An Embedded Representation for TongYiCi CiLin and Its Evaluation on Tasks

DUAN Yuguang[1,3], LIU Yang[1,2], YU Shiwen[1,2]

(1. Key Laboratory of Computational Linguistics (Ministry of Education), Peking University,
2. Institute of Computational Linguistics, Peking University,
3. Yuanpei College, Peking University, Beijing 100871, China)

Abstract: In natural language processing (NLP), to learn embedded representation is an effective approach to capturing semantics from language resources. However, by now, this has been much limited to using large-scale corpora, with little attention to extracting rational knowledge from knowledge bases. In this paper, based on TongYiCi CiLin, a famous Chinese thesaurus, we present a method for implanting rational knowledge into embedded representation, then evaluate it on different NLP tasks. According to the hierarchical encodings for morphemic and lexical meanings in TongYiCi CiLin, we design multiple templates to create instances as pseudo-sentences from these pieces of knowledge, and apply word2vec to obtain CiLin2Vec, the sememe and word embeddings of new kinds as for TongYiCi CiLin. For evaluation, tasks of semantic compositionality, analogical reasoning and word similarity measurement are taken into consideration. We make progress and breakthrough on the tasks, reaching an accuracy of over 0.9 for both semantic compositionality and analogical reasoning, which proves that the pieces of rational knowledge have been appropriately implanted and shows very promising prospects for adoption of the knowledge bases.

Keywords: TongYiCi CiLin, embedded representation, semantic compositionality, analogical reasoning, word similarity measurement

Natural language understanding and natural language analysis are very important to artificial intelligence. The main approaches to these tasks can be divided into two types of methods—knowledge-based rational methods and corpora-based empirical methods. As for rational methods, TongYiCi CiLin(CiLin) is a classic knowledge base for Chinese. CiLin provides a semantic hierarchical structure for each word by categorizing words into synonym groups and assigning a hierarchical code to each group. It is very useful to natural language processing(NLP) tasks, like word similarity measurement[1-3], entity relationship extraction[4-5], semantic role labeling[6], and texts categorization[7]. As for empirical method, corpora-based distributed representation has developed fast, from Latent Semantic Analysis (LSA)[8-10], to feedforward neural network(FFNN)-trained word embedding[11], serving as the mainstream methods in NLP applications[12-14].

Rational methods, incorporating common sense and expertise, are more interpretable, but always need to design specific algorithms for specific tasks, and have poor domain adaptability. Empirical methods, on the other hand, have high automation based on non-supervision learning, and the obtained word embeddings can be applied to multiple NLP tasks. Given the complementary advantages of these two methods, it would be worthwhile to apply the empirical methods to extract semantic knowledge from knowledge base to maximize the common sense and expertise in human knowledge, and to obtain word embeddings that are adaptable to multiple tasks.

Previously, some researchers have already noticed the need in implanting rational knowledge into word embeddings. There are work trying to dig out the simplified adjacency relationships in WordNet graphs[15], or to use definitions in dictionaries as reference[16], to implant such rational knowledge into word embeddings. Other research groups focused on obtaining representations for sememes or synonyms from pre-trained word embeddings[17], or improve the interpretability of word embeddings by mapping pre-trained word embeddings to synonym sets[18]. There are also some research trying to add semantic or syntactic knowledge into corpora to obtain more informative word embeddings[19], or train word embeddings on pseudo-corpora created through random walk method[20]. However, these methods are mostly based on real large-sized corpora, only partially adding semantic or syntactic knowledge into the training process. Some methods attempt to get rid of real corpora[15,20], but are rather indirect and complicated in implanting knowledge or building pseudo-corpora. Besides, there are no direct and general strategies that can implant all kinds of knowledge base into word embeddings.

This paper proposes a new pseudo-corpora generation method that can implant all kinds of knowledge base into word embeddings. We use the extensive version of CiLin http://www.ltp-cloud.com/download, designed by Harbin Institute of Technology（HIT）， as ontology, to create 3 types of pseudo-corpora based on the hierarchical semantic code for each word: sememe-coded type, sememe-coded extensive type, and word-coded type. The pseudo-corpora incorporate knowledge from CiLin along with its distributional pattern. Then we obtain word embeddings and sememe embeddings with word2vec on them. Finally, we evaluate the performance of the obtained embeddings on three NLP tasks: semantic compositionality, analogy, and word similarity measurement.

