Automatic Recognition of Allusions in Tang Poetry Based on BERT

Xuemei Tang[1,2], Shichen Liang[1,2], Jianyu Zheng[3], Renfen Hu[1,2], Zhiying Liu[1,2]
[1]Institute of Chinese Information Processing, Beijing Normal University, Beijing, China
[2]UltraPower-BNU Joint Laboratory for Artificial Intelligence, Beijing Normal University, Beijing, China
{tangxuemei, shhichen, irishu}@mail.bnu.edu.cn,liuzhy@bnu.edu.cn
[3]Advanced Innovation Center for Future Education, Beijing Normal University, Beijing, China
zheng_jianyu@126.com

***Abstract*—In this paper, we propose an automated method for recognize allusions in Tang poetry. The representation of text is trained by BERT pre-trained by *The SiKuQuanShu*. The TOP-20 candidate allusions have the highest semantic similarity to the original sentence. Then update the ranking of candidate allusions by rule-based ranking algorithm. In the final experimental results, the precision of the correct allusion same as the final ranking TOP-1 reached 63.74%, the precision of the correct allusion appears in the final ranking TOP-3 reached 70.66%, and the precision of the correct allusion appears in the final ranking TOP-5 reached 74.82%.**

Keywords-Tang poetry; allusions; automated recognize; BERT

I. INTRODUCTION

Tang poetry is another monument after the "Book of Songs" and "Chu Ci" in the history of Chinese poetry. Tang poetry often expresses rich ideological significance in a concise form. For example, Tang Dynasty often quoted allusions, and allusions often contain far-reaching meanings. Poets can quote stories or words in ancient books to express their thoughts. The Tang Dynasty poets preferred to quote allusions in many methods, they also good at adopting the methods of explicitly quote, implicitly quote, positively quote, oppositely quote, practicality, virtually quote, etc. The Tang Dynasty intellectuals had broad vision, rich knowledge structure and the allusions they quoted almost all over the classic literature before Tang Dynasty [1]. The method of poet quoted allusions change frequently and allusions is far from the source, which increases the language barriers for readers to read and appreciate. In order to solve this problem, in 1980s, some scholars have compiled the "Full Dictionary of Tang Poetry" to help Tang poet learners learn and understand Tang poetry [2].

In the environment of high-speed informationization, the way of knowledge propagation in the original book form can no longer meet people's needs. How to enable learners to acquire knowledge efficiently and conveniently is a problem that researchers are solving. With the rise of artificial intelligence, it has made breakthroughs on syntactic analysis and semantic analysis. This paper takes the automatic recognition the allusions in Tang poetry as a natural language processing task, combine semantic computing and ranking algorithm to automatically recognize the allusions in Tang poetry.

When the poet quotes the same allusion in poem sentence, the thoughts or feelings expressed by sentence may be very similar. For example, the allusion of "桃花源 comes from Tao Yuanming's "桃花源记 (Peach Blossom Spring)", which is mostly referred to as fairyland or seclusion in the Tang poetry. Based on this idea, we assume that if two poems quoted a same allusion then two poems are semantically similar. Therefore, we select some candidate allusions by calculating the semantic similarity, and then use the ranking algorithm to rank the candidate allusions to get the final recognition result. According to the sixth edition of the Modern Chinese Dictionary, the concept of the allusion: "The story or expression in the ancient book quoted in the poetry" [3]. The allusions to be recognized in this paper refer to the stories or words in the ancient books quoted in the poems of Tang poetry.

In this paper, we propose a method to automatically recognize the allusions in Tang poetry, and obtain better experimental results. At the same time, we propose a rule-based ranking algorithm suitable for Tang poetry allusion recognition. We hope that our work will provide assistance to Tang poetry learners and provide some reference for Tang poetry researchers.

II. RELATED WORK

The automatic recognition of allusions in Tang poetry is a special task, and some researchers have achieved some achievement in related tasks. Yao Rui [4] in 2011 segmented the allusions and poems, and then used the word matching method to identify the allusions in the poems. The dictionary method and mutual information method are combined in the word segmentation method. If the allusion completely appears in the poem, the method can effectively identify the allusion of the poem, but when the allusion quoted indirectly, the method has a poor recognition effect. Moreover, the method is based entirely on keyword matching and does not incorporate any natural language processing techniques. Dan Roth [5] 2014 used natural language processing techniques to identify entities in Wikipedia documents and link to Wikipedia pages of entities that match the current context. The method of recognize entities, utilize n-gram, phrase extraction and other methods, while adding external entity resources. After the entity is recognized, an entity often corresponds to multiple Wikipedia pages. By computing the semantic similarity between Wikipedia pages of the entities and the context in which the entity currently located, choose the entity page that best matches the current context.

978-1-7281-5014-7/19/$31.00 ©2019 IEEE

The method proposed by Dan Roth can effectively recognize the entities and give the explain pages of the entities. Inspired by this method, this paper uses this method to recognize the allusions quoted in Tang poems. Because of the wide variety of allusions, named entity recognition is not suitable. Therefore, we first calculate the semantic similarity between the example sentences (below called original sentence) and the poems in the candidate set. The pomes in the candidate set are known which allusion quoted. Then the TOP-K candidate allusions with the highest semantic similarity are ranked. Through ranking, the highest order is the allusion quoted by the original sentence.

III. DATA AND METHOD

A. Data

The data used in this paper comes from the "Complete Dictionary of the Complete Tang Poetry" written by Fan Zhilin [2], which includes the dictionary of the allusions, interpretation, and some example Tang poems quoted the allusions. The dictionary includes two volumes, which basically include all the allusions quoted in the Tang Dynasty poems. The example contents of the dictionary are shown in Figure 1.

After obtaining the data of the entire dictionaries from the CNKI China Tool Book Network Publishing Library, the data was cleaned and simplified, obtain 19,630 allusions examples. In the dictionaries, there are a large number of synonymous allusions. For example, the interpretation of the "团扇" is "见班女扇条"(equal to “班女扇”). In the experiment, the "团扇" and the "班女扇" will be merged into synonymous allusions, retain one of them. Through the synonymous merger, obtain 4,961 allusions. We extracted 815 allusions with more than 5 example Tang poems’ sentences from the 4961 allusions. Then, one sentence is randomly selected as the test data, other four sentences are selected as the candidate example sentences, and a total of 815 example sentences in the test set. The candidate set has a total of 3260 example sentences, and each example sentence corresponds to the allusion quoted in the sentence. At the same time, we retain the interpretation of 815 allusions. Table 1 shows a part of test data.

Table 1. A part of test data

NO.	Allusions	Candidate
1	非熊(FeiXiong)	非熊德愧当周辅，称杰叨惭首汉臣
2	许由(XuYou)	不学尧年隐，空令傲许由
3	嵇阮(JiRuan)	嵇阮没来无酒客，应刘亡后少诗人
4	五柳(WuLiu)	旧业传一经，新官栽五柳
5	冯唐(FengTang)	此地含香从白首，冯唐何事怨明时

B. Method

The experiment mainly includes three parts: text vectorization, semantic similarity calculation and ranking. The experimental methods used by these three modules are described below.

Text vectorization is a very important step in natural language processing tasks. In this paper, we use Google’s 2018 release Bidirectional Encoder Representation from Transformers (BERT) to represent text. This model was pre-trained by two tasks: 1) predict central word by context words; 2) predict next sentence for current sentence.

BERT training process: First, some words of the corpus is masked, so that the model predicts the masked words according to the context, and the general model is initially trained. Then, pick out successive context sentences from the corpus and let the transformer model identify whether the sentences are the next sentences. These two steps together to complete the pre-training, become a language representation model that can achieve omnidirectional prediction of the context.

The BERT training process is unsupervised, don’t require manual intervention and labeling, making it possible to train with very large-scale corpora at low cost. At the same time, the model combines the context of all layers of the neural network for training, so that the trained model can understand the semantics in combination with context and achieve more accurate text prediction generation when dealing with question- answer or language reasoning tasks. The model achieves optimal results in multiple natural language processing tasks through pre-training and fine tuning. The structure of the BERT model is shown in Figure 2. The training parameters are set as Table 2.

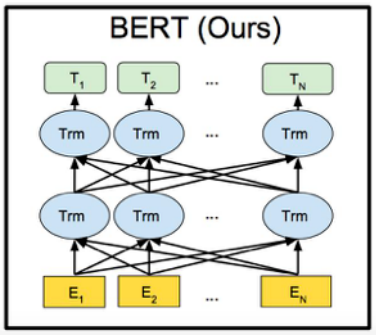


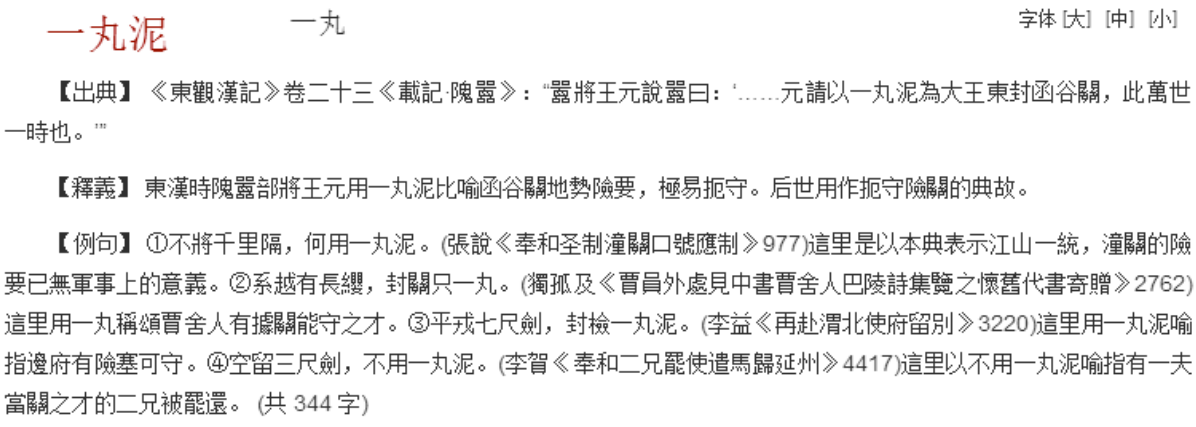
Figure 1. BERT model structure [6]

Layers	Hidden	Self-	All
12	768	12	110M

Table 2. BERTBASE parameters set

Because the data of our paper is poetry, most are single-character words, it may be not suitable to directly use the BERT pre-trained Chinese character model published by Google. For our task, we use *The SiKuQuanShu* to pre-train the BERT model to get the classical Chinese BERT language model.

The BERTBASE model structure has 12 layers, and each layer can output a sentence representation vector. Inorder to choose which layer of output is more suitable as a representation for allusion recognition task, we output the poem sentence representation from the BERTBASE model -1, -2, and -3 layers, respectively, to compare the effect of the semantic representation of the poem sentences by three layers. We use three layers’ representation to calculate three Tang poem sentences with the highest semantic similarity to the poem sentence "杜陵犹识汉，桃源不避秦" (original sentence). The semantic similarity calculation function shown as equation (1). As shown in Table 3, through calculating withthe –l layer representation. The sentence with highest semantic similarity to original sentence is "不知今有汉，唯言昔避秦", not only semantic similar, but also quoted same


一丸泥　　一丸　　字体 [大] [中] [小]

【出典】《東觀漢記》卷二十三《載記·隗囂》："囂將王元說囂曰：'……元請以一丸泥為大王東封函谷關，此萬世一時也。'"

【釋義】東漢時隗囂部將王元用一丸泥比喻函谷關地勢險要，極易扼守。后世用作扼守險關的典故。

【例句】①不將千里隔，何用一丸泥。(張說《奉和圣制潼關口號應制》977)這里是以本典表示江山一統，潼關的險要已無軍事上的意義。②系越有長纓，封關只一丸。(獨孤及《賈員外處見中書賈舍人巴陵詩集覽之懷舊代書寄贈》2762)這里用一丸稱頌賈舍人有據關能守之才。③平戎七尺劍，封檢一丸泥。(李益《再赴渭北使府留別》3220)這里用一丸泥喻指邊府有險塞可守。④空留三尺劍，不用一丸泥。(李賀《奉和二兄罷使遣馬歸延州》4417)這里以不用一丸泥喻指有一夫當關之才的二兄被罷還。(共 344 字)

Figure 1. Example contents of the dictionary

allusion "避秦". The results output by -2 and -3 layers are not ideal. Since we choose the sentence representation vector
output from the last layer of the model as the semantic representation for the poems and allusions' interpretation.

After vectorization of all poem sentences and allusions, then we calculate semantic similarity between example sentence and candidates. In the experiment, we use the cosine similarity as the semantic similarity calculation method, shown as equation (1).

$$s^n = \frac{A \cdot B^n}{\|A\|\|B^n\|} = \frac{\sum_{i=1}^{d} A_i \times B_i}{\sqrt{\sum_{i=1}^{d} A_i^2}\sqrt{\sum_{i=1}^{d} B_i^2}} \quad (1)$$

$\boldsymbol{A}$ is the representation of original sentence; B is the representation of the candidate sentence. The dimension of vector is 768, $i \in [1,768]$. $S^A = \{s^1, s^2 \ldots s^n\}$, $n \in [1,3260]$. If the value of s^n close to 1, illustrate that NO.n candidate is more similar to sentence $\boldsymbol{A}$.

By calculating the semantic similarity between the original poem sentence and the candidate sentences, get the K-sentence candidates with the highest semantic similarity to the original sentence, there is an initial ranking among K-sentence, and then the candidate example sentences and their corresponding allusions are ranked by the ranking algorithm. Finally, the highest ranking is the recognize result.

If the same N-gram phrase appears in two poems, then the two poems are more semantically related, and the content to be expressed may have certain overlap, and they are more likely to quote the same allusion; the Tang Dynasty poets often directly quote allusion, there may be some N-gram overlap between the poem sentence and the allusion; at the same time, the interpretation of the allusion is the meaning expressed by the allusion, then the poem sentences quoted allusion and the interpretation of allusion may be semantic similar. Therefore, in the ranking process, we use the number of N-gram overlap to update the initial semantic similarity $s^{1..n}$ and update the ranking of the candidate allusions. In this paper, because the poem sentence is short, bi-gram and unigram are used when counting the number overlap grams. The whole ranking process is as follow:

RANK0: According to the semantic similarity, there are TOP-K candidate sentences with the highest semantic similarity, we get an initial ranking for TOP-K sentences. If the candidate sentence of the similarity TOP-1 has a bi-gram overlap with the original sentence, we maintain the initial ranking. If not, entry RANK1.

RANK1: Count the number of overlap bi-gram between TOP-K candidate sentences and the original sentence respectively, N. $s^{1..n}$ updated by N to $s_1^{1\ldots n}$ according equation (2). To ensure that the semantic similarity is not cleared to 0, the initial value of N is set to 1. The candidate sentences are ranked by the new $s_1^{1\ldots n}$.

$$s_1^{1\ldots n} = N \times s^{1\ldots n} \ (N \geq 1,\ n \in [1,3260]) \quad (2)$$

After RANK1, if the new ranking is different from the initial ranking, the ranking of RANK1 would be finally result; if the ranking is the same as the initial ranking, imply the RANK1 is invalid, and entry RANK2.

RANK2: We choose the allusions corresponding to the TOP-K candidate sentences as candidate allusions, calculate the unigram overlap degree between the candidate allusions and the original sentence, and obtain the unigram overlap number M. Use M to update the semantic similarity $s_2^{1\ldots n}$, as calculated by equation (3).

$$s_3^{1\ldots n} = N \times s_2^{1\ldots n} \ (s_1^{1\ldots n} = N \times s^{1\ldots n}) \quad (3)$$

$s_2^{1\ldots n}$ is different from $s^{1\ldots n}$ and $s_1^{1\ldots n}$., and $s_2^{1\ldots n}$ is the semantic similarity between the interpretation of the candidate allusion and the original sentence, and the calculation method is consistent with $s^{1\ldots n}$. The candidate allusions are ranked by a new semantic similarity $s_3^{1\ldots n}$.

The final highest-order allusion is the final result of the recognition of the allusion of original sentence.

IV. Experiment Result and Discuss

A. Experiment result

Based on the sentence embedding trained by pre-training BERTBASE, the semantic similarity of the 815 original sentence in the test set and the 3260 candidate sentences in the candidate set are calculated respectively, and the TOP-20 sentences with the highest similarity are chosen as the candidate poem sentences,

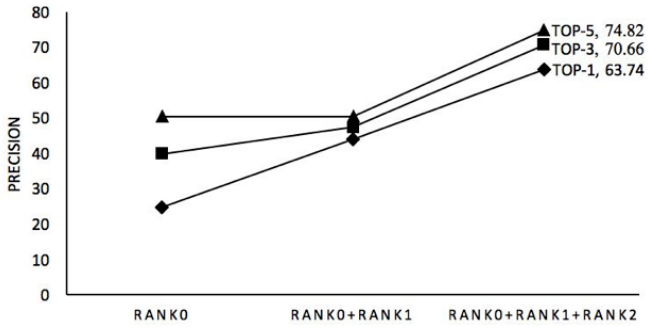


Figure 2. After different ranking method, the changes of precision

Table 3. The results of different ranking method (%)

	TOP-1	TOP-3	TOP-5
RANK0	24.71	39.72	50.34
RANK0+RANK1	43.87	47.34	50.34
RANK0+RANK1+RANK2	**63.74**	**70.66**	**74.82**

and then the allusions corresponding to TOP-20 sentences are candidate allusions, then rank candidates by rule-based ranking method. In the experiment, we choose precision metrics to measure the experimental result, and the precision function as equation (4). c is the number of correct recognition; a is the size of test data.

$$p = \frac{c}{a} \qquad (4)$$

B. Maintaining the Integrity of the Specifications

The template is used to format your paper and style the text. All margins, column widths, line spaces, and text fonts are prescribed; please do not alter them. You may note peculiarities. For example, the head margin in this template measures proportionately more than is customary. This measurement and others are deliberate, using specifications that anticipate your paper as one part of the entire proceedings, and not as an independent document. Please do not revise any of the current designations.

The precision of different ranking methods is shown in Table 3. TOP-1 indicates that the precision of the correct allusion appears in the first place of the final ranking; TOP-3 indicates that the precision of the correct answer appears in the TOP-3 of the final ranking result; TOP-5 indicates that the precision of the correct answer appears in the top five of the final ranking result.

It can be seen from the change of the accuracy rate in Figure 3 that the precision from RANK0 to RANK2 shows an upward trend; as the ranking range of the calculation precision expanding (TOP-1 to TOP-5), the precision also increases. From the experiment results of RANK0, the precision from TOP-1 to TOP-5 increased by 25.63%, which proves our assumption that the poems quoted the same allusion have similar characteristics in semantics. More than 50% of the poems quoted same allusion as the original sentence are gathered into the TOP-5. The precision increase from 24.71% to 43.87% by RANK1 combined with the N-gram overlap number, indicating that RANK1 is a valid ranking method, and the similarity is updated by the number of bi-gram overlap. The method improves correct candidate to the more advanced position from the previous order; the combination of RANK1 to RANK2, the precision increases by 19.87%, indicating that the way is effective to update the similarity by the candidate allusion and the original sentence unigram overlaps.

After combining the three RANK, the precision of TOP-1 allusion recognition reached 63.74%; the recognition precision of TOP-3 reached 70.66%; the recognition precision of TOP-5 reached 74.82%. experiment result imply that we proposed semantic similarity and rule-based ranking method can effectively recognize the allusions in some Tang poems.

B. Discuss

The ranking algorithm proposed in this paper has achieved good results in the experiment. The following will specifically analyze the role of the two ranking algorithms in the allusion recognition experiment. The original sentence is "天生逸世姿，竹马不曾骑", the allusion quoted in this sentence is "竹马（Bamboo horse）", and the candidate sentence of the similarity TOP-20 obtained by calculating the semantic similarity with the 3260 candidate sentences. As shown in Table 4, the poem sentence that quoted the same allusion as the original sentence, "爱向竹栏骑竹马，懒于金地聚金沙" is ranked 14th. The original sentence and this sentence has bi-gram overlap, so when entry RANK1, the similarity of the sentence will become higher and the ranking will improve. As shown in Table 5, the sentence has a bi-gram overlap with the original sentence. At this time, N is 2, s_1 becomes twice as big as s, and the ranking rises to the position of TOP-1. The ranking result and initial ranking is different, since don't need to carry out RANK2, then "竹马(Bamboo horse)" corresponding to is the TOP-1 sentence, "竹马（Bamboo horse）" is the result of the recognition, same as the allusion quoted in the original sentence, the recognition is correct. The method of updating the ranking by using the bigram overlap number is more effective for the poems directly quoted allusions, and the effect for the poems indirectly quoted allusions is poor.

When RANK1 is invalid, the candidate allusions entry RANK2. The original sentence "冶长空得罪，夷甫岂言钱 " quoted the allusion " 治长非罪 ", in this poem sentence did not directly quote the allusion completely, resulting in RANK1 invalid. Then carried out RANK2 and we obtain the corresponding allusion through the TOP-20 candidate example sentences, that is, the TOP-20 candidate allusions. Then we calculate semantic similarity s_2 between the interpretation of the candidate allusion and the original sentence, get a new ranking for Top-20 candidate allusions. M is the number of unigram overlap of the candidate allusions and the original sentence, the semantic similarity is updated by M to s_3. According to s_3, update the new ranking. As shown in Table 7, the candidate allusion "冶长非罪" is ranked TOP-1, and the allusion "一钱(one penny)"

Table 4. Candidate sentences initial ranking for original sentence“天生逸世姿，竹马不曾骑”

Ranking	Candidate allusions	Candidate sentences	S
TOP-1	仁风(mercy wind)	衣同莱子曾分笔，扇似袁宏别有天	0.794
TOP-2	将军树(General tree)	虽投定远笔，未坐将军树	0.790
TOP-4 --Top-13	…	…	...
TOP-14	竹马(bamboo horse)	爱向竹栏骑竹马，懒于金地聚金沙	0.774

Table 5. After RANK1, the ranking result of“天生逸世姿，竹马不曾骑”

Ranking	Candidate allusions	Candidate sentences	S_1
TOP-1	竹马(bamboo horse)	爱向竹栏骑竹马，懒于金地聚金沙	1.549
TOP-2	仁风(mercy wind)	衣同莱子曾分笔，扇似袁宏别有天	0.794
TOP-3	将军树(General tree)	虽投定远笔，未坐将军树	0.790
TOP-4	酷似仰牢之	酷似牢之玉不如，落星山下白云居	0.789

Table 6. After RANK1, the ranking result of“冶长空得罪，夷甫岂言钱”

Ranking	Candidate allusions	Candidate sentences	S_1
TOP-1	献赋(offer poem)	甘泉未献扬雄赋，吏道何劳贾谊才	0.849
TOP-2	原宪贫	贾生独未达，原宪竟忘贫	0.832
TOP-7	冶长非罪	公冶本非罪，潘郎一为民	0.818
TOP-11	一钱(one penny)	宋均颜未老，刘宠骨应贫	0.815

Table 7. After RANK2, the ranking result of“冶长空得罪，夷甫岂言钱”

Ranking	Candidate allusions	S_3
TOP-1	冶长非罪()	2.101
TOP-2	一钱(one penny)	1.085
TOP-3	曹刘(CaoLiu)	0.565
TOP-4	张博望(Zhang Bowang)	0.539

coincides with the original sentence, so the ranking is also improved. RANK2 has a better recognition effect on poem sentences partially quoted allusions.

The allusion recognition bad case mainly includes two cases. 1) We need to calculate the semantic similarity between many original sentences and candidate example sentences. We only take out the TOP-20 candidate allusions, it is found that the correct allusion does not appear in TOP-20, resulting in recognition errors. For example, "秋风能再热，团扇不辞劳". Although the original sentence is a directly quoted the "团扇(Circular fan)" allusion, but because the TOP-20 semantically most similar candidate poem sentences does not appear the "团扇(Circular fan)" allusion, ranking is invalid, causing the final recognition result to be incorrect. In order to solve this problem, TOP-30 and TOP-50 candidate poem sentences can be selected for ranking, and try to ensure that correct allusions appear in the candidate sentences. 2) Because the number of bi-gram and unigram overlap between the original sentence and the candidate sentence and the candidate allusions need to be counted during ranking, when there is no overlap, the ranking method fails. For example, the original sentence "古时填渤澥 ，今日凿崆峒 " completely indirectly quoted the allusion "精卫(JingWei)", there is no bigram and unigram overlap between the candidate sentence and the allusion and the original sentence, and the ranking method fails, only depend on initial ranking. To solve this problem, the semantic similarity can be updated by the number of synonym pairs between candidate sentences and original sentence in the ranking process. Although the original sentence and the candidate sentences do not have bi-grams and unigrams overlap, but there may be multiple pairs of synonyms, then the two poems may also quote the same allusion.

V. CONCLUSION

In this paper, we use the semantic similarity and ranking algorithm to recognize the allusions quoted in Tang poetry, and obtain good experimental results. Firstly, BERTBASE pre-train by The SiKuQuanShu corpus, the pre -trained BERT model is used to get the

representation for poems and allusions. Then calculate the semantic similarity between the original sentence and the candidate sentences. Next, select the TOP-20 candidate poem sentences with the highest semantic similarity, get a initial ranking. Finally, according to the number of bi-gram and unigram overlap between the original sentence and the candidate sentences or candidate allusions to update the semantic similarity, update the initial ranking by the new semantic similarity. After two rounds of ranking, get the final ranking. In the final experiment results, the precision of the correct allusion same as the final ranking TOP-1 reached 63.74%, the precision of the correct allusion appears in the final ranking TOP-3 reached 70.66%, and the precision of the correct allusion appears in the final ranking TOP -5 reached 74.82%. In the future work, we will continue to improve the ranking algorithm, hoping to improve the precision of the allusion recognition.

ACKNOWLEDGMENT

This work is supported by the Fundamental Research Funds for the Central Universities, National Language Committee Research Program of China (No. ZDI135-42) and National Social Science Fund of China (No. 18CYY029).

REFERENCES

[1] Yang Liu. Reading "*The Complete Dictionary of the Complete Poems of Tang Dynasty*" [J]. Literary Heritage. 1989-01: 123-126

[2] Fan Zhilin, Wu Gengqi. "The Complete Dictionary of the Complete Poems of Tang Dynasty" [M]. Hubei Dictionary Press. 1989

[3] Dictionary Editing Room, Institute of Linguistics, Chinese Academy of Social Sciences. Modern Chinese Dictionary [M]. The Commercial Press. 2012

[4] Yao Rui. An automatic analysis system for poetry based on the ontology of allusions [J]. Software Guide. 2011, Volume 10, Issue 06: 80-82

[5] Roth D, Ji H, Chang MW, et al. *Wikification and beyond: the challenge of entity and concept Grounding*[C]. Proceeding of the 52nd Annual Meeting of the Association for Computational Linguistics: Tutorial, 2014: page7

[6] Devlin J, Chang MW, Lee K, et al. *BERT: pre-training of deep bidrectional transformers forlanguage understanding*[C]. Proceeding of the 16th North American Chapter of the Association for Computational Linguistics, 2018

[7] Xiong Qingyuan. *The Emotion of the Complete Dictionary of the Complete Poems of Tang Dynasty*[J]. Journal of Huanggang Teachers College. 1997, Volume 17, 02: 24-26

[8] Yu Tingting. Research on the Allusions of the Analects of Confucius in Tang Dynasty[D]. Shaanxi Normal University, Master's Degree. 2017

[9] Wang Guanghan. *Rethinking on the Origin of Allusions* [J]. Ancient Chinese Studies. 2004, 04: 90-93

[10] Summer. *Research on Semantic Similarity Calculation of Chinese Words*[J]. Computer Engineering. 2007, 06: 1191-19

[11] Liu Hongzhe. *Research on text semantic similarity calculation method* [D]. Beijing Jiaotong University, doctoral thesis, 2012

[12] Xu Dezhi, Wang Huaimin. *Research on semantic similarity calculation method based on ontology*[J]. Computer Engineering and Applications. 2007-08: 154-156

[13] Yoon Kim. 2014. Convolutional neural networks for sentence classification. In Proceedings of EMNLP 2014:1746–1751.

[14] Jeffrey Pennington. Richard Socher. Christo- pher D. Manning. 2014. *Glove: Global vectors for word rep- resentation*. In Proceedings of EMNLP 2014:1532– 1543.

[15] Melanie J.Bell. Martin Schafer. 2013.Semantic Transparency Challenges for Distributional Semantics.Proceedings of the IWCS 2013 workshop: Towards a formal distributional semantics. Potsdam: Association for Computional Linguistics: 1-10.