Report for Task 2

20307130030 Shen Jianzhi

2023.5.28

1 Task

To establish a Weibo user profile using blog texts and social network, and then use that profile for sex, age and area prediction.

2 Data and PTMs

The data mainly used is the users' blog contents, and the followed users' ids from Weibo. The following links only covers approximately half users in the field of modeling. There are many promotional posts in the text field, which makes the text field repetitive and noisy. The text field contains irregular semantic units like hashtags, emojis and hyperlinks.

The data is split into training, testing and dev sets with respect to the users' ids.

The pre-trained word embedding model used are SGNE trained on Weibo corpus[1]. The text encoding model is ERNIE 3.0 nano[2].

3 Implementation

The model is a simple MLP with one hidden layer using dropout and L2 penalty. The feature of a user is extracted as below.

3.1 Text Embedding

There are two text embedding methods, of which one is token-level, and the other is blog-level.

Token-level text embedding is the weighted average of vectors of all the words a user posted, and the weights are derived as TF-IDF. This is an abstract of a user's wording habit, since TF-IDF highlights the words unique to the user's posts. TF-IDF also helps clear some noise, since the promotion posts are highly homogeneous in wording, which makes the TF-IDF of promotion-related words low.

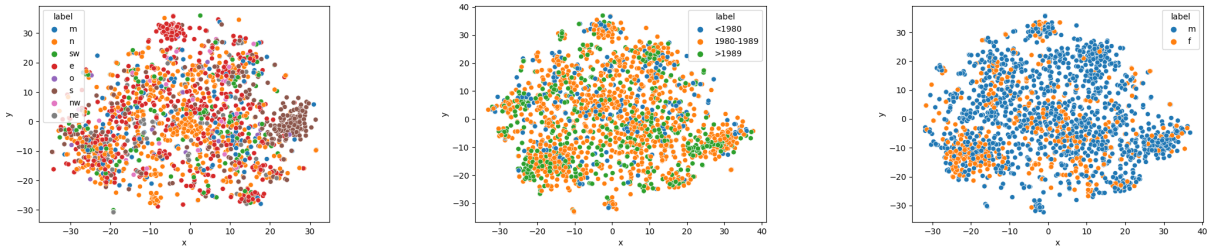
Blog-level text embedding is the mean of blogs encoded into vectors, and the encoding is done with ERNIE, since ERNIE is pre-trained on Chinese corpus. This method focuses on what kind of posts a user usually posts, since ERNIE extracts some semantic information for classification. One way to reduce the noise brought by promotional posts is to rank a user's blogs by its likes and reposts. However, in this dataset almost all blogs have no likes or reposts, so this method is not implemented.

3.2 Graph Embedding

The aim of graph embedding is to make the vectors close where nodes are similar. The idea is similar to that of Word2Vec and the corpus is derived from random traverse on the graph. DeepWalk[3] uses DFS as random traverse, LINE[4] uses BFS instead while taking into consideration the second order similarity. Node2Vec[5] is a combination of the two and SDNE[6] fix the prone-to-be-zero issue in sparse graphs. ProNE[7] is a high-efficiency sparse graph embedding method, which is based on sparse matrix decomposition and utilizes spectral propagation to further exploit higher order information in the graph.

Due to the scale of the external linkage information, here we use ProNE as graph embedding. Specifically, we extract the linkage information concerning second order neighbors of nodes in the user field, which cover approximately 20,000 nodes and 2,000,000 edges, and run ProNE on the subgraph.

As can be seen in Figure 1, some patterns has been captured by graph embedding. For example, on Area there are some clear clusters made up mainly by people from one particular area (on the right people from the South and on the top people from the East). The pattern on Sex is also noticeable, since some clusters are mainly male while some are mainly female, and a combined zone lays in the middle.



(a) Area (b) Age (c) Sex

Figure 1: t-SNE plots of ProNE on different classification tasks

3.3 Emoji Embedding

Graph embedding provides some applicable information for area and sex classification, but seems to help little in age classification. Emoji usage, on the other hand, may differs in different age groups. Therefore, we count the usage of 512 different emojis (in the dataset as phrases enclosed by square brackets) as emoji embedding.

However, as Figure 2 shows, the pattern does not clearly emerge. It is mainly due to the fact that the cluster of discrete and sparse data is hard to establish based on similarity.

4 Result

As shown in Table 2, although there is random fluctuation, that the blog-level embedding impedes the model is clear. Graph embedding improves greatly the classfication of area, and emoji embedding improves the sex and age classfication. However, when combining emoji embedding and graph embedding, the accuracy

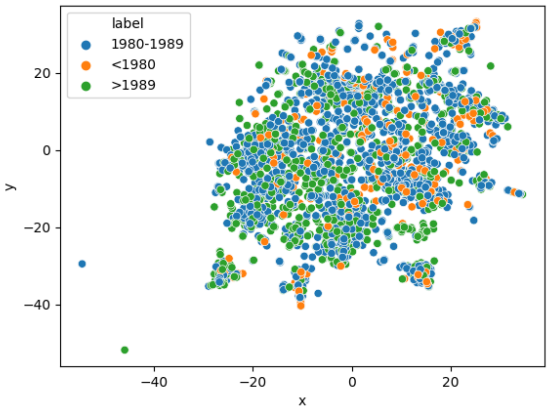


Figure 2: t-SNE plots of emoji embedding on Age

Batch size	64
Epoch	20 (Area) 5 (others)
Number of hidden units	100
Optimizer	Adam
Learning rate	1×10^{-3}
Weight decay	1×10^{-2}
Dropout rate	0.5

Table 1: Shared hyperparameters

Model	Sex	Age	Area
Major	78.15	57.42	27.09
Token	78.21	57.42	28.30
Token+Graph	78.23	57.42	**37.82**
Blog+Graph	78.15	57.42	27.09
Token+Blog+Graph	78.15	57.44	29.63
Emoji+Graph	80.32	60.24	33.14
Token+Graph+Emoji	**80.35**	**60.80**	32.34

Table 2: Percentage test accuracy of different embedding combinations on the three tasks

on Area decreases, which is possibly due to overfitting. Therefore, a compact choice of embedding when doing specific tasks is advisable.

5 Discussion

5.1 Use what information?

Using hashtags is an option, since bloggers intentionally choose them and they reflect what the blog mainly concerns. However, in this dataset, many hashtags are also contaminated by promotions, so this is not implemented.

Replacing the hyperlinks with the information about the linked website may help.

Image data like profile photos and posted images can further augment the model, with the help of CNN.

If more data, such as who likes or reposts a post, is available, we can build a GAT on user-post multi-relationship graph. Since many users do not post many posts related to themselves, but their interaction with others may provide some profile information.

5.2 Some upstream tasks

The noisy promotional posts are the main hurdle of text embedding, but they are actually informative for user profile. Therefore, we can build a classifier to tell which interest group the promotion belongs to, and then add the classifying result to the user's representation, such that the information, like that the user frequently uses an app, is embedded.

Another possible improvement is, when comment data is available, we can build a sentiment analysis model to predict the sentiment polarity of a user to a certain post, and then the user-post multi-relationship graph can be further augmented.

5.3 Why LLMs fail to work?

First, due to restriction in computing power, we only used the smallest ERNIE as a static encoder. If fine-tune is allowed, we may achieve better performance.

Second, without a task-specific pre-train procedure, the LLM can only provide a comprehensive semantic discrimination. However, in user profiling, we mainly want to know what aspect of everyday life a post is related to, like news, health or being emotional.

Third, the blog-level semantic information may be not that important. Because as the connectivity of people gets more and more tight, both homogeneity and polarization in people's thoughts are strengthened. What makes finer difference tend to be the nuance in wording habit or peculiar concepts. Therefore building a dictionary based or token-level model may be a better choice.

6 What I've Learned

1. Some graph embedding techniques.
2. Data mining in real-life data environment.

References

[1] S. Li, Z. Zhao, R. Hu, W. Li, T. Liu, and X. Du, "Analogical reasoning on chinese morphological and semantic relations," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 138–143, Association for Computational Linguistics, 2018.

[2] Y. Sun, S. Wang, S. Feng, S. Ding, C. Pang, J. Shang, J. Liu, X. Chen, Y. Zhao, Y. Lu, W. Liu, Z. Wu, W. Gong, J. Liang, Z. Shang, P. Sun, W. Liu, X. Ouyang, D. Yu, H. Tian, H. Wu, and H. Wang, "Ernie 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation," 2021.

[3] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, ACM, aug 2014.

[4] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "LINE," in *Proceedings of the 24th International Conference on World Wide Web*, International World Wide Web Conferences Steering Committee, may 2015.

[5] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," 2016.

[6] D. Wang, P. Cui, and W. Zhu, "Structural deep network embedding," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, (New York, NY, USA), p. 1225–1234, Association for Computing Machinery, 2016.

[7] J. Zhang, Y. Dong, Y. Wang, J. Tang, and M. Ding, "Prone: Fast and scalable network representation learning," in *Proceedings of the 28th International Joint Conference on Artificial Intelligence*, IJCAI'19, p. 4278–4284, AAAI Press, 2019.