1 Background

1.1 CiLin knowledge representation

CiLin is a Chinese synonym categorization knowledge base[21], created by Jiaju Mei, and extended by HIT. The extensive version contains 77,343 words, 90,102 sememes, which are divided into 12 super categories, 95 major categories, 1,428 subordinate categories, 4,026 word groups, 17,797 meta word groups. The super category is represented by a capitalized letter; the major category is represented by a lower case letter; the subordinate category is represented by two integers; the word group is represented by a capitalized letter; the meta word group is represented by two integers, and there is a punctuation suffix serving as additional note: “=” refers to synonym group; “#” refers to relative word group; “@” refers to single word group. Take “Aa01A01=” as an example. This code refers to a synonym group containing {人(person), 士(scholar), 人物(figure), …}. A more specific illustration is given in Figure 1. We will refer to such code as semantic code later.

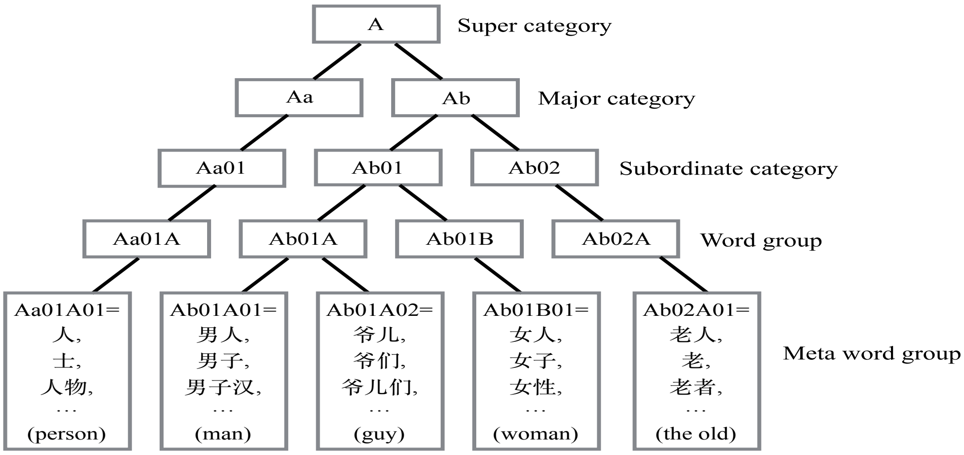
A
Super category
Aa
Ab
Major category
Aa01
Ab01
Ab02
Subordinate category
Aa01A
Ab01A
Ab01B
Ab02A
Word group
Aa01A01= 人, 士, 人物, … (person)
Ab01A01= 男人, 男子, 男子汉, … (man)
Ab01A02= 爷儿, 爷们, 爷儿们, … (guy)
Ab01B01= 女人, 女子, 女性, … (woman)
Ab02A01= 老人, 老, 老者, … (the old)
Meta word group

Fig. 1 The structure of CiLin

1.2 Distributional semantics and distributed representation

Distributional semantics is a kind of data-driven semantic analysis, based on Harris' hypothesis[22]: linguistic items with similar distributions have similar meanings. It aims to measure semantic similarity from a statistical perspective.

Based on this, Hinton et al[23] proposed distributed representation, which is also called embedded representation. The core idea is that the properties of a concept are represented by a pattern of activation over feature units and this pattern of activation is determined by the interactions of a potentially very large number of units for instances of the concept. Embedded representation differs from one-hot representation in the form: the one-hot representation are high-dimensional vectors where each dimension represents a specific word, while embedded representation uses dense low-dimensional vectors to represent all the words, as shown in Table 1. Therefore, embedded representation can better represent the similarity among synonyms by assigning similar vectors to them, and they are more expressive. Even in binary form, an n-dimensional embedded representation system can represent 2^n different concepts, while the one-hot representation system can only represent n concepts[24].

Tab. 1 One-hot representation vs. embedded representation

word	One-hot	Embedded
cat	[0,0,…,0,1,0,0,…,0]	[0.14,…,0.61,…,-0.27]
dog	[0,0,…,0,0,1,0,…,0]	[0.18,…,0.71,…,-0.31]

1.3 Semantic compositionality

Semantic compositionality is an important NLP task, since it is a core competence of human to generate infinite sentences with a limited set of units and rules[25]. A lot of previous work focused on using word embeddings to compose phrase embeddings or sentence embeddings[26-29]. However, such embeddings are hard to obtain or interpret, and semantic compositionality remains to be a tough question in both computer and cognitive science[30-31].

Previously, most research on this task only studied the possibility to use word embeddings to compose higher level units, and ignore the lower level units, like sememes. However, words are not the basic-level units in language. Noreen[32] pointed out that basic-level units are semantemes, equaling to specific definitions of a certain word in the dictionary. By decompose a semanteme, we can obtain the minimum units—sememes. For example, man="person"×"male"×"adult"[33]. Here, "person", "male", "adult" are all sememes. Based on this, we proposed a new semantic compositionality task, which aims to use sememe embeddings to compose word embeddings. We carried out a case study on CiLin to evaluate this task.

1.4 Analogy

Analogy task was first raised by Mikolov et al[34]. This task is used to predict semantic and syntactic relