\tau \cdot H_\tau^{(l)} \cdot W_\tau^{(l)}). \tag{7}$$

Here, $\mathcal{B}_\tau$ represents the attention matrix, whose element in the v^{th} row v'^{th} column is $\beta_{vv'}$ in Eq. 6.

3.3 Model Training

After going through an L-layer HGAT, we can get the embeddings of nodes (including short texts) in the HIN. The short text embeddings $H^{(L)}$ are then fed to a softmax layer for classification. Formlly,

$$Z = \text{softmax}(H^{(L)}), \tag{8}$$

During model training, we exploit the cross-entropy loss over training data with the L2-norm. Formally,

$$\mathcal{L} = -\sum_{i \in D_{\text{train}}} \sum_{j=1}^{C} Y_{ij} \cdot \log Z_{ij} + \eta \, \|\Theta\|_2, \tag{9}$$

where C is the number of classes, D_{train} is the set of short text indices for training, Y is the corresponding label indicator matrix, Θ is model parameters, and η is regularization factor. For model optimization, we adopt the gradient descent algorithm.

4 Experiments

In this section, we evaluate the empirical performance of different methods for semi-supervised short text classification.

4.1 Experimental Setup

4.1.1 Datasets

We conduct extensive experiments on 6 benchmark short text datasets: AGNews, Snippets, Ohsumed, TagMyNews, MR and Twitter.

AGNews: This dataset is adopted from Zhang et al. (2015). We randomly select 6,000 pieces of news from AGNews, evenly distributed into 4 classes.

Snippets: This dataset is released by Phan et al. (2008). It is composed of the snippets returned by a web-search engine.

Ohsumed[3]: We use the benchmark bibliographic classification dataset released by Yao et al. (2019), where the documents with multiple labels are removed. We use the titles for short text classification.

TagMyNews: We use the news titles as instances from the benchmark classification dataset

[3]http://disi.unitn.it/moschitti/corpora.htm

	#docs	#tokens	#entities	#classes
AGNews	6,000	18.4	0.9 (72%)	4
Snippets	12,340	14.5	4.4 (94%)	8
Ohsumed	7,400	6.8	3.1 (96%)	23
TagMyNews	32,549	5.1	1.9 (86%)	7
MR	10,662	7.6	1.8 (76%)	2
Twitter	10,000	3.5	1.1 (63%)	2

Table 1: Statistics of the datasets.

released by Vitale et al. (2012), which contains English news from really simple syndication (RSS) feeds.

MR: It is a movie review dataset, in which each review only contains one sentence (Pang and Lee, 2005). Each sentence is annotated with positive or negative for binary sentiment classification.

Twitter: This dataset is provided by NLTK[4], a library of Python, which is also a binary sentiment classification dataset.

For each dataset, we randomly select 40 labeled documents per class, half of which for training and the other half for validation. Following Kipf and Welling (2017), all the left documents are for testing, which are also used as unlabeled documents during training.

We preprocess all the datasets as follows. We remove non-English characters, the stop words, and low-frequency words appearing less than 5 times. Table 1 shows the statistics of the datasets, including the number of documents, the number of average tokens and entities, the number of classes, and the proportion of texts containing entities in parentheses. In our datasets, most of the texts (around 80%) contain entities.

4.1.2 Baselines

To comprehensively evaluate our proposed method for semi-supervised short text classification, we compare it with the following nine state-of-the-art methods:

SVM: SVM classifiers using TF-IDF features and LDA features (Blei et al., 2003), are denoted as SVM+TFIDF and SVM+LDA, respectively.

CNN: CNN (Kim, 2014) with 2 variants: 1) CNN-rand, whose word embeddings are randomly initialized, and 2) CNN-pretrain, whose word embeddings are pre-trained with Wikipedia Corpus.

LSTM: LSTM (Liu et al., 2016) with and without pre-trained word embeddings, named LSTM-rand and LSTM-pretrain, respectively.

[4]https://www.nltk.org/

PTE: A semi-supervised representation learning method for text data (Tang et al., 2015). It firstly learns word embedding based on the heterogeneous text networks containing three bipartite networks of words, documents and labels, then averages word embeddings as document embeddings for text classification.

TextGCN: Text GCN (Yao et al., 2019) models the text corpus as a graph containing documents and words as nodes, and applies GCN for text classification.

HAN: HAN (Wang et al., 2019) embeds HINs by first converting an HIN to several homogeneous sub-networks through pre-defined meta-paths and then applying graph attention networks.

For fair comparison, all of the above baselines, such as SVMs, CNN and LSTM, have used entity information.

