Gathering cyber-threat intelligence from Dark Web forums using Unsupervised Learning methods*

Raj Sanjay Shah[1[0000−1111−2222−3333]], Shray Mathur[2,3[1111−2222−3333−4444]], and Vinti Agarwal[3[2222−−3333−4444−5555]]

Dept. of Computer Science and Information Systems,
Birla Institute of Technology and Science Pilani, Pilani, India
{f20171181, f20171180, vinti.agarwal}@pilani.bits-pilani.ac.in

Abstract. The southern poverty law center has linked more than 100 hate crimes with the iron march dark web forum. Several members of the US armed forces and an ICE detention center captain were found to be users of the forum. Recently, researchers at the Center on Terrorism, Extremism, and Counter-terrorism used the forum data to train GPT-3 autoregressive language model. They found that the trained model could be used for misinformation and radicalization purposes. This has brought lot of scrutiny on online forums such as this one. In this paper, we plan to use the forum to build an unsupervised framework to predict emerging threats. We do this by generating vectors of all the nodes in the forum text graph.

Keywords: Dark Web Mining · Graph Convolutional Networks · Variational Graph Autoencoder · Structural Deep Network Embedding · Self Organizing Maps

1 Preliminary analysis of the dataset

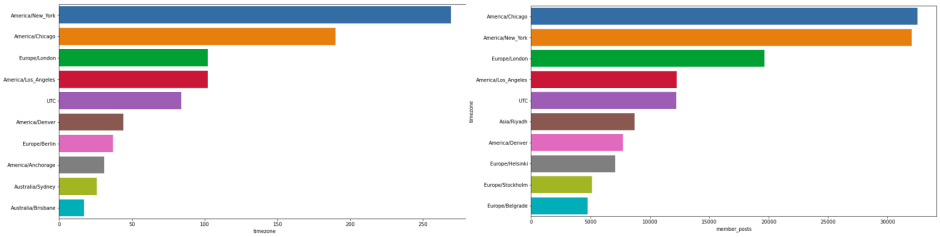
Table 1: Information about the Data set

Number of members	15218
Number of (last) active members	1207
Highest Member Posts	7715
Member email(forum founder)	slavros_a@mail.ru

Listing 1.1: Top twenty five most frequent words:

```
keywords = ['ipsquote','data','core','would','div',
```

* Supported by organization BITS Pilani.



(a) Timezones with most members (b) Timezones with most activity

Fig. 1: Geographic Statistics of the data

	member_id	member_name	member_posts	timezone
0	1	Александр Славрос	7715	Asia/Riyadh
19	9288	Змајевит	3783	Europe/Belgrade
3	132	Myrrysmies	3756	Europe/Helsinki
26	9571	Captain Oakleaf	2950	America/New_York
31	9668	MOONLORD	2669	America/Denver
7	6113	Aquila	2500	America/New_York
21	9353	Hyperborean	1337	Europe/Bucharest
12	7636	Богатырь	1269	America/New_York
4	274	Palmer	1170	America/Chicago
22	9446	NotEqual	1160	Asia/Seoul

Fig. 2: Most Influential/active users

```
'span','emoticons','class','know','conversation',
'messaging','messenger','forum','also','username',
'well','style','one','cite','timestamp',
'dux','think','fascist','new','thanks']
```

Some observations:

1. While the America New York timezone had the most members, the America Chicago timezone is the most active.
2. The London timezone has fewer members, but ranked third in active posts.
3. The Asia/Riyadh timezone which is linked to one Russian e-mail ranks 6th in active posts.

2 Graph Convolutional Neural Network

Many important real-world datasets come in the form of graphs or networks: social networks, knowledge graphs, protein-interaction networks, the World Wide

Web, etc. Yet, until recently, very little attention has been devoted to the generalization of neural network models to such structured datasets.

In the last couple of years, a number of papers re-visited this problem of generalizing neural networks to work on arbitrarily structured graphs ([1]; [2]; [3]), some of them now achieving very promising results in domains that have previously been dominated by, e.g., kernel-based methods, graph-based regularization techniques and others.

