2022 IEEE International Conference on
Systems, Man, and Cybernetics (SMC)

PerKG: A Personality Knowledge Graph for Personality Analysis

Yangfu Zhu, Zhanming Guan, Siqi Wei, and Bin Wu*

2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC) | 978-1-6654-5258-8/22/$31.00 ©2022 IEEE | DOI: 10.1109/SMC53654.2022.9945152

***Abstract*—With the blossoming of online social networks (OSN), personality analysis based on OSN texts has gained much research attention in recent years. The previous methods mainly focus on human-designed features extracted through psychological dictionaries or semantic features extracted through language models. However, the shallow statistics features can not fully convey the personality information and the language models can not capture enough psychological background knowledge. Besides, the lack of large labeled datasets has been a serious obstacle impending further research. To tackle these problems, we propose a personality analysis model, namely PerKG, which combines personality knowledge graph and heterogeneous graph representation learning to exploit external knowledge from psycholinguistics and learn the group-level information to predict users' personalities accurately. Specifically, we construct a personality knowledge graph based on existing psycholinguistics knowledge. And then, for each user, we align the user information with the knowledge graph to obtain the personality heterogeneous graph. Finally, the personality vector of each entity node is learned for prediction by designing a walk strategy on the personality heterogeneous graph. Detailed experimentation shows that our proposed PerKG architecture can effectively improve the performance and alleviate the label sparsity problem of personality analysis.**

***Index Terms*—knowledge graph, network representation learning, personality analysis, online social network**

I. Introduction

The personality traits are defined as the characteristic sets of behaviors, cognitions, and emotional patterns that evolve from biological and environmental factors [1]. Traditional questionnaire-based approaches to personality detection are time-consuming and laborious [2]. Nowadays, people publish their daily moods, activities, and opinions on social networks. Those posts could convey a lot about people's personalities. Many studies show that the personality analysis on online social networks (ONS) has a wide range of practical applications, such as recommendation systems [3], career interviews [4] and politics participate analysis [5]. Hence, an accurate grasp of users' personalities is significant for helping to get a deeper insight into their emotions, preferences, and behaviors.

The current personality analysis based on OSN texts mainly relies on psychological lexicon [6]–[8] or applies deep neural model [9]–[12]. The common psychological lexicon-based methods extract word frequency statistics from OSN user texts through psychological dictionaries and then use machine learning algorithms to predict the personality scores. Studies have shown that those features are helpful for personality analysis. However, it is hard for the model to learn the inherent connection between linguistic information and personality.

To avoid feature engineering, deep neural networks have been employed for personality analysis. Although the neural model made some progress due to learning abundant semantic information from the corpus, they fail to fully exploit external psycholinguistics knowledge, which could help a deeper understanding of personality psychology. For example, psychologists found that people with *"neuroticism"* personality usually treat problems from a pessimistic perspective (Emotion), and are too sensitive to surroundings and overstressed (Mental-sate). At the same time, being in contact with a *"agreeableness"* people will give you a comfortable feeling. He/She will care about you very much and is willing to give assistance when you are in trouble (Team-identity) [13]. In addition, annotation of personality data requires a high level of psychological skills, and it is impossible to carry out large-scale personality data annotation, which makes the sparsity of labeled data a serious problem that we have to solve.

To overcome the above limitations, we construct a personality knowledge graph based on existing psycholinguistics knowledge, and after aligning user information, personality analysis is achieved through personality heterogeneous graph representation. Specifically, the PerKG contains different entities (words, personalities, language styles) and different relationships between those entities. Then, for each user, we align the user information with the knowledge graph semantics to obtain the personality heterogeneous graph. Finally, the personality vector of each entity node is learned for prediction by designing walk strategy on the personality heterogeneous graph. Extensive experiments demonstrate that PerKG significantly outperforms state-of-the-art models. In summary, our main contributions include:

- To the best of our knowledge, this is the first effort to construct a personality knowledge graph (PerKG), providing a new perspective for exploiting the psycholinguistics knowledge for personality analysis.
- Combining the PerKG and user language styles, we designed a modified walk strategy to embed the personality heterogeneous graph and transform personality analysis into a semi-supervised learning task. Our model could generate personality vectors that contain abundant personality information for users without labels.
- Extensive experiments on 3 benchmark datasets demonstrate that our method makes a good performance in comparison with 7 baselines. Moreover, even if the label rate is low, our model could maintain a good performance, which is meant to alleviate the label sparsity problem.

