Overview of the NLPCC-ICCPOL 2016 Shared Task: Chinese Word Similarity Measurement

Yunfang Wu(✉) and Wei Li

Key Laboratory of Computational Linguistics,
Peking University, Beijing 100871, China
{wuyf, liweitj47}@pku.edu.cn

Abstract. Word similarity computation is a fundamental task for natural language processing. We organize a semantic campaign of Chinese word similarity measurement at NLPCC-ICCPOL 2016. This task provides a benchmark dataset of Chinese word similarity (PKU-500 dataset), including 500 word pairs with their similarity scores. There are 21 teams submitting 24 systems in this campaign. In this paper, we describe clearly the data preparation and word similarity annotation, make an in-depth analysis on the evaluation results and give a brief introduction to participating systems.

Keywords: Word similarity · Similarity computation · Semantic campaign

1 Introduction

Word similarity computation is to automatically predict the similarity degree of word pairs, which is a fundamental task for many natural language processing (NLP) systems, such as question answering, information retrieval, paraphrase detection and textual entailment. There are two kinds of ways to evaluate word similarities: (i) intrinsic evaluation: compute the correlation coefficient between the automatic predicted results with the human labelled similarity scores; (ii) extrinsic evaluation: apply the word similarities to a specific downstream task, like name entity recognition or relation extraction. The extrinsic evaluation is a valid method, but it does not allow us to understand the properties of lexical similarities without further analysis. Therefore, this paper focuses on the first method of intrinsic evaluation. There needs a benchmark dataset to evaluate and compare different systems on computing lexical similarity, and thus to encourage more researches on this problem.

In English, there are quite a few open datasets that are commonly used as benchmark for evaluating word similarity. The first data RG is from Rubenstein and Goodenough [1]. The data contains 65 pairs of nouns, and the human subjects were asked to order the pairs according to the amount of "similarity meaning" and give a similarity value from 0.0–4.0. Miller and Charles [2] selected 30 of those pairs (MC data), and studied semantic similarity as a function of contexts in which words are used. The most widely used dataset is WordSim-353 [3]. It selected out 353 pairs of nouns and asked 16 human annotators to assign a numerical similarity score between 0 and 10. In the recent time, Huang et al. [4] created a new dataset, where the word pairs were presented in sentential context rather than in isolation.

C.-Y. Lin et al. (Eds.): NLPCC-ICCPOL 2016, LNAI 10102, pp. 828–839, 2016.
DOI: 10.1007/978-3-319-50496-4_75

However, such an open benchmark has been absent in Chinese for a long time, which becomes a bottleneck for Chinese word similarity computation. In the early and notable work of Liu and Li [5], only 39 word pairs were selected for evaluating. Jin and Wu [6] organized a campaign of evaluating Chinese word similarity at Semeval-2012. They translated the word pairs of WordSim-353 data to Chinese, and asked twenty human annotators to give a similarity score between 0 and 5. But only two teams submitted four systems in this campaign. Guo et al. [7] constructed a Chinese Polysemous Word Similarity Dataset, which contains 401 word pairs selected from Hownet, but this data focuses on polysemous words and so the data diversity is limited.

We organize a campaign of Chinese word similarity measurement at NLPCC-ICCPOL 2016, and construct a benchmark dataset (namely PKU-500). The dataset contains 500 Chinese word pairs, which are assigned similarity scores by twenty subjects. Researchers have shown great interest in this problem and totally 49 teams registered in this task. Finally 21 teams participated in our campaign and submitted 24 systems.

In this paper, we make an overview of this task. In Sect. 2, we introduce the dataset construction, including the diverse criteria for data collection and the annotation scheme of similarity scores. Section 3 describes the task setup and the evaluation metrics. In Sect. 4, we report the evaluation results of 24 systems and make a detailed analysis on the experimental results. We analyze the following factors that may affect Chinese word similarity computation: (i) the inter-annotator agreement in assigning similarity scores, (ii) part of speech, (iii) word length and (iv) polysemous words. In Sect. 5, we make a brief introduction to the participating systems. Section 6 gives the conclusion.