Graph Convolutional Networks are a very powerful neural network architecture for machine learning on graphs. Given a graph G = (V, E), a GCN takes as input:

- An input feature matrix N * F, X, where N is the number of nodes and F is the number of input features for each node.
- An N * N matrix representation of the graph or the adjacency matrix

We can now represent each layer of the GCN as a simple non-linear activation similar to what is seen in typical neural network, however this will lead to two major limitations that need to be addressed. First, to incorporate features of the node itself we have to include self loops in the graph. This is acheived by adding the identity matrix to the given adjaceny matrix. Second, is that nodes with large degrees will have large values in their feature representation while nodes with small degrees will have small values. The feature representations can be normalized by node degree by transforming the adjacency matrix A by multiplying it with the inverse degree matrix D. As introduced in [3] we finally use the layer wise propagation rule as stated below.

$$f(H^l, A) = \sigma(\hat{D}^{-1/2}\hat{A}\hat{D}^{-1/2}H^{(l)}W^{(l)})$$

$\hat{A} = A + I$ where I is the Identity matrix and $\hat{D}$ is the is the diagonal degree matrix of $\hat{A}$,

3 Kohonen Self Organizing Maps

Self organizing maps [4] are a type of Artificial Neural Networks whose training is unsupervised in nature and produces a two-dimensional map. It is a method for dimensionality reduction and thereafter uses the the output to visualise the data. It differs from traditional dimensionality reduction techniques by using a competitive strategy to learn as opposed to error based learning. This visualization can be used in conjunction of clustering to visualise similarity between clusters.

The architecture used by us can be described by the following points.

1. We use a hexagonal topology for creating and plotting a 2-d lattice of network nodes. The topology consists of 25 * 25 nodes.
2. We calculate the best matching unit by using Euclidean distance (activation distance).

3. The best matching unit's local neighbourhood is determined by the following exponential decay function: $\sigma(t) = \sigma_0 e^{-(t/\lambda)}$. The initial value of σis 1.5.
4. The decay of learning rate is calculated using this equation : $L(t) = L_0 e^{-(t/\lambda)}$. The initial learning rate is 0.7.

Self organizing maps allows us to visualize high dimensional data and understand the similarity between clusters.

4 The graph structure

The iron march forum has a reddit like structure as shown in figure XXX. In each topic, there are a multiple posts that may or may not be by the same users. We build the text-graph of the forum by modifying the text graph generation proposed by L. Yao et al [8].

There are three types of nodes in the text graph of the iron march forum: Topics, Posts and Users. We calculate the Topic-Word and Topic-User edges weights as their TF-IDF value. The edges are built based on word occurrence in Topics (Topic-word edges), user occurrence in Topics (Topic-user edges) and word co-occurrence in the whole corpus (word-word edges). For weights of word-word edges we use a fixed size sliding window to calculate a Point wise Mutual Information measure. A sub graph of the generated text graph can be seen in figure 3.

5 Embeddings

In this section, we present the Structural Deep Network Embeddings and the Variational Graph Autoencoders. Network embedding is an important method to learn low-dimensional representations of vertexes in networks, aiming to capture and preserve the network structure. Almost all the existing network embedding methods adopt shallow models. However, since the under- lying network structure is complex, shallow models cannot capture the highly non-linear network structure, resulting in sub-optimal network representations.

5.1 Structural Deep Network Embedding (SDNE)

SDNE uses an auto-encoder structure to optimize the first-order and second-order similarities at the same time and the learned vector representation can retain the local and global structure [7]. By jointly optimizing the First Order and Second Order Similairty in the autoencoder deep model, the method can preserve both the local and global network structure and is robust to sparse networks.

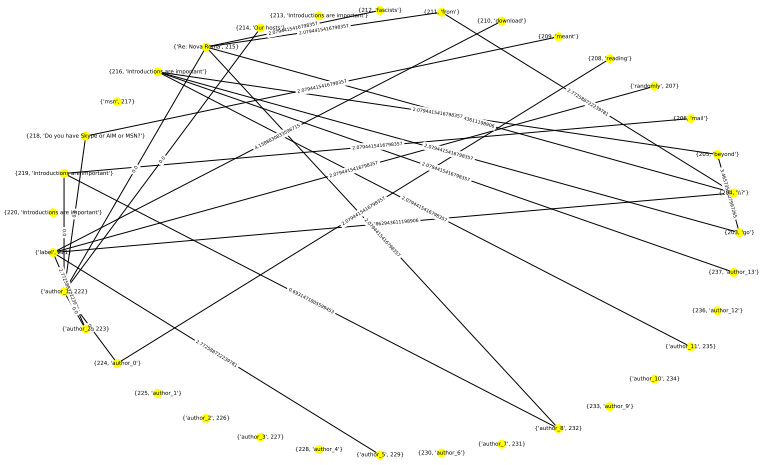


Fig. 3: Text Graph of the Iron March Forum

First Order Similarity The first-order similarity is the local pairwise similarity only between the vertices linked by edges, which characterizes the local network structure.The below loss functions incurs a penalty when similar vertexes are mapped far away in the embedding space.

$$L_{1st} = \sum_{i,j=1}^{n} s_{i,j} ||y_i^{(k)} - y_j^{(k)}||$$

Here, $s_{i,j} \in \{0, 1\}$ depending on whether there is an edge between node i and node j and y_i^k, y_j^k denote the embeddings for node i and node j respectively.

Second Order Similarity This is used to capture global network structure. Due to the sparsity of networks, the number of non-zero elements in the adjacency matrix is far less than that of zero elements. If we directly use the adjacency matrix as the input to the traditional autoencoder, it is more prone to reconstruct the zero elements in S. However, this is not what we want. To address this a weighted loss function is used, which has a higher penalty coefficient for non-zero elements. The revised loss function is given below:

$$L_{2nd} = \sum_{i=1}^{n} ||(\hat{x}_i - x_i) \circ b_i||$$

where, $b_i = \{b_{i,j}\}_{j=1}^{n}$. If $s_{i,j} = 0, b_{i,j} = 1, else b_{i,j} = b > 1$. $\hat{x}_i$ represents the autoencoder output for the instance x_i

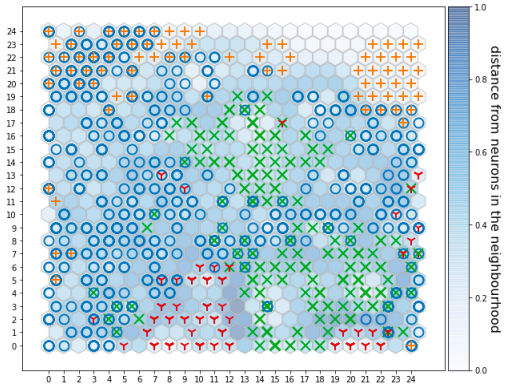


Fig. 4: Self organizing maps for different clustering algorithms on the SDNE Embedding

5.2 Variational Graph Autoencoder (VGAE)

We use the implementation of the Graphical Convolutional Neural network-based variational autoencoder proposed by Kipf and Welling [3]. The VGAE model learns latent variables for an undirected graph, which are a part of a distribution rather than a single point. These can be best understood by the set of equations shown below:
Let the feature matrix, X be an identity matrix with of size (node_size , node_size). The adjacency matrix, Adj be a matrix of size (node_size , node_size) with edge weights as defined in the previous section.

$$A = D^{-1/2} Adj D^{-1/2}$$

where D is the degree matrix, and where $D_{ii} = \sum_j A_{ij}$
The Encoder Representation is given by table 2. W_0 and W_1 are weight matrices of the respective layers.

We then use the parameterization trick: $z = \mu + \sigma\epsilon$, where ϵ belongs to normal distribution with mean as zero and standard deviation as 1.
The decoder is represented by the inner product between latent variable Z. The output of the reconstructed adjacency matrix:

$$\hat{Adj} = \sigma(ZZ^T)$$

where σ() represents the logistic sigmoid function.

Table 2: Encoder representation

Layers	Equations
1^{st} layer	$X^1\ (GCN(X,A)) = RELU(AXW_0)$
2^{nd} layer	$\mu\ (GCN_\mu(X^1,A)) = RELU(AX^1W_1)$
	$log\sigma^2\ (GCN_\sigma(X^1,A)) = RELU(AX^1W_1)$

The loss function of the variational graph autoencoder is defined as:

$$Loss = E_{q(Z|X,A)}logp(A|Z) - KL[q(Z|X,A)||p(Z)]$$

The first term $E_{q(Z|X,A)}logp(A|Z)$ gives the reconstruction error for the adjacency matrix. The second term is the Kulback Leibler Divergence, which compares the output of our latent space with a normal distribution $N(0,1)$ and therefore regularizes our latent space to a gaussian distribution.