4.1.3 Parameter Settings

We choose the parameter values of K, T and δ that achieve the best results on the validation set. To construct HIN for short texts, we set the number of topics $K = 15$ in LDA for the datasets AGNews, TagMyNews, MR and Twitter. We set $K = 20$ for Snippets and $K = 40$ for Ohsumed. For all the datasets, each document is assigned to top $P = 2$ topics with the largest probabilities. The similarity threshold δ between entities is set $\delta = 0.5$.

Following previous studies (Vaswani et al., 2017), we set the hidden dimension of our model HGAT and other neural models to $d = 512$ and the dimension of pre-trained word embeddings to 100. We set the layer number L of HGAT, GCN-HIN and TextGCN as 2. For model training, we set the learning rate as 0.005, dropout rate as 0.8 and the regularization factor $\eta = 5e\text{-}6$. Early stopping is applied to avoid overfitting.

4.2 Experimental Results

Table 2 shows the classification accuracy of different methods on 6 benchmark datasets. We can see that our methods significantly outperform all the baselines by a large margin, which shows the effectiveness of our proposed method on semi-supervised short text classification.

The traditional method SVMs based on the human-designed features, achieve better performance than the deep models with random initialization, i.e., CNN-rand and LSTM-rand in most cases. While CNN-pretrain and LSTM-pretrain using the pre-trained vectors achieve significant

Dataset	SVM +TFIDF	SVM +LDA	CNN -rand	CNN -pretrain	LSTM -rand	LSTM -pretrain	PTE	TextGCN	HAN	HGAT
AGNews	57.73	65.16	32.65	67.24	31.24	66.28	36.00	67.61	62.64	**72.10***
Snippets	63.85	63.91	48.34	77.09	26.38	75.89	63.10	77.82	58.38	**82.36***
Ohsumed	41.47	31.26	35.25	32.92	19.87	28.70	36.63	41.56	36.97	**42.68***
TagMyNews	42.90	21.88	28.76	57.12	25.52	57.32	40.32	54.28	42.18	**61.72***
MR	56.67	54.69	54.85	58.32	52.62	60.89	54.74	59.12	57.11	**62.75***
Twitter	54.39	50.42	52.58	56.34	54.80	60.28	54.24	60.15	53.75	**63.21***

Table 2: Test accuracy (%) of different models on six standard datasets. The second best results are underlined. The note * means our model significantly outperforms the baselines based on t-test ($p < 0.01$).

Dataset	GCN -HIN	HGAT w/o ATT	HGAT -Type	HGAT -Node	HGAT
AGNews	70.87	70.97	71.54	71.76	**72.10***
Snippets	76.69	80.42	81.68	81.93	**82.36***
Ohsumed	40.25	41.31	41.95	42.17	**42.68***
TagMyNews	56.33	59.41	60.78	61.29	**61.72***
MR	60.81	62.13	62.27	62.31	**62.75***
Twitter	61.59	62.35	62.95	62.45	**63.21***

Table 3: Test accuracy (%) of our variants.

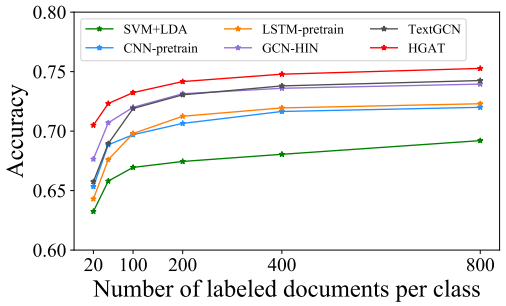


Figure 3: The test accuracy with different number of labeled documents.

improvements and outperform SVMs. The graph based model PTE achieves inferior performance compared to CNN-pretrain and LSTM-pretrain. The reason may be that PTE learns text embeddings based on word co-occurrences, which, however, are sparse in short text classification. Graph neural network based models TextGCN and HAN achieve comparable results with the deep models CNN-pretrain and LSTM-pretrain. Our model HGAT consistently outperforms all the state-of-the-art models by a large margin, which shows the effectiveness of our proposed method. The reasons include that 1) we construct a flexible HIN framework for modeling the short texts, enabling integration of additional information to enrich the semantics and 2) we propose a novel model HGAT to embed the HIN for short text classification based on a new dual-level attention mechanism. The attention mechanism not only captures the importance of different neighboring nodes (reducing the weights of noisy information) but also the importance of different types of nodes.