This work is supported by the NSFC-General Technology Basic Research Joint Funds under Grant (U1936220), the National Natural Science Foundation of China under Grant (61972047), the National Key Research and Development Program of China (2018YFC0831500) and the BUPT Excellent Ph.D. Students Foundation (CX2022219). Y. Zhu, Z. Guan, Q. Wei, and B. Wu are with the Beijing Key Laboratory of Intelligence Telecommunication Software and Multimedia, Beijing University of Posts and Telecommunications, Beijing 100876, PR China *(e-mail: zhuyangfu, gzm2062, Weisiqi, wubin@bupt.edu.cn;) (Corresponding author: Bin Wu).*

II. Related Work

A. Psychological Lexicon-based

Personality analysis has gained much research attention in the past. Linguistic Inquiry and Word Count (LIWC) [14] is a word-based text analysis application, which counts the word frequency of different psychologically meaningful categories. Mairesse used LIWC and MRC [15] to extract different textual features and made a comparison of the effectiveness of those features [6]. Golbeck collected 279 users' Twitter posts and their Big Five scores through a Twitter application and gathered 161 statistics features to predict their personality scores [7]. However, as LIWC is a predefined dictionary that can't catch each word and phrase in the social media corpus, it may lose some potentially linguistic information.

B. Neural Language Model-based

With the blossoming of deep neural networks, many neural language models are making better performances and they are also applied in the field of personality analysis. Sun et al. proposed a complicated model including a sentence encoder and a document encoder to extract semantics features for personality prediction [10]. Kampman et al. used a deep learning method to detect one's personality from audio, text, and video [16]. In recent years, pre-training word embeddings is widely applied [17]. Researchers try to let the model learn useful features from these vectorized representations to detect personality [9], [18]. Network representation learning (NRL) has also been widely accepted due to its ability to learn graph structure information. Adawalk [11] is the first paper to apply NRL methods to personality analysis. The PersonalityGCN [12] modified TextGCN [19] to model the whole user text information corpus as a heterogeneous graph for personality analysis. However, these studies only consider the semantics or simply concatenate the statistical features, without considering domain knowledge information.

C. Domain Knowledge Graph

In recent years, the knowledge graph has made huge progress. Due to the ability to make use of external knowledge and the advantages of interpretability, knowledge graphs have been applied in many scenarios, such as recommendation system [20], fake news detection [21], and misinformation detection in healthcare [22]. Dun et al. proposed a knowledge-aware attention network that incorporates external knowledge from a knowledge graph for fake news detection [21]. Cui et al. proposed a medical domain knowledge-guided graph attention network for misinformation detection in healthcare [22]. Although knowledge graph has been proved effective in many fields, it has not been applied to personality analysis.

III. Methodology

In this section, we first elaborate the construction of our personality knowledge graph, then present our proposed user semantic alignment and personality heterogeneous graph embedding module.

A. Personality Knowledge Graph

We construct the personality knowledge graph based on the influences and connections between various language styles and personalities which are proposed in the professional psychology study *The Secret Life of Pronouns* [13] as shown in Fig. 1. Since we are taking a first look into the feasibility of applying knowledge graph on personality analysis, we don't focus too much on making use of comprehensive psycholinguistics knowledge. We summarize the connections between language styles and personalities in the book and represent those connections with 10 types of entities and 7 relationships. Among those entity types, $word$ type contains categories of words that could reflect users' language styles, and the other 9 types of entities describe users' personalities from different aspects. The details about those entities are as following:

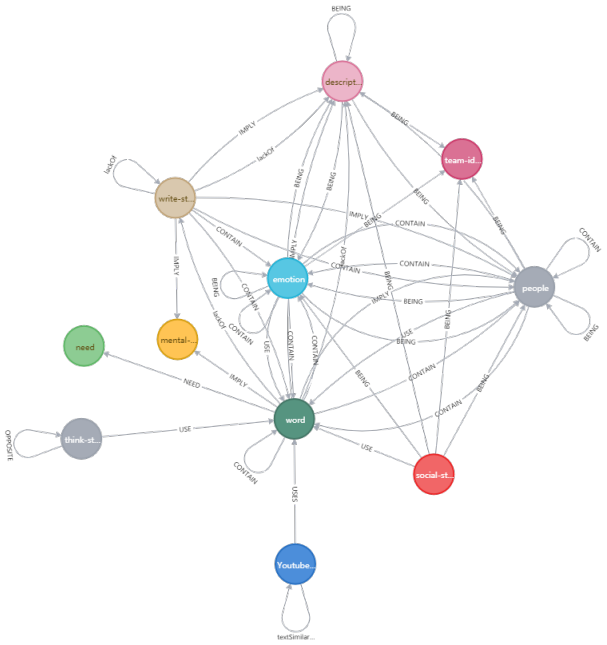


Fig. 1. Schematic Diagram of Personality Knowledge Graph

- $word$: It consists of 45 categories of functional words and we construct 3 entities $c+$, c, $c-$ for each category c individually, representing the use frequency of this category of words. We construct the dictionary based on LIWC psychology dictionary. For those words that are missing LIWC dictionary such as big words, we use WordNet [23] and Wiki [24] to extend the categories according to their definition.
- $think-style$: It contains 6 entities of thinking styles, including classification vs dynamic, simple vs complicated, casual vs rigorous.
- $write-style$: It contains 3 entities of users' writing styles, including narrative, analytical and formal.
- $description$: It contains 26 entities of descriptions of characteristics such as arrogance, artistic, self attention and so on.
- $emotion$: It contains positive emotion and negative emotion and negative emotion could be further divided into sadness and angry.
- $mental-state$: It contains 2 entities, mental-healthy and mental-unhealthy.
- $need$: It contains three major needs of people: need for power, need for belonging and need for achievement.
- $people$: It contains several kinds of people in the world, such as male vs female, young vs old and some special kinds including patients with depression and politicians.

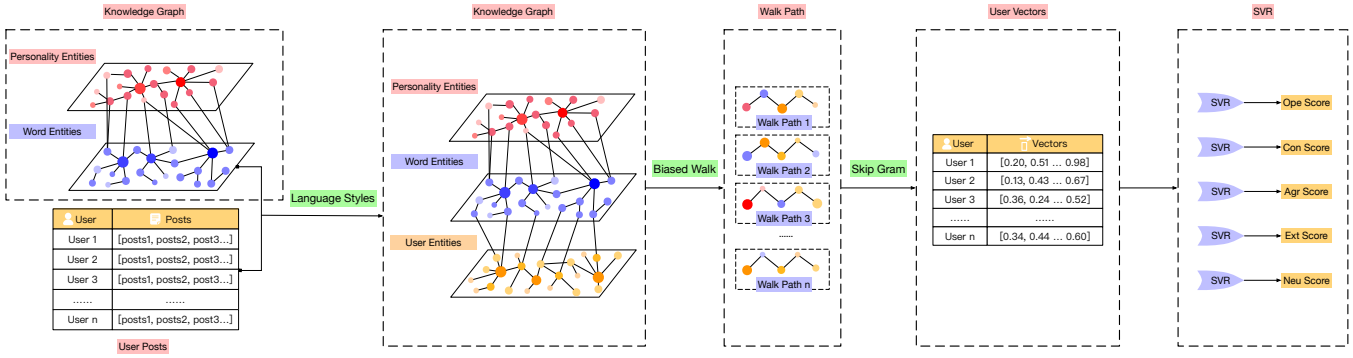


Fig. 2. Overall Process of PerKG

- $social - status$: It contains 2 entities of social status, high status and low status.
- $team - identity$: It contains 3 entities of the sense of identity that people hold in a team, including discrimination, prejudice and modeled.

Besides the above 10 types of entities, we also set up 7 relationships including $imply$, $contain$, $being$, $need$, $opposite$, use and $lackof$. We connect all those nodes with different relationships and make the knowledge well displayed in our graph.

B. User Semantic Alignment

To perform the personality analysis, we need to add users into the graph. For each dataset, we calculate the average word frequency f_i for each category i of words in our dictionary. Then for each user u, we calculate the word frequency f_{ui} for each word category i. User u's word frequency of category i is determined to be low if $f_{ui} < f_i * 0.9$, to be high if $f_{ui} > f_i * 1.1$ and to be medium for the other conditions. Then we connect user entity u with the entity of this word category i which is chosen from $i+$, i, $i-$ according to the word frequency. Based on the word usage, we align users with personality knowledge graph and make each user connect with different personality entities indirectly through the $word$ type entities and thus contain personality information. The schematic diagram of the knowledge graph is as shown in Fig. 1. To make the graph contain semantic information, we also added a $similarTo$ relationship between users. We use $tfidf$ [25] to calculate the similarities between every two users, if the similarity is larger than a threshold, we connect those two users with $similarTo$ relationship. For the selection of the threshold, a too high threshold will decrease the number of relationships and make the walk paths produced in the subsequent walk strategy too simple, whereas a too low threshold would make it hard to form an effective group and hard to learn the group-level information. We carry out experiments and find out that retaining 25% similarity relationships could achieve the best performance and that is exactly how we select the threshold.

C. Personality Heterogeneous Graph Embedding

Through user node integration, we construct a complete personality heterogeneous graph $G(U, E)$ based on the dataset. Here U is the all type node set, E is the edge set. Our problem can be treated as multi-regression that learnning a user node embedding matrix $\phi \in \mathbb{R}^{|u| \times d}$ for predicting personality score. As shown in Fig. 2, we design a heterogeneous network representation learning method to refine user node embedding. it can not only learn the node-level features of entities in the graph but also learn the group-level information and thus make use of the mutual influences within the groups.

Specifically, we design a new heterogeneous graph walk strategy. Given an ongoing path, assuming the current node is u and the previous node is v, the probability of next node x being selected is defined as follows:

$$p(x|u,v) = \begin{cases} \frac{\alpha(x,u,v)}{Z}, & \text{if } edge(u,x) \in E \\ 0, & \text{else} \end{cases} \quad (1)$$

where E is the edge set of our graph, Z is the normalization constant and α is a control coefficient. Assuming $sim(e_1, e_2)$ is the text simialrity between texts of user $e1$ and user $e2$ when $e1$ and $e2$ are both $user$ type nodes. If $e1$ and $e2$ are not both $user$ type nodes, we set $sim(e_1, e_2)$ as zero. α is defined as follows:

$$\alpha(x,u,v) = \begin{cases} \frac{1+sim(v,u)}{p}, & \text{if } d = 0 \\ 1 + sim(v,u), & \text{if } d = 1 \\ \frac{1+sim(v,u)}{q}, & \text{if } d = 2 \end{cases} \quad (2)$$

where d is the distance between the next node x and the previous node v. As shown in equation 2, while facing a selection between user-type nodes and other-type nodes, we introduce text similarity into the walk strategy to make it more likely to select user-type nodes, and for different user nodes, the strategy is more likely to choose a user who has a greater text similarity with the current node. (p, q) are hyperparameters used to adjust the walk strategy to depth first search (DFS) or breadth first search (BFS). It should be mentioned that when it is walking among personality-type and word-type nodes, we introduce a new pair of hyperparameters (pf, qf) to make the walk strategy more likely to select a new word-type node while the current node is a personality-type node and then make path go back to user-type nodes. Our walk strategy could enable word-type nodes and personality-type nodes in the graph to control the generation of walk paths so that users with similar language styles, similar characteristics, and similar text information could appear in the same path with greater probability and finally generated paths of nodes.

Then we use skip-gram [26] to embed those paths of nodes into vectors. The skip-gram algorithm maximizes the probability of each node and its context nodes in a generated path, the function is:

$$L = \sum_{u \in C} \prod_{context(\mathrm{r})} P(N(u)|u) \tag{3}$$

where C is the nodes corpus and given a node u, $N(u)$ represents its context nodes, r is context size. Basically, the personality heterogeneous graph embedding algorithm and biased walk algorithm shown in Algorithm1 1,2.

Algorithm 1 PerKG Emmbeding Algorithm

Input:
$G(U, E)$ personality heterogeneous graph, context size r, walk length l, walks per node μ, embedding size b.
Output:
user node embedding matrix $\phi \in \mathbb{R}^{|u| \times d}$

initialize: walks to Empty;
for $iter = 1$ to μ **do**
  for each node u **do**
    walk = PerKG Walk $(G(U, E), u, l)$from Algorithm (2);
    append walk to walks;
  end for
end for
$\phi \in SkipGram(r, b, walks)$ from Eq (3);
return ϕ

Algorithm 2 PerKG Walk Algorithm

Input:
$G(U, E)$ personality heterogeneous graph, context size r, walk length l, walks per node μ, embedding size b. traversal parameters p, q, pf, qf node neighbors $N(u)$.
Output:
a random walk from graph $G(U, E)$

initialize:$walk = [u]$;
choose a neighbors node $u \in N(u)$;
for $walk\_len$ =1 to l **do**
  $cur = walk[-1]$;
  $prev = walk[-2]$;
  for each node $u \in N(cur)$ **do**
    if $u \in N(prev)$ **then**
      $\alpha(x, u, v) = \frac{1+sim(v,u)}{p}$;
    else if u = $prev$ **then**
      $\alpha(x, u, v) = 1 + sim(v, u)$;
    else
      $\alpha(x, u, v) = \frac{1+sim(v,u)}{q}$;
    end if
  end for
  calculate the next node u transition probability from Eq (1);
  append u to $walk$;
end for
return $walk$

D. Model Training

We use SVR to predict the Big Five personality scores with users' vectors and use Mean Absolute Error (MAE) as our evaluation metrics, which could be calculated as follows:

$$MAE = \frac{1}{n} \sum_{i=1}^{n} \left| s^i_{pred} - s^i_{true} \right| \tag{4}$$

where n denotes the number of instances, s^i_{pred} is the predicted value and s^i_{true} is the truth. Considering our datasets are relatively small, we carry out downstream model training and testing experiments with 5-fold cross-validation and the average MAE is recorded.

IV. Experiments

In this section, We first introduce the datasets and 7 baseline methods that we compared with, then present our main results.

A. Big Five Personality and Datasets

The Big Five model of personality is a generally accepted and influential metric in psychology for characterizing and measuring personality traits. Big Five formalizes personality along with five bipolar personality traits: Openness, Conscientiousness, Extroversion, Agreeableness, and Neuroticism, each trait is represented by a score. We select 3 public datasets to carry out the experiments. The details of the datasets are as follows:

- MyPersonality[1]: MyPersonality project is a Facebook App that allowed users to participate in psychological research by filling in a personality questionnaire. It contains 114,958 users' posts and their Big Five scores.
- Youtube [27]: This dataset consists of Youtube speech transcriptions of 404 users and their Big Five scores.
- PAN [2]: This dataset comes from data science competition PAN2015, which includes 4 languages dataset and we choose the English dataset which contains 294 users' posts and their Big Five scores.

B. Baseline Methods

To prove the effectiveness and superiority of our method from different aspects, we compared our method with 7 popular methods on three datasets.

By artifical features:

- Mairesse [6]: This method used LIWC features to analyze personality from texts, as mentioned in Section 2.

By network representation learning:

- AdaWalk [11]: This is the first method that introduces NRL into personality analysis. It uses tfidf to calculate the texts similarity between users and construct the network, then a modified walk strategy designed to learn the nodes embeddings.
- node2vec [28]: This method uses a biased walk to generate paths and uses word2vec to embed nodes into vectors. It introduces 2 hyperparameters to control the walk strategy between DFS and BFS. The construction of the network is as same as adawalk.

By supervised deep learning methods:

[1]http://mypersonality.org
[2]https://pan.webis.de/clef15/pan15-web/author-profiling.html

- 2CLSTM [10]: This method constructs a sentence encoder using bi-lstm and a document encoder using CNN. Then it combines those features with users' LIWC features as the input of the downstream classifier.
- BERT [29]: BERT is a powerful pretrain language model with great performance on many NLP tasks. We follow [9] and use bert-base-uncased version pretrained model and fine tune it in downstream personlality prediction tasks.

By knowledge graph embedding methods:

- TransE [30]:TransE encodes both entities and relations as vectors in the same space. Considering triples (h, r, t), in which h, r and t represents the head, the relation, and the tail respectively. TransE trains the model on all the triples to make $\mathbf{h} + \mathbf{r} \approx \mathbf{t}$ when (h, r, t) holds, where $\mathbf{h}$, $\mathbf{r}$, $\mathbf{t}$ are the corresponding vector of h, r and t.
- TransD [31]:TransD is TransE's extensions by incorporating more sophisticated translation mechanisms to model complex relations.

TABLE I
MAE FOR PERSONALITY PREDICTION

Dataset	Methods	OPE	CON	EXT	ARG	NEU
Youtube	**PerKG**	**0.4691**	**0.4464**	**0.6857**	**0.5121**	**0.5031**
	AdaWalk	0.5772	0.5581	0.7329	0.6424	0.6033
	node2vec	0.5853	0.5674	0.7566	0.6883	0.6108
	BERT	0.6213	0.6063	0.7992	0.7613	0.6105
	2CLSTM	0.6431	0.6118	0.8066	0.7755	0.6438
	Mairesse	0.5503	0.5652	0.7469	0.6218	0.6257
	TransD	0.5046	0.4743	0.7105	0.5268	0.5109
	TransE	0.5172	0.4726	0.7083	0.5796	0.5458
MyPersonality	**PerKG**	**0.3624**	**0.5091**	**0.5644**	**0.4124**	**0.5259**
	AdaWalk	0.4583	0.5991	0.6112	0.5031	0.6405
	node2vec	0.4748	0.6182	0.6246	0.5308	0.6518
	BERT	0.4893	0.6317	0.6742	0.5859	0.6532
	2CLSTM	0.4936	0.6428	0.6963	0.5834	0.6809
	Mairesse	0.4677	0.6117	0.6289	0.5192	0.6401
	TransD	0.4091	0.5313	0.5728	0.4772	0.5539
	TransE	0.4052	0.5574	0.5717	0.4659	0.5782
PAN	**PerKG**	**0.0963**	**0.0906**	**0.0891**	**0.0931**	**0.1245**
	AdaWalk	0.1188	0.1054	0.1126	0.1083	0.1742
	node2vec	0.1191	0.1109	0.1188	0.1053	0.1849
	BERT	0.1239	0.1242	0.1211	0.1094	0.1395
	2CLSTM	0.1287	0.1213	0.1248	0.1162	0.1433
	Mairesse	0.1302	0.1246	0.1373	0.1264	0.1805
	TransD	0.1003	0.0997	0.0974	0.1025	0.1326
	TransE	0.1109	0.0982	0.1108	0.1011	0.1387

C. Results Analysis

As shown in Table I, PerKG outperforms other methods in most traits. We believe the reasons are two fold: (1) Our model leverages personality heterogeneous graphs to learn better user representations, which injected some psychological clues into personality analysis. (2) User correlations are well captured through semi-supervised learning on graphs, which reduces the risk of overfitting on a small training set.

Artificial features don't perform well in most traits, except that the results of Mariesse on Youtube and MyPersonality are competitive. We think it is because the length of the posts of those two datasets is relatively longer than those in PAN dataset, and thus helps LIWC extract more useful information.

Deep learning methods don't perform well as expected, even BERT, which makes perfect performance on a lot of NLP tasks. On the one hand, those deep learning methods need a huge amount of data to train the model, our datasets are small for them to fit the data. On the other hand, it might take only a shallow understanding to solve problems such as text classification but when it comes to the analysis of personality, the model needs to understand the texts from a psychological professional perspective.

Obviously, NRL methods make competitive performances. Besides, the embedding processes of those methods don't need any labeled data and this makes those methods more applicable. Nevertheless, those methods construct the network simply based on their texts and make no use of the external knowledge, which limits the performance to some extent.

The results of TransD and TransE show that although their performances outperform most methods, they are not as good as our method. The reason might be the small size of our knowledge graph, especially the personality part of our graph which is too simple and doesn't contain enough psychological information and it couldn't generate enough triples while training. This problem could be solved by adding more relative psychological knowledge into our knowledge graph to make it larger and more professional. The other reason could be that those methods couldn't learn the group-level information and the mutual influences between users within a group, which are important for personality prediction.

D. Parameter Sensitivity

On three datasets, we manually adjust the hyperparameters and select the combination with the lowest MAE value. We compare different walk lengths l, and context sizes r. The results are shown in Fig. 3. On Youtube, MyPersonality, and PAN datasets, walk lengths l are 80, 70, and 70. The context sizes r are 2, 8, and 6, respectively. The hyperparameters (p, q) and (pf, qf) are set up as [1, 1] and [2, 0.25], the graph embedding size $d = 200$. the kernel of SVR is set to be radial. As for the other methods, we use default hyperparameters as in the related papers.

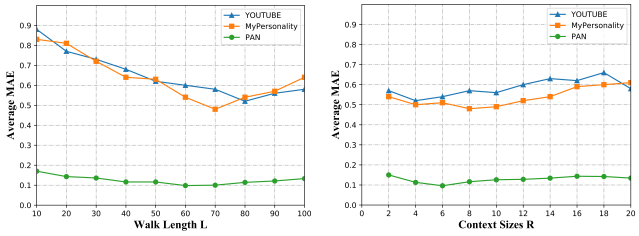
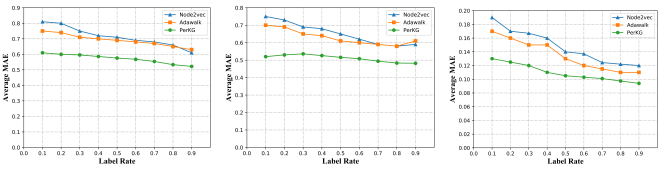


Fig. 3. Average MAE results with different walk lengths and context sizes.

E. Impact of Number of Training Samples

In order to verify that our method could learn the psycholinguistics knowledge from our personality knowledge graph, we compare our method with other NRL methods on the downstream prediction with the label rate decreasing from 0.9 to 0.1, we select NRL methods as baselines because those methods are more competitive and they are less reliable on labels than deep learning methods. The result curves are shown in Fig. 4. We give the average MAE of 5 traits for each method and each dataset. As shown in the Fig. 4, the performance of our method outperforms other methods at every label rate, and the superiority of our method is even larger when the label rate is getting down. It is important that the performance of our method doesn't decrease much while the label rate decreases from 0.9 to 0.1, this means the user vectors generated from our



(a) Youtube (b) MyPerosonality (c) PAN

Fig. 4. Average MAE results with different label rate.

method have learned abundant personality information from our knowledge graph and group-level information during the embedding process and don't rely much on the downstream training process. This result is exciting because it proves that it is truly effective to introduce a knowledge graph into personality analysis. In the future, as long as we add more psychological knowledge to expand our knowledge graph and fix the way appropriately to add users to it, the method could learn more personality information, which makes our method more effective and more applicable.

V. Conclusion

In this paper, we construct a personality knowledge graph based on existing psycholinguistics knowledge and combine the user information to achieve personality analysis by heterogeneous network representation. The experiment results have proved that it is truly effective and helpful to make use of external psychological knowledge and to alleviate the label sparsity problem. In the future, we will keep on expanding our personality knowledge graph by adding more psychological knowledge and we believe this could help the embedding vectors contain more personality information.

References

[1] S. Štajner and S. Yenikent, "A survey of automatic personality detection from texts," in *Proceedings of the 28th International Conference on Computational Linguistics*, 2020, pp. 6284–6295.

[2] R. S. Camati, A. A. Scaduto, and F. Enembreck, "Using the projective themathic apperception test for automatic personality recognition in texts," in *2021 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2021, pp. 78–85.

[3] S. Dhelim, N. Aung, M. A. Bouras, H. Ning, and E. Cambria, "A survey on personality-aware recommendation systems," *Artificial Intelligence Review*, vol. 55, no. 3, pp. 2409–2454, 2022.

[4] M. L. Kern, P. X. McCarthy, D. Chakrabarty, and M.-A. Rizoiu, "Social media-predicted personality traits and values can help match people to their ideal jobs," *Proceedings of the National Academy of Sciences*, vol. 116, no. 52, pp. 26 459–26 464, 2019.

[5] J. Usher and P. Dondio, "Brexit: Psychometric profiling the political salubrious through machine learning: Predicting personality traits of boris johnson through twitter political text," in *Proceedings of the 10th International Conference on Web Intelligence, Mining and Semantics*, 2020, pp. 178–183.

[6] F. Mairesse, M. A. Walker, M. R. Mehl, and R. K. Moore, "Using linguistic cues for the automatic recognition of personality in conversation and text," *Journal of artificial intelligence research*, vol. 30, pp. 457–500, 2007.

[7] J. Golbeck, C. Robles, and K. Turner, "Predicting personality with social media," in *CHI'11 extended abstracts on human factors in computing systems*, 2011, pp. 253–262.

[8] Y. Mehta, S. Fatehi, A. Kazameini, C. Stachl, E. Cambria, and S. Eetemadi, "Bottom-up and top-down: Predicting personality with psycholinguistic and language model features," in *2020 IEEE International Conference on Data Mining (ICDM)*. IEEE, 2020, pp. 1184–1189.

[9] Z. Ren, Q. Shen, X. Diao, and H. Xu, "A sentiment-aware deep learning approach for personality detection from text," *Information Processing & Management*, vol. 58, no. 3, p. 102532, 2021.

[10] D. Xue, L. Wu, Z. Hong, S. Guo, L. Gao, Z. Wu, X. Zhong, and J. Sun, "Deep learning-based personality recognition from text posts of online social networks," *Applied Intelligence*, vol. 48, no. 11, pp. 4232–4246, 2018.

[11] X. Sun, B. Liu, Q. Meng, J. Cao, J. Luo, and H. Yin, "Group-level personality detection based on text generated networks," *World Wide Web*, pp. 1–20, 2019.

[12] Z. Wang, C.-H. Wu, Q.-B. Li, B. Yan, and K.-F. Zheng, "Encoding text information with graph convolutional networks for personality recognition," *Applied Sciences*, vol. 10, no. 12, p. 4081, 2020.

[13] J. W. Pennebaker, "The secret life of pronouns," *New Scientist*, vol. 211, no. 2828, pp. 42–45, 2011.

[14] J. W. Pennebaker, R. L. Boyd, K. Jordan, and K. Blackburn, "The development and psychometric properties of liwc2015," Tech. Rep., 2015.

[15] M. Coltheart, "The mrc psycholinguistic database," *The Quarterly Journal of Experimental Psychology Section A*, vol. 33, no. 4, pp. 497–505, 1981.

[16] O. Kampman, E. J. Barezi, D. Bertero, and P. Fung, "Investigating audio, video, and text fusion methods for end-to-end automatic personality prediction," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, 2018, pp. 606–611.

[17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[18] P.-H. Arnoux, A. Xu, N. Boyette, J. Mahmud, R. Akkiraju, and V. Sinha, "25 tweets to know you: A new model to predict personality with social media," in *Proceedings of the International AAAI Conference on Web and Social Media*, vol. 11, no. 1, 2017.

[19] L. Yao, C. Mao, and Y. Luo, "Graph convolutional networks for text classification," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, 2019, pp. 7370–7377.

[20] X. Wang, T. Huang, D. Wang, Y. Yuan, Z. Liu, X. He, and T.-S. Chua, "Learning intents behind interactions with knowledge graph for recommendation," in *Proceedings of the Web Conference 2021*, 2021, pp. 878–887.

[21] Y. Dun, K. Tu, C. Chen, C. Hou, and X. Yuan, "Kan: Knowledge-aware attention network for fake news detection," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 1, 2021, pp. 81–89.

[22] L. Cui, H. Seo, M. Tabar, F. Ma, S. Wang, and D. Lee, "Deterrent: Knowledge guided graph attention network for detecting healthcare misinformation," in *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2020, pp. 492–502.

[23] C. Fellbaum, "Wordnet," *The encyclopedia of applied linguistics*, 2012.

[24] B. Leuf and W. Cunningham, *The Wiki way: quick collaboration on the Web*. Addison-Wesley Longman Publishing Co., Inc., 2001.

[25] J. Ramos *et al.*, "Using tf-idf to determine word relevance in document queries," in *Proceedings of the first instructional conference on machine learning*, vol. 242. Piscataway, NJ, 2003, pp. 133–142.

[26] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[27] J.-I. Biel, V. Tsiminaki, J. Dines, and D. Gatica-Perez, "Hi youtube!: Personality impressions and verbal content in social video," in *Proceedings of the 15th ACM on International conference on multimodal interaction*. ACM, 2013, pp. 119–126.

[28] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2016, pp. 855–864.

[29] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[30] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[31] G. Ji, S. He, L. Xu, K. Liu, and J. Zhao, "Knowledge graph embedding via dynamic mapping matrix," 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)*, 2015, pp. 687–696.