ITR Lab Week 5: Social Networks as Graphs and Graph Mining

Current Usage Investigation

Traditional machine learning methods such as node2vec focus more on dimensionality reduction that preserves the similarity between nodes. However, in some cases, lower dimensional representation of edges may also play an important role. For example, to predict the probability of link formation between two points in the future, mapping nodes into lower dimensional space may be useful, considering the sparse nature of the original correlation matrices of social networks. To solve this problem, Wang et al. presented an algorithm, edge2vec, that can be used to embed edges in social networks into low-dimensional space[2]. The researchers integrated a skip-gram model and a autoencoder in a neural network to preserve both global and local structural features related to edges. Similarly, the relationships between nodes are captured through neighborhood vectors. The test of this algorithm on six real-world datasets showed that it performed better than traditional graph representation methods considering tasks related to the prediction of edges. In social network analysis, the embedding of edges provides a useful tool for research about edge related topics, such as studies targeting the classification of edges in labeled networks (for example, trust and distrust).

Based on the traditional model of node2vec that emphasize the topological relationship and similarity between nodes, Zhou et al. proposed a "attribute-based" version of node2vec and its possible application in the study of co-authorship network[4]. Traditionally, the node attributes (this may include research interest or writing style in the case of co-authorship network analysis) are ignored in algorithms developed for the purpose of general network analysis, which is understandable as the importance, meaning, and measurement of these attributes varies from topic to topic. However, the authors proposed that these attributes can be highly informative, which should be taken into account when applying these algorithms to address specific research questions. Therefore, the authors leveraged the research interests of the authors as intrinsic node attributes. Latent Dirichlet Allocation (LDA) is used to generate topic distributions for each publication, which are then averaged on all publications of a particular author to represent his research interests. The normalized absolute distances among these distributions are used to represent the similarity of different authors' research interest, which is incorporated into the embedding such that transitions between nodes that are both topologically and semantically similar are prioritized. After embedding, the authors also conducted several analysis based on the embedding, including static and dynamic community detection, and evolution analysis of community center. From my perspective, more node attributes can be included into the embedding algorithms to increase its performance. Considering co-authorship network, potential attributes may include institution belonging and geolocation. Meanwhile, the dynamic community detection methods applied in this research can also be utilized to quantitatively measure the historical changes of semantic networks constructed on historical corpus.

Sen et al. introduced an interesting graph-based approach to enhance word embeddings[1]. Traditionally, word embedding models such as word2vec focus attention on co-occurrence in a small, local neighborhood. In this paper, a graph-based word embedding method is proposed that can capture both local and non-local co-occurrence. To achieve this goal, a graph is constructed, in which the weight of edges among vertices (words) is the product of local and non-local co-occurrence probabilities between words. However, in the node embedding part, the stratified sampling strategy prioritize direct neighbors over more distant neighbors, such that noise from distant neighbors is avoided. Apart from its methodology innovation, this paper also provides me with inspiration about the difference between word2vec and node2vec: word2vec tends to ignore co-occurrence in longer context and the 'co-occurrence chain' (for example, a co-occurs with b, b co-occurs with c, although a doesn't necessarily co-occurs with c, this 'co-occurrence chain' is still highly informative), while node2vec seems to be able to capture word's similarity based on their common relationship with certain nodes.

Data Application

This week I applied Node2vec algorithm to a small subset of the large dataset I used in the previous ITR lab, one that contains all posts on Chinese social media platform 'weibo' in January 2020 related to pandemic[3]. For this week, on the basis of the social media semantic network constructed last week, I removed the nodes that were not connected to other users and extracted the largest connected networks from them. In this largest connected community, I tried to apply Node2vec algorithms to explore the structure of the network.

At first, I reviewed the posts published by these users (as these users were all frequent users that published at least five posts every month, I was able to infer some of their characteristics from the tweets they posted, such as the topics they follow and so on), and manually labeled the users according to the topics they were interested in.

At first, I simply divided the users into two groups: those users who actually focused their posts on women's role in the pandemic and those who don't. The model's performance was rather satisfactory, with accuracy converging at about 0.97. However, the loss converged at around 2 and the plot of distribution after dimensionality reduction didn't show a clear disparity among the two clusters. The high accuracy of prediction may be caused by the uneven distribution of the two labels.

After that, I further divided the group with users not focusing on women's role in pandemic into five distinct groups, focusing on topic content and geometric distribution (I didn't use the geometric stamp here, the decision is made merely based on the post texts), with reference to the result of the community detection algorithm last week. The result is still not very satisfying, with loss converging at around 1.91 and accuracy converging at around 0.87. The plot of distributions of users also didn't reveal a clear boundary among different user groups.