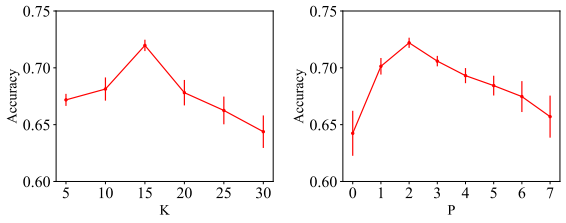
4.2.1 Comparison of Variants of HGAT

We also compare our model HGAT with some variants to validate the effectiveness of our model. As shown in Table 3, we compare our HGAT with four variant models. The basic model GCN-HIN directly applies GCN on our constructed HIN for short texts by concatenating the feature spaces of different types of information. It does not consider the heterogeneity of various information types. HGAT w/o ATT considers the heterogeneity through our proposed heterogeneous graph convolution, which projects different types of information to an implicit common space with respective transformation matrices. HGAT-Type and HGAT-Node respectively consider only type-level attention and node-level attention.

We can see from Table 2, HGAT w/o ATT consistently outperforms GCN-HIN on all datasets, demonstrating the effectiveness of our proposed heterogeneous graph convolution which considers the heterogeneity of various information types. HGAT-Type and HGAT-Node further improve HGAT w/o ATT by capturing the importance of different information (reducing the weights of noisy information). HGAT-Node achieves better performance than HGAT-Type, indicating that node-level attention is more important. Finally, HGAT significantly outperforms all the variants by considering the heterogeneity and applying dual-level attention mechanism including node-level and type-level attentions.

4.2.2 Impact of Number of Labeled Docs

We choose 6 representative methods with the best performance: SVM+LDA, CNN-pretrain, LSM-



(a) K: Number of topics. (b) P: Top P relevant topics.

Figure 4: The average accuracy with different number of topics and top relevant topics on AGNews.

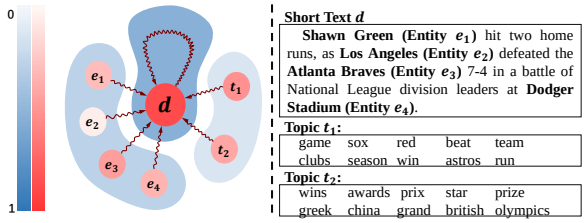


Figure 5: Visualization of the dual-level attention including node-level attention (shown in red) and type-level attention (shown in blue). Each topic t is represented by top 10 words with highest probabilities.

pretrain, GCN-HIN, TextGCN and HGAT, to study the impact of the number of labeled documents. Particularly, we vary the number of labeled documents per class and compare their performance on the AGNews dataset. We run each method 10 times and report the average performance. As shown in Figure 3, with the increase of labeled documents, all the methods achieve better results in terms of accuracy. Generally, the graph based methods GCN-HIN, TextGCN and HGAT achieve better performance, indicating that graph-based methods can make better use of limited labeled data through information propagation. Our method outperforms all the other methods consistently. When fewer labeled documents are provided, the baselines exhibit obvious performance drop, while our model still achieves relatively high performance. It demonstrates that our method can more effectively take advantage of the limited labeled data for short text classification. We believe our method benefits from the flexible HIN and the proposed heterogeneous graph attention networks with dual-level attention.

4.2.3 Parameter Analysis

Figure 4 (a) and (b) show the test accuracy of our HGAT model on the AGNews dataset with different number of topics K and Top P relevant topics assigned to a document. As we can see clearly, for the number of topics, the test accuracy first increases with the increase of the number of topics, reaching the highest value at 15; it falls when its number is larger than 15. We also tried the different numbers of topics for baselines, and have observed that the best K is the same as in our model. This is consistent with the intuition that the number of topics should fit the dataset, i.e., it should be model free. For the number of top relevant topics P assigned to the documents, the test accuracy first increases with the increase of P and then decreases when P is larger than 2. In our experiments, the two parameters are set based on the validation set of each dataset.

4.2.4 Case Study

As Figure 5 shows, we take a short text from AGNews as an example (which is classified to the class of sports correctly) to illustrate the dual-level attention of HGAT. The type-level attention assigns high weight (0.7) to the short text itself, while lower weights (0.2 and 0.1) to entities and topics. It means that the text itself contributes more for classification, than the entities and topics. The node-level attention assigns different weights to neighboring nodes. The node-level weights of nodes belonging to a same type sum to 1. As we see, the entities e_3 (Atlanta Braves, a baseball team), e_4 (Dodger Stadium, a baseball gym), e_1 (Shawn Green, a baseball player) have higher weights than e_2 (Los Angeles, referring to a city at most time). The topics t_1 (game) and t_2 (win) have almost the same importance for classifying the text to the class of sports. The case study shows that our proposed dual-level attention can capture key information at multiple granularities for classification and reduce the weights of noisy information.