2 Dataset Construction

2.1 Word Selection

The commonly-used WordSim-353 dataset [3] only contains nouns and tries to have word pairs with a diverse set of similarity scores. But in recent years, researchers pay more attention to the semantic representations of some special words, like rare words and polysemous words. Accordingly, more important and diverse criteria should be considered in constructing the dataset.

- **Domain.** Covering both the traditional formal language and the recent web language.
- **Frequency.** The high-frequency words, middle-frequency words and low-frequency words should all be included.
- **Part of Speech.** Not only nouns but also verbs and adjectives should be included. Not only the content words (noun, verb and adjective) but also the functional words (e.g., adverb, conjunction) should be considered.
- **Word Length.** A Chinese word may be composed of one character, two characters, three characters or four characters. All these different types of words should be considered.

- **Word Sense.** Ambiguous words with multiple meanings are the most difficult part for lexical semantic researches, so some polysemous words should be included.
- **Polarity.** Words with different semantic orientations (positive vs. negative) should be included.

According to the above criteria, we selected words in the following procedure.

1. The data comes from two domains: three month *People's Daily News* and a large collection of WeiBo data.
2. All the data was word segmented and POS tagged using the open software ANSJ.
3. We extracted words separately from the two domains according to their frequency, part of speech and word length.
4. The automatically extracted words were further picked out manually by the first author of the paper, according to the word senses and semantic polarities. Finally, we got 514 words and 202 words from *People's Daily News* and WeiBo data, respectively.

2.2 Word Pair Generation

In total we selected 716 single words from the corpus in the previous phrase. Now, we will generate word pairs for evaluating word similarity.

1. For each target word, we automatically extracted three candidate words from HIT-CIR Tongyici Cilin (Extended): the first word lies in the same synset; the second is one of the words belonging to the parent node; the third one is randomly selected from other words.
2. For each target word, the candidate words were further picked out manually by the first author of the paper. We removed some candidate words and add some new words by linguistic insight, according to the criteria described in Sect. 2.1. Finally, we got 470 word pairs.
3. We selected other 30 word pairs that were translated from the WordSim-353 data.
4. Finally, we got 500 word pairs for Chinese word similarity measurement.

2.3 Similarity Score Annotation

We asked twenty graduate students to annotate similarity scores of word pairs. All the students are Chinese native speakers and major in Chinese linguistics. The similarity score is set to [1, 10], where 1 means two words are totally different and 10 means two words carry the same meaning. We calculated the average value of twenty humans as the final similarity score of each pair.

We didn't give any annotation guidelines to the human annotators, so the annotators were encouraged to judge just by their intuition of language. We didn't make a clear distinction between semantic relatedness and semantic similarity, because it is a notorious problem in lexical semantics and it depends on the application task you will do.

Tables 1 and 2 give some examples of word pairs with their similarity scores. Table 1 lists the top 10 similar words and Table 2 lists the top 10 dissimilar words.

Table 1. The top 10 similar word pairs

Word 1	Word 2	Score
WTO	世界贸易组织	10
紫禁城	故宫	10
计算机	电脑	9.9
赢	胜	9.8
维他命	维生素	9.5
化肥	化学肥料	9.5
课程表	课表	9.5
互联网	因特网	9.5
假货	赝品	9.5

Table 2. The top 10 dissimilar word pairs

Word 1	Word 2	Score
讲价	打架	1
教授	黄瓜	1
控制	恐高	1
阻力	天花板	1
结盟	无理取闹	1.1
调查	努力	1.1
玻璃	魔术师	1.1
干扰	上网	1.1
课堂	美食	1.1

Table 3. The top 10 word pairs with low standard deviation

Word 1	Word 2	Score	Stad.
教授	黄瓜	1	0.00
控制	恐高	1	0.00
阻力	天花板	1	0.00
WTO	世界贸易组织	10	0.00
紫禁城	故宫	10	0.00
讲价	打架	1	0.22
玻璃	魔术师	1.1	0.30
干扰	上网	1.1	0.30
课堂	美食	1.1	0.30

Table 4. The top 10 word pairs with high standard deviation

Word 1	Word 2	Score	Stad.
没戏	没辙	4.9	3.03
只管	尽管	4	2.94
GDP	生产力	6.5	2.80
包袱	段子	2.6	2.71
日期	时间	6	2.67
由此	通过	3.4	2.66
爱面子	好高骛远	4	2.56
一方面	一边	5.4	2.54
托福	GRE	8	2.54

In assigning similarity scores, some word pairs have quite high inter-annotator agreement while some other word pairs have very low inter-annotator agreement. Tables 3 and 4 list the top 10 word pairs with low standard deviation and high standard deviation, respectively.

It can be seen that two types of word pairs are likely to have high inter-annotator agreement. (1) Word pairs that refer to the same entity. For example, the word pair "紫禁城" vs. "故宫" gets a standard deviation 0.0 that means all twenty annotators give it the same score 10. (2) Word pairs that are totally different. For example, all twenty annotators give the pair "教授" vs. "黄瓜" the same score 1. However, for the following word pairs, different annotators tend to give different similarity scores. (1) Two words are functional words (e.g., "只管" vs. "尽管", "由此" vs. "通过", "一方面" vs. "一边"). (2) Two words are sematic related (e.g., "GDP" vs. "生产力", "日期" vs. "时间", "托福" vs. "GRE"). Annotators handle semantic related words differently: some annotators assign them a high similarity score, while other annotators assign them a low score. It is an interesting topic to investigate why annotators give different similarity scores to a pair of word, so for further research Appendix A lists the 91 word pairs with standard deviation greater than 2.0.

3 Task Setup

All 500 word pairs serve as the test data, and no training data is provided. We first released 40 word pairs as the trial data before the evaluation phrase. All kinds of strategies are welcome, including the traditional corpus-based distributional similarity, dictionary-based similarity computation, as well as the recently developed word embedding methods and deep learning models. Also, the participating systems are encouraged to use external resources.

In order to avoid over-fitting on this small test dataset, we released a large collection of 10,000 word pairs in the testing phrase. Our 500 word pairs were mixed in the large data, and the other word pairs were randomly generated from a large dictionary.

We use Spearman's rank correlation coefficient to evaluate the statistical dependence between the automatic computing results and the golden human labelled scores:

$$\rho = 1 - \frac{6\sum_{i=1}^{n}(R_{Xi} - R_{Yi})^2}{n(n^2 - 1)} \tag{1}$$

where n is the number of word pairs being evaluated, R_{Xi} and R_{Yi} are the standard deviations of the rank of automatic computing results and human labelled scores, respectively.

4 Evaluation Results and Analysis

4.1 Overall Results

Together 21 teams participated in our task and submitted 24 systems. Table 5 reports the evaluation results, listing the team ID, organizations and Spearman scores.

The best result of the wining team is 0.518, which is significantly better than the second best system by 6.1%. The last system gets a score 0.000, because it destroys the format of the testing data. Only the wining first system gets a Spearman score greater than 0.50; 16 systems are in 0.30–0.50; 7 systems are below 0.30. In the recent work of Schnabel et al. [8], they report a score of 0.640 on WordSim-353 data by using CBOW word embeddings. It demonstrates that there is still a large gap between Chinese and English word similarity computation.

4.2 Inter-annotator Agreement and Evaluation Results

We think those word pairs with low inter-annotator agreement in assigning similarity scores will have negative effects on evaluation results. To address the inconsistency of annotators, we remove those word pairs with a standard deviation greater than 2.0, and we get 401 word pairs with low standard deviation. Figure 1 reports the evaluation results of the top three systems on 401 word pairs.

Table 5. The overall evaluation results

Team ID	Organization	Spear.
SXUCFN-QA	Shanxi University	0.518
DLUT_NLPer	Dalian University of Technology	0.457
CQUT_AC996	Chongqing University of Technology	0.436
nlp_polyu	The Hong Kong Polytechnic University	0.421
BLCU_CNLR	Beijing Language and Culture University	0.414
Cbrain	Institute of Automation, Chinese Academy of Sciences	0.412
CIST	Beijing University of Posts and Telecommunications	0.405
wanghao.ftd	Shanghai Jiao Tong University	0.405
DUTNLP	Dalian University of Technology	0.372
BIT_CWSM	Beijing Institute of Technology	0.371
NJUST-CWS	Nanjing University of Science and Technology	0.365
TJIIP	Tongji University	0.357
SWJTU_CCIT	Southwest Jiaotong University	0.349
QLUNLP_1	Qilu University of Technology	0.327
QLUNLP_2		0.328
QLUNLP_3		0.314
QLUNLP_4		0.234
CCNU BeliefTeam	Central China Normal University	0.316
DM&S Lab	Beijing University of Technology	0.286
Zsw	University of South China	0.272
AngryXYZ	University of Beijing Science and Technology	0.268
Zutcsnlp2016	Zhongyuan University of Technology	0.206
whut_nlp	Wuhan University	0.014
USC	University of South China	0.000

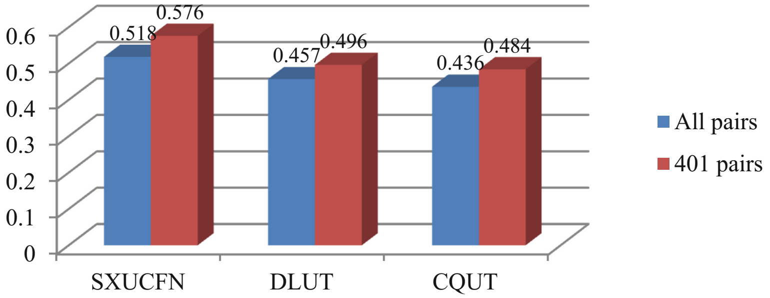

Fig. 1. The performances of top three systems on word pairs with low standard deviation

As our expectation, the performances of the top three systems are consistently improved on 401 word pairs, and the Spearman scores ρ * 100 are improved by 5.8, 3.9 and 4.8 for SXUCFN, DLUT and CQUT, respectively. For further research, Appendix A lists the 91 word pairs with high standard deviation.

4.3 Part of Speech on Similarity Computation

Both the RG and WordSim-353 dataset only contain nouns. We want to know whether different parts of speech will affect the automatic computation of word similarity, so we make a comparison on different parts of speech by the top three systems, as shown in Fig. 2. Those word pairs where both words are nouns constitute a set of Noun, and we compare the automatic scores of word pairs within this set with the golden human labelled score. In the same way, we compute the Spearman scores of Verb and Adjective. The Noun set includes 247 word pairs, and the Verb set includes 104 word pairs and Adjective set 52 word pairs. The other word pairs are: (i) two words have different parts of speech; (ii) both words are functional words. These three sets of word pairs will be released separately in the website of NLPCC-ICCPOL 2016 for further researches.

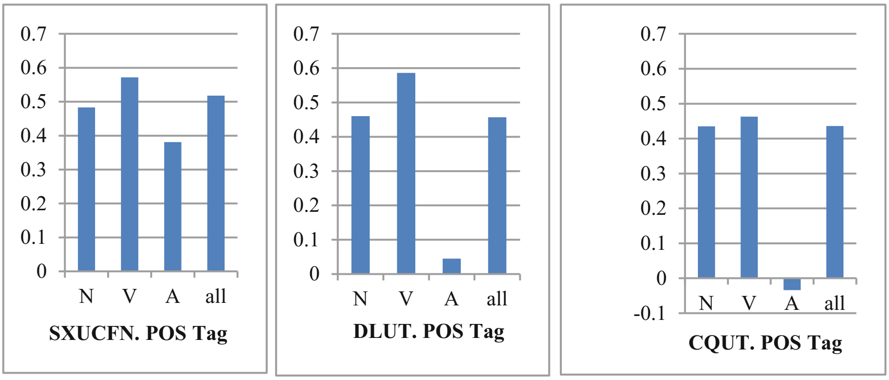

Fig. 2. The performances of top three systems on different parts of speech

To our surprise, the Verb gets the highest score by all three systems. As our expectation, the Noun also gets promising results. However, the Adjective gets a quite low score, which is much lower than the average value. The system CQUT gets a negative value for adjectives, which assigns the similarity score of 24 adjective word pairs as over 9.0.

4.4 Word Length on Similarity Computation

We make a further analysis on the lexical similarity of words with different word length. We regard a word pair as One Character if it contains a single-character word; we regard a word pair as Three Characters if it contains a three-character word. In the same way, we extract the word pairs of Two Characters and Four Characters. Please note that these different sets may have overlap. In our testing data, the One Character set includes 25 word pairs, and the Three Characters set includes 90 word pairs and the Four Characters set 43 word pairs. The majority part is the set of Two Characters, which includes 450 word pairs. These different sets of word pairs will be released in the

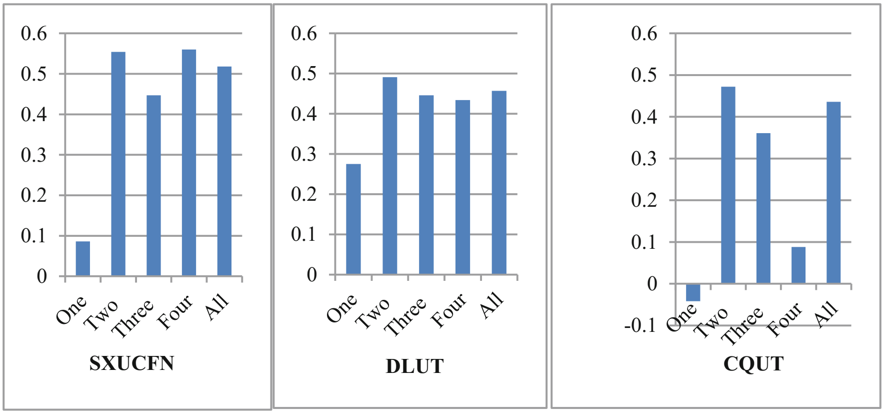

Fig. 3. The performances of word pairs with different word length

website of NLPCC-ICCPOL 2016 for more researches. The following Fig. 3 reports the evaluation results of the top three systems on different word length.

It can be seen that all three systems get very high scores for word pairs where one of the words is composed of two characters; however, they get quite low scores for word pairs that contain a single-character word. This is because (i) the character information is important to predict the meaning of a Chinese compound word while we can't make use of this knowledge for a single-character word; (ii) most of single-character words are ambiguous and have multiple senses.

4.5 Polysemous Words on Similarity Computation

The semantic representations of polysemous words have been a challenging task in natural language processing, and word sense disambiguation is a traditional hot topic in the research community. In recent years, there has been an increasing interest in learning sense embeddings from large corpus [4, 7, 9].

In our task, we want to investigate whether polysemous words pose more difficulties in word similarity computation. We divide our dataset into two subsets: both words are monosemous; one of the words is polysemous. Following the work of Guo et al. [7], we first extracted polysemous words based on HowNet (version 2000), but it ended up 415 word pairs, because most words in our data were assigned multiple senses by HowNet. Therefore, we resort to "Xiandai Hanyu Cidian (version 5)" to find polysemous words, and get 268 word pairs that contain at least one polysemous word in each word pair. To encourage more researches in this problem, these two subsets will be released in the website. Figure 4 reports the evaluation results of the top three systems on polysemous words compared with monosemous words.

It can be seen that all the three systems get a worse performance on polysemouswords than monosemous words, but only by a small margin. It drops 1%, 3.8%, and 1.2% for the wining system, the second system and the third system, respectively. It demonstrates that the effect of polysemous words on lexical similarity computation isn't as large as people thought.

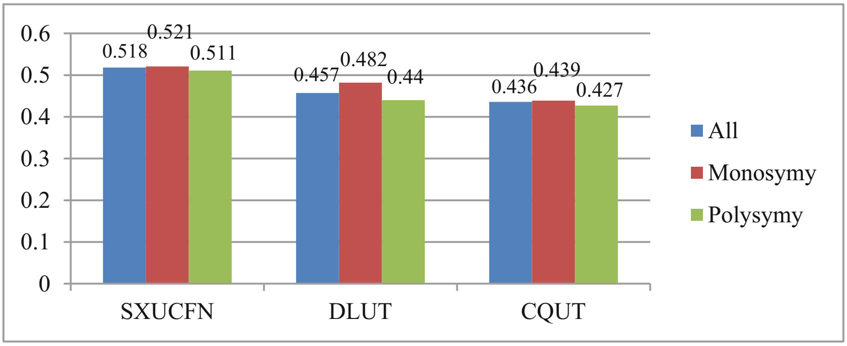

Fig. 4. The performances of polysemous words compared with monosemous words

5 Participating Systems

Our task has attracted much concern in the research community, and 21 teams participated in our task and submitted their results. Another one team submitted results after the deadline. In this section, we will make a brief introduction to the top three systems. We will introduce the methods and resources they used, and discuss the results they achieved.

Extensive researches have been conducted on lexical similarity computation. Theses work can be briefly clustered into three groups. (i) Thesaurus-based method. The similarity degree of a word pair is predicted based on the manually constructed dictionary, like WordNet [10, 11], HowNet [5] and Tongyici Cilin [12]. These methods rely heavily on the hierarchical structure of the dictionary, and can't deal with those out of vocabulary (OOV) words. (ii) Corpus-based method [13–15]. These methods are based on the distributional hypothesis which assumes that similar words occur in similar context. They represent the contextual features as vectors from a large corpus and then compute the similarity score between vectors. (iii) Corpus-based embedding method. These methods exploit neural networks to extract distributed representations of words based on a large corpus, and then compute the similarity of word embeddings, such as the work of Mikolov et al. [16] and Guo et al. [7]. Due to the limitations of single methods, most systems in our task exploit the combining strategy.

The wining first system comes from Shanxi University [17], which achieves a Spearman score 0.524. They propose a combining strategy that exploits different similarity computing methods based on a variety of semantic resources. Their work can be divided into three steps. (1) Compute the similarity score based on Hownet (*sim1*), and compute the cosine distance between two vectors that are pre-trained using the Word2Vector tool (*sim2*). Then the similarity score is set to be the average value $sim = (sim1 + sim2)/2$. (2) If both words of a pair evoke the same frame according to the Chinese FrameNet [18], the score should be $sim = (sim + 10.0)/2$. (3) If both words are in DaiCilin, Synonym dictionary (Tongyiic Cilin) and Antonym dictionary, the score is further updated according to heuristic rules. The experimental results show that (i) Combing the HowNet-based method and corpus-based embedding method gets a significant improvement over the individual methods; (ii) Adding the semantic resources of DaiCilin, Synonym dictionary and Antonym dictionary further improves the performance.

The second best system comes from Dalian University of Technology [19], which achieves a Spearman score 0.457. They propose a framework by combining word embeddings and Tongyiic Cilin. The similarity computation method based on Tongyici Clilin is developed from the algorithm of Tian and Zhao [12], which fully exploits the coding and hierarchical structure information to predict the similarity degree between two words. They employ the skip-gram model to learn the continuous word vectors from text corpora and then do similarity computation on embedding vectors. The interesting part is that they use the weak similarity score (WSS) to weakly supervise the learning process of word embedding, which is automatically computed based on some retrieval statistics and linguistic features. The combination strategy exploits different ensemble learning methods, including Max, Min, Replace, Arithmetic Mean and Geometric Mean. The Cilin method gets a Spearman value 0.405, the W2V method gets 0.311, and the combing method with Arithmetic Mean gets 0.457, which ranks the second in all participating systems. After the submission, they make further studies: (1) enhance the embedding model by merging equivalent English word embeddings; (2) learn the co-occurrence sequence via LSTM networks. They finally get a Spearman value 0.541, which is the best result on PKU-500 data to date.

The third best system is from Chongqing University of Technology. They adopt the lexicon-based method by using Tongyici Cilin, and their method is derived from the algorithm of Tian and Zhao [12], which takes advantage of the structure information and coding rules to estimate the similarity of a word pair. They propose two improvements. (1) For polysemous words, they take the average value of similarity scores across all senses rather than the biggest similarity score. (2) For OOV words that are not recorded in Tongyici Cilin, they split the words into characters and extract those words that contain the character. They then calculate the similarities of those words and take the average value. Their work gets a Spearman score 0.436. However, as shown in Figs. 1 and 2, their method is not robust and behaves badly for adjectives and single-character words.

6 Conclusion

This paper gives an overview of the NLPCC-ICCPOL 2016 shared task 3 “Chinese word similarity measurement”. We describe clearly the diverse criteria in selecting words, the data preparation and similarity score annotation by twenty graduate students. In total 24 systems participated in our task. We make a detailed analysis in the evaluation results, considering the inter-annotator agreement, part of speech, word length and polysemous words.

The wining first system gets a Spearman score 0.518, which is much lower than the Spearman score 0.640 on WordSim-353 data reported in the work of Schnabel et al. [8], suggesting that there is much room to improve for Chinese word similarity computation. Most works employ the traditional methods (thesaurus-based method) and the simple corpus-based embedding method, therefore more advanced model and novel ways are encouraged to use. Besides the intrinsic evaluation based on our PKU-500 dataset, extrinsic evaluation is also welcome by applying it to real-world applications, such as question answering and machine translation.

Acknowledgement. This work is supported by National High Technology Research and Development Program of China (2015AA015403), National Natural Science Foundation of China (61371129, 61572245), Key Program of Social Science foundation of China (12&ZD227).

Appendix A: 91 Word Pairs with Standard Deviation Greater Than 2

[没戏 没辙] [只管 尽管] [**GDP** 生产力] [包袱 段子] [日期 时间] [由此 通过]
[爱面子 好高骛远] [一方面 一边] [托福 **GRE**] [严厉 严谨] [抄袭 克隆]
[悲喜 大悲大喜] [亏 幸亏] [老气 土气] [蹩脚 差强人意] [容易 顺利]
[狭隘 狭窄] [害臊 腼腆] [理解 理会] [的哥 司机] [娇艳 幽美] [幻境 红楼梦]
[自然 环境] [权限 权利] [几乎 差点儿] [酣睡 打鼾] [振兴 建设] [节日 假日]
[依稀 清晰] [伟大 壮烈] [典型 代表] [出神 发楞] [冷僻 晦涩] [面 首]
[发票 账单] [物品 物质] [回收站 垃圾篓] [必须 必需] [路子 后门]
[牛脾气 我行我素] [免费 便宜] [江湖 红尘] [塞车 拥挤] [要面子 虚荣心]
[琢磨 镂刻] [大小 多少] [候选人 备胎] [旅客 驴友] [多角度 多元化]
[信物 物件] [豆蔻年华 黄金时代] [血液 红细胞] [酷 爽] [质量 重量]
[牺牲 粉身碎骨] [隆重 重要] [天赋 技能] [身姿 身手] [事变 后院起火]
[鸣谢 酬答] [硅谷 中关村] [平凡 平庸] [了不得 好] [许可证 执照]
[线路 行程] [与 以及] [和谐 平安] [怯懦 胆小鬼] [是非 方圆] [大 高]
[手续 过程] [高峰 山巅] [崛起 凸起] [辛勤 夜以继日] [环境 生态]
[渣 废品] [杂事 闲事] [商标 符号] [右翼 左派] [实践 进行] [借口 理由]
[收费 缴纳] [享受 大快朵颐] [吸引力 地磁力] [工作日 开放日]
[合理 合理性] [违纪 贪污] [言语 语言] [买卖 营销] [光盘 硬盘]

References

1. Rubenstein, H., Goodenough, J.B.: Contextual correlates of synonymy. Commun. ACM **8** (10), 627–633 (1965)
2. Miller, G.A., Charles, W.G.: Contextual correlates of semantic similarity. Lang. Cogn. Neurosci. **6**(1), 1–28 (1991)
3. Finkelstein, L., Gabrilovich, E., Matias, Y., Rivlin, E., Solan, Z., Wolfman, G., et al.: Placing search in context: the concept revisited. TOIS **20**, 116–131 (2002)
4. Huang, E.H., Socher, R., Manning, C.D., Ng, A.Y.: Improving word representations via global context and multiple word prototypes. In: Proceedings of the Association for Computational Linguistics (2012)
5. Liu, Q., Li, S.: Word similarity computing based on HowNet. Int. J. Comput. Linguist. Chin. Lang. Process. **7**, 59–76 (2002)

6. Jin, P., Wu, Y.: SemEval-2012 task 4: evaluating Chinese word similarity. In: First Joint Conference on Lexical and Computational Semantics (2012)
7. Guo, J., Che, W., Wang, H., Liu, T.: Learning sense-specific word embeddings by exploiting bilingual resources. In: Proceedings of COLING 2014 (2014)
8. Schnabel, T., Labutov, I., Mimno, D., Joachims, T.: Evaluation methods for unsupervised word embeddings. In: Proceedings of Empirical Methods in Natural Language Processing (2015)
9. Trask, A., Michalak, P., Liu, J.: Sense2vec - a fast and accurate method for word sense disambiguation in neural word embeddings (2015). arXiv preprint: arXiv:1511.06388
10. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: International Joint Conference on Artificial Intelligence (1995)
11. Meng, L., Huang, R., Gu, J.: A review of semantic similarity measures in WordNet. Int. J. Hybrid Inf. Technol. **6**, 1–12 (2013)
12. Tian, J.L., Zhao, W.: Words similarity algorithm based on Tongyici Cilin in semantic web adaptive learning system. J. Jilin Univ. **28**(06), 602–608 (2010)
13. Lin, D.: Automatic retrieval and clustering of similar words. In: Proceedings of Coling-ACL 2002 (2002)
14. Agirre, E., Alfonseca, E., Hall, K., Kravalova, J., Paşca, M., Soroa, A.: A study on similarity and relatedness using distributional and WordNet-based approaches. In: Proceedings of Human Language Technology (2009)
15. Shi, J., Wu, Y., Qiu, L., Lv, X.: Chinese lexical semantic similarity computing based on large-scale corpus. J. Chin. Inf. Process. **27**(1), 1–6 (2013)
16. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems (2013)
17. Guo, S., Guan, Y., Li, R., Zhang, Q.: Chinese word similarity computing based on combination strategy. In: Proceedings of NLPCC 2016 (2016)
18. Liu, K.: Research on Chinese FrameNet construction and application technologies. J. Chin. Inf. Process. **6**, 47 (2011)
19. Pei, J., Zhang C., Huang, D., Ma, J.: Combining word embedding and semantic lexicon for Chinese word similarity computation. In: Proceedings of NLPCC 2016 (2016)