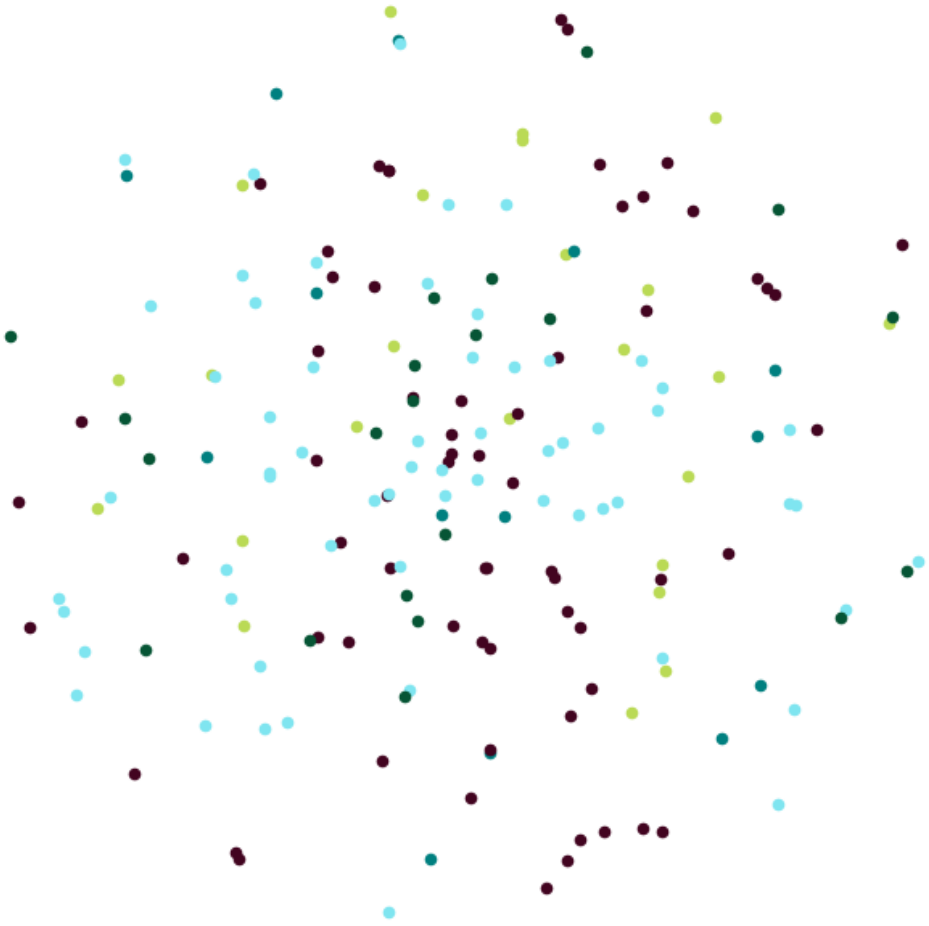

Figure 1: Node2Vec on Weibo Dataset

I also applied node embedding to further explore relationship between nodes (words) in the semantic networks of yearly People's Daily corpus. Yearly corpus of people daily are larger in size compared with the weibo network, and the embedding can well distinguish words related or not related to women portrayals. Figure 2 is the t-SNE representation of the node embedding model, in which Class 0 consists of words related to women portrayals and Class 1 consists of words not related. The result enabled a good classification of these two types of worlds utilizing only simple classification algorithms such as logistic regression. I also marked the misclassified word nodes with red color, which includes word such as 'iron girl' (铁姑娘) and 'women cadres' (女干部).