5 Conclusion

In this paper, we propose a novel heterogeneous graph neural network based method for semi-supervised short text classification, which takes full advantage of both limited labeled and large unlabeled data by information propagation. Particularly, we first present a flexible HIN framework for modeling the short texts, which can integrate any additional information and capture their rich relations to address the semantic sparsity of short texts. Then, we propose a novel model HGAT to embed the HIN based on a dual-level attention mechanism including node-level and type-level at-

tentions. HGAT considers the heterogeneity of various information types by projecting them into an implicit common space. Additionally, the dual-level attention captures the key information at multiple granularity levels and reduces the weights of noisy information. Extensive experimental results demonstrated that our proposed model significantly outperforms the state-of-the-art methods across six benchmark datasets consistently.

As our model HGAT is a general HIN embedding approach, it would be interesting to apply it to other tasks, e.g., HIN based recommendation.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 61806020, 61772082, 61702296), the National Key Research and Development Program of China (2017YFB0803304), the Beijing Municipal Natural Science Foundation (4182043), the CCF-Tencent Open Fund, and the Fundamental Research Funds for the Central Universities.

References

Charu C Aggarwal and ChengXiang Zhai. 2012. A survey of text classification algorithms. In *Mining text data*, pages 163–222. Springer.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.

Xingyuan Chen, Yunqing Xia, Peng Jin, and John Carroll. 2015. Dataless text classification with descriptive lda. In *AAAI*.

Harris Drucker, Donghui Wu, and Vladimir N Vapnik. 1999. Support vector machines for spam categorization. *IEEE Transactions on Neural Networks*, 10(5):1048–1054.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *EMNLP*, pages 1746–1751.

Thomas N Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *ICLR*.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2016. Recurrent neural network for text classification with multi-task learning. In *IJCAI*, pages 2873–2879. AAAI Press.

Yue Lu and Chengxiang Zhai. 2008. Opinion integration through semi-supervised topic modeling. In *WWW*, pages 121–130. ACM.

Yu Meng, Jiaming Shen, Chao Zhang, and Jiawei Han. 2018. Weakly-supervised neural text classification. In *CIKM*, pages 983–992. ACM.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *ACL*, pages 115–124. Association for Computational Linguistics.

Xuan-Hieu Phan, Le-Minh Nguyen, and Susumu Horiguchi. 2008. Learning to classify short and sparse text & web with hidden topics from large-scale data collections. In *WWW*, pages 91–100. ACM.

François Rousseau, Emmanouil Kiagias, and Michalis Vazirgiannis. 2015. Text categorization as a graph classification problem. In *ACL*, volume 1, pages 1702–1712.

Kazuya Shimura, Jiyi Li, and Fumiyo Fukumoto. 2018. HFT-CNN: Learning hierarchical category structure for multi-label short text categorization. In *EMNLP*, pages 811–816, Brussels, Belgium. Association for Computational Linguistics.

Koustuv Sinha, Yue Dong, Jackie Chi Kit Cheung, and Derek Ruths. 2018. A hierarchical neural attention-based text classifier. In *EMNLP*, pages 817–823, Brussels, Belgium. Association for Computational Linguistics.

Ge Song, Yunming Ye, Xiaolin Du, Xiaohui Huang, and Shifu Bie. 2014. Short text classification: A survey. *Journal of Multimedia*, 9(5):635.

Jian Tang, Meng Qu, and Qiaozhu Mei. 2015. Pte: Predictive text embedding through large-scale heterogeneous text networks. In *SIGKDD*, pages 1165–1174. ACM.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 5998–6008. Curran Associates, Inc.

Daniele Vitale, Paolo Ferragina, and Ugo Scaiella. 2012. Classification of short texts by deploying topical annotations. In *ECIR*, pages 376–387. Springer.

Chenguang Wang, Yangqiu Song, Haoran Li, Ming Zhang, and Jiawei Han. 2016. Text classification with heterogeneous information network kernels. In *AAAI*, pages 2130–2136.

Jin Wang, Zhongyuan Wang, Dawei Zhang, and Jun Yan. 2017. Combining knowledge with deep convolutional neural networks for short text classification. In *IJCAI*, volume 350.

Sida Wang and Christopher D Manning. 2012. Baselines and bigrams: Simple, good sentiment and topic classification. In *ACL*, pages 90–94. Association for Computational Linguistics.

Xiang Wang, Ruhua Chen, Yan Jia, and Bin Zhou. 2013. Short text classification using wikipedia concept based document representation. In *ICITA*, pages 471–474. IEEE.

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

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Graph convolutional networks for text classification. *AAAI*.

Chunyong Yin, Jun Xiang, Hui Zhang, Jin Wang, Zhichao Yin, and Jeong-Uk Kim. 2015. A new svm method for short text classification based on semi-supervised learning. In *AITS*, pages 100–103. IEEE.

Jichuan Zeng, Jing Li, Yan Song, Cuiyun Gao, Michael R Lyu, and Irwin King. 2018. Topic memory networks for short text classification. In *EMNLP*, pages 3120–3131.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *NIPS*, pages 649–657.