Link Prediction and Embedding Generation The model recreates a modified version of the adjacency matrix with all the edge weights set to one or zero, in accordance to the presence or the absence of an edge. While the model tries to reconstruct the modified adjacency matrix, we use the original adjacency matrix for the message passing abilities of the Graphical Neural Networks. This allows us to capture the graph topology accurately. The size of the first hidden layer is 32 and the size of the second hidden layer is 16, thus the latent space consists of 16 dimensions. Experimental results work best on these sizes of the hidden layers. We run and compare two models: Graph Autoencoder (GAE), Variational Graph Autoencoder(VGAE) for 200 epochs and the results for them are seen in table 3.

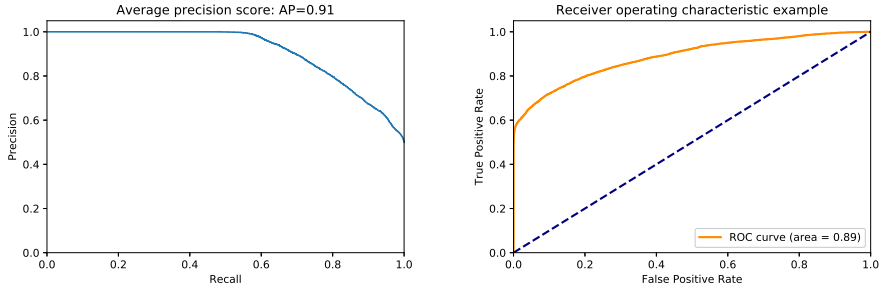
Table 3: Link Prediction

Model	GAE	VGAE
Test set: Area under the curve (ROC)	82.5780%	**89.4462%**
Test set: Average Precision	76.3090%	**91.4099%**

The receiver operating characteristic curve and the average precision for the variational graph autoencoder can be observed in the figure 5a and figure 5b.

6 Embedding Quality Analysis

We determine the quality of the generated embedding layer by analysing the outputs of the different clustering algorithms like Affinity Propagation, KMeans,



(a) Average Precision - VGAE (b) ROC curve - VGAE

Fig. 5: VGAE link prediction metrics

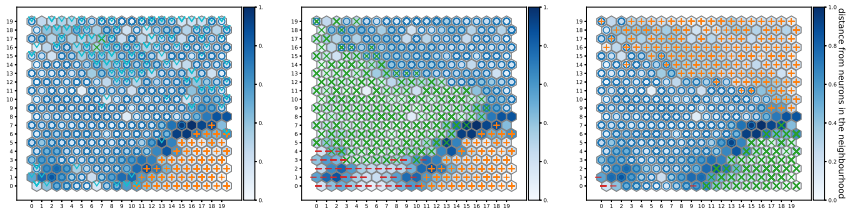
Spectral Clustering, DBScan using self organizing maps and silhouette score. We determine the parameters of the clustering algorithms based on the silhouette scores. The silhouette scores for different algorithms and parameters can be seen in table 4.

Table 4: Silhouette Score - Clustering Algorithms on the generated embeddings by VGAE

Model	Score
Affinity Propagation	0.327
Kmeans (k=2)	0.572
Kmeans (k=3)	**0.614**
Kmeans (k=4)	0.411
Kmeans (k=5)	0.318
Kmeans (k=6)	0.324
Kmeans (k=7)	0.256
DBScan (eps = 0.5)	0.407
DBScan (eps = 1.0)	0.639
DBScan (eps = 1.5)	0.735
DBScan (eps = 2.0)	0.792
DBScan (eps = 2.3)	**0.797**
Spectral Clustering (k=2)	0.628
Spectral Clustering (k=3)	0.632
Spectral Clustering (k=4)	**0.663**
Spectral Clustering (k=5)	0.332
Spectral Clustering (k=6)	0.332
Spectral Clustering (k=7)	0.324

We use the parameters obtained from the above table for generating the clusters based on different algorithms. The outputs of the self organizing maps

for different clustering techniques can be seen in figures6a, 6b and 6c. The self organizing maps show that the best results are obtained from spectral clustering with three clusters.



(a) dbscan (b) kmeans (c) spectral clustering

Fig. 6: Self organizing maps for different clustering algorithms on the VGAE Embedding

6.1 Clustering the entire embedding together

When we run spectral clustering over the entirety of the generated embeddings and thereafter only observe the topics, we make the following observations:

1. Two clusters contain no topics at all. (Highlighted by colours blue and red)
2. Cluster one contains most of the topics related to introductions, hi , hello and questions. (Highlighted by colour green)
3. Rest of the topics are all clustered in the second cluster. (Highlighted by colour orange)

We find no patterns in the authors when we observe the clusters, all the authors cluster together in a single cluster.

6.2 Clustering the topics and authors separately

We cluster the topics separately into four clusters based on the silhouette scores. The table 5 shows the clustering outputs of the topics in the iron march dataset. We observe that the clusters are more balanced (in terms of number of topics in each cluster) but the outputs show no legible distinction between similarly named topics. One distinction we notice is that topics with the same name but in different clusters are often in the different clusters because some influential members have a lot of posts on those topics.

We cluster the authors into four clusters based on silhouette score. We notice that the first cluster consists of authors who are involved in large number of posts.

The second cluster has only one author who talks a lot about MSN, skype. The third cluster consists of eight authors, who are clustered together because of their posts are in a particular languages and they have a lot of links on their posts. The last cluster seems to have no pattern between users, but some of the users talk about their location.

Table 5: Topics Clustered from VGAE

Cluster 0	Cluster 1	Cluster 2	Cluster 3
Our hosts	msn	forum structure	Introductions are important
Re: Nova Roma	Do you have Skype or AIM or MSN?	contacts	Revolt Against the Modern World
Introductions are important	Introductions are important	Pointless question	Real quick
Welcome back	Introductions are important	Question	Introductions are important
Could I have the banner without text and without the symbol?	Presidency	Introduction	Introductions are important
All you have to do is post a youtube link	Password	New computer?	book
I've a feeling me and you are going to get on well	Herp	Hi	What do you think of this?
Do you know what videos these clips were taken from?	Collaboration/Skype	Hey	Hello
Fascist archive	hey	Section Leader	Message on hotmail
Precise Location	oi!		Do you know what videos these clips were taken from?
Help	mushi mushi		hey
Hello	Welcome, comrade!		You...
Hello			I've a feeling we'll get on fine
			hey
			Thanks
			hi
			hey
			Introductions are important

7 Diachronic Analysis: Semantic Shifts

We generate embeddings for each time spell. Thereafter, we generate a diachronic component of the Graph Embeddings as proposed by Samtani et al [6]. This diachronic component gives us the semantic shifts in words over time, these shifts indicate how much the meaning of the word shifts as more and more words are added to the topics. These shifts can also be extended to behaviors of users or the topics themselves. This helps us capture the emerging changes in behavior over time. We align the embedding spaces by using orthogonal Procrustes matrix operations. Specifically, embedding spaces across time-spells are aligned while retaining cosine similarities by optimizing the following objective function:

$$R^{(t)} = argmin_{Q^T Q=I} ||W(t)Q - W(t+1||F$$

where $|| * ||F$ denotes the Frobenius norm.

These shifts indicate the maximum change in the generated embeddings over time. For words this means the co-occurrence or meaning shift. For authors it can mean change in association with different topics or other authors through posts and for topics it can mean change in the type of posts and (users) posting on the topic.

The words with the maximum semantic shifts over time stamps are in the list below.

Listing 1.2: Top twenty five words with the most shifts:

```
Maximum  Shifts = ['websites','swear','official','ad',
```

```
'von','liberal','coughlin','phone','everybody','pop',
'favouring','why','secondly','run','china','toi',
'reconcile','mix','every','movie','cordial','rockwell',
'foe','commandments','tune','negroid','caucasoid']
```

The topics with the maximum semantic shifts over time stamps are in the list below.

Listing 1.3: Top five topics with the most shifts:

```
Maximum Shifts = ['Message on hotmail','Introductions are important ',
'Revolt Against the Modern World ','Herp',
'Do you know what videos these clips were taken from?']
```

The authors with the maximum semantic shifts over time stamps are in the list below.

Listing 1.4: Top five authors with the most shifts:

```
Maximum Shifts = ['author_12','author_39','author_27','author_56',
'author_45']
```

Author 0 (Founder) has the third least change in semantic. This shows that his behaviour in spreading radical views and propaganda remains unchanged over time.