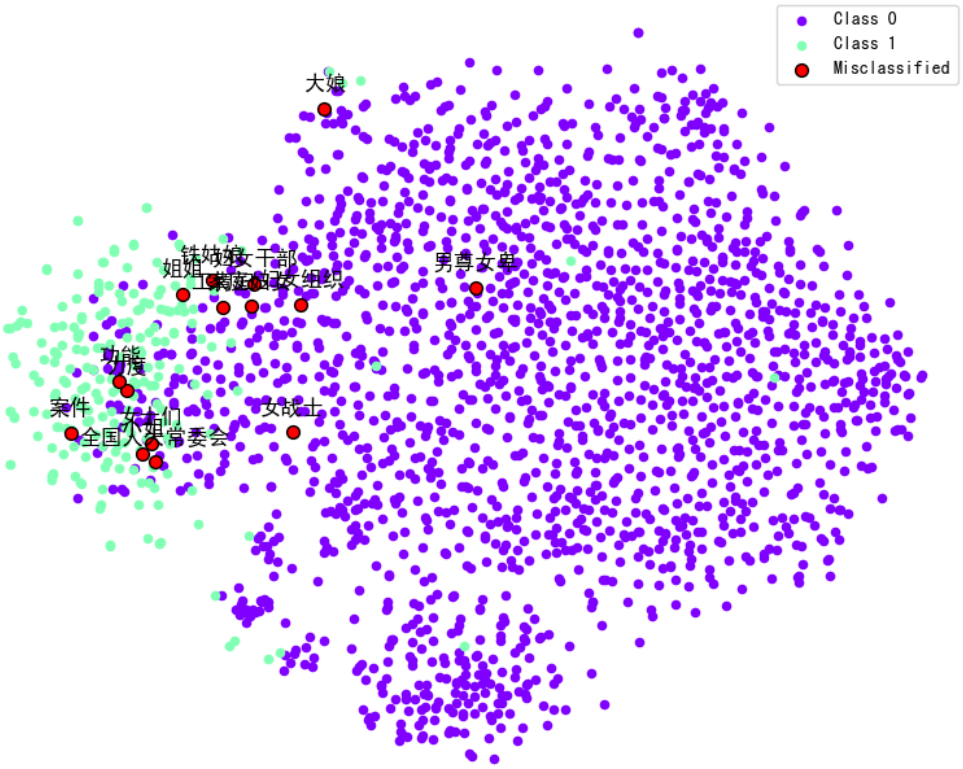

Figure 2: Node2Vec on People's Daily Dataset

Reflection

The results of the application of Node2Vec on two different corpus (social media and traditional media) provides empirical evidence for my research questions from different perspectives. First, considering social media, graphs based on cosine similarity between posts are much denser compared with those based on interactions (such as liking or forwarding). If the size of the network is too small, Node2Vec will not be able to capture the relationship and difference between different nodes (users). However, if we include considerable amount of vertices into the network, the time spent on Jaccard similarity calculation would grow exponentially, as the number of pairs needed to be calculated correlate exponentially with the post number in the corpus. In further analysis, I may consider using other algorithms to generate the user similarity network. For example, I may consider comparing the word sets of different users that are the combinations of all words used in their posts. In this way, the time complexity would reduce from $O(x^2y^2)$ to $O(x^2)$, where x is the number of the users we studied and y is the average number of posts published by a user. However, as we can see, in this case, construction of the model still has a exponential time complexity.

The analysis on the People's Daily corpus apparently provided more implication for our study. First, after embedding the semantic network, we can make use of classification or clustering models to distinguish words similar to a certain word or words group. This can be useful for the expansion of research dictionary. Secondly, the misclassification of those algorithm can also provide insights into the semantic space. For example, by observing the list of misclassified words generated by classification algorithm, we can find that a major portion of the misclassified words related closely with Chinese government's propaganda strategy at certain time periods, such as 'iron girl' (铁姑娘) and 'women cadres' (女干部) I mentioned above. These words are generally uniquely used in articles with

themes of women empowerment under the general background of rapid modernization and industrialization, thus having a rather different context compared with other, more commonly used words related to women portrayals. Analyzing such examples of misclassfication can reveal deeper patterns in how language evolves in response to ideological shifts (for example, these words shifting severely from the general cluster are expected to be highly correlated with certain policies and likely to vanish or emerge rapidly with the evolution of policy orientation, which can be further validated using casual inference techniques) and can provide qualitative evidence for the interplay between semantic representations and socio-political influences.

References

[1] Procheta Sen, Debasis Ganguly, and Gareth Jones. Word-Node2Vec: Improving Word Embedding with Document-Level Non-Local Word Co-occurrences. In *Proceedings of the 2019 Conference of the North*, pages 1041–1051, Minneapolis, Minnesota, 2019. Association for Computational Linguistics.

[2] Changping Wang, Chaokun Wang, Zheng Wang, Xiaojun Ye, and Philip S. Yu. Edge2vec: Edge-based Social Network Embedding. *ACM Transactions on Knowledge Discovery from Data*, 14(4):1–24, August 2020.

[3] Runbin Xie, Samuel Kai Wah Chu, Dickson Kak Wah Chiu, and Yangshu Wang. Exploring Public Response to COVID-19 on Weibo with LDA Topic Modeling and Sentiment Analysis. *Data and Information Management*, 5(1):86–99, January 2021.

[4] Tong Zhou, Rui Pan, Junfei Zhang, and Hansheng Wang. An attribute-based Node2Vec model for dynamic community detection on co-authorship network. *Computational Statistics*, March 2024.