8 Adversarially Regularized Graph Autoencoder (ARGA)

Traditional GAEs and VGAEs have a probability that the gradient descent algorithm may not converge. Therefore, we use Adversarially Regularized Graph Autoencoder proposed by Pan S. et al. [5] which generalizes better on community detection (clustering) tasks. The obtained results are substantially better than VGAE.

8.1 Clustering the entire embedding together

When we run spectral clustering over the entirety of the generated embeddings and thereafter only observe the topics, we make the following observations:

1. Topics titled introductions are important occur in two different clusters.
2. Most of the topics with hey or hello as the title occur in the same cluster.
3. Topics started by more active users tend to belong to the same cluster.

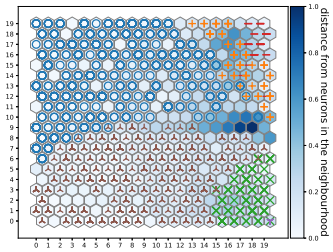


Fig. 7: Self organizing maps for clustering on ARGA embeddings

Table 6: Topics Clustered from ARGA

Cluster 0	Cluster 1	Cluster 2	Cluster 3	Cluster 4
Introductions are important	Welcome, comrade!	I've a feeling me and you are going to get on well	Fascist Archive	oi!
Our hosts	Section Leader	I've a feeling we'll get on fine	Message on hotmail	Pointless question
Re: Nova Roma				Question
msn				New computer?
Do you have Skype or AIM or MSN?				
Introductions are important				
Introductions are important				
Welcome back				
Could I have the banner without text and without the symbol?				
Revolt Against the Modern World				
All you have to do is post a youtube link				
Presidency				
Real quick				
Password				
forum structure				
Introductions are important				
book				
What do you think of this?				
Herp				
Hello				
Do you know what videos these clips were taken from?				
Do you know what videos these clips were taken from?				
Hello				
hey				
You...				
contacts				
hey				
Collaboration/Skype				
Thanks				
hey				
hi				
hey				
Introductions are important				
mushi mushi				
hi				
Precise Location				
Help				
Hello				

8.2 Clustering the topics and authors separately

We cluster the topics separately into five clusters based on the silhouette scores. The table 6 and figure 7 shows the clustering outputs of the topics in the iron march dataset. Some of the most important observations of the table are that all the introductory posts are in cluster 0. All clusters contain topics with similar content in them as observed in cluster 1 and 2.

We cluster the authors into two clusters based on silhouette score. We notice that the first cluster consists of authors who are involved in large number of posts. They show lot of activity, no author in this cluster has less than ten posts. The second cluster also has few authors with a lot of posts, but most of

these posts talk about things like federalism, conservatives, classical liberalism, and corporatism etc.

ARGA produces considerably lesser quantization error in the self organizing maps than VGAE based embeddings.

Note: We have implemented ARGA in the last two days, this is why we have not added the theory to this report, only the results. We are also actively looking at two other variants of VGAE which are optimized for better Community detection (VGAECD and VGAECD-OPT).

References

1. Bruna, J., Zaremba, W., Szlam, A., LeCun, Y.: Spectral networks and locally connected networks on graphs. arXiv preprint arXiv:1312.6203 (2013)
2. Henaff, M., Bruna, J., LeCun, Y.: Deep convolutional networks on graph-structured data (2015)
3. Kipf, T.N., Welling, M.: Variational graph auto-encoders. NIPS Workshop on Bayesian Deep Learning (2016)
4. Kohonen, T.: Self-organizing maps, vol. 30. Springer Science & Business Media (2012)
5. Pan, S., Hu, R., Long, G., Jiang, J., Yao, L., Zhang, C.: Adversarially regularized graph autoencoder for graph embedding. In: IJCAI. pp. 2609–2615 (2018)
6. Samtani, S., Zhu, H., Chen, H.: Proactively identifying emerging hacker threats from the dark web: A diachronic graph embedding framework (d-gef). ACM Trans. Priv. Secur. **23**(4) (Aug 2020). https://doi.org/10.1145/3409289, https://doi.org/10.1145/3409289
7. Wang, D., Cui, P., Zhu, W.: Structural deep network embedding. In: Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 1225–1234 (2016)
8. Yao, L., Mao, C., Luo, Y.: Graph convolutional networks for text classification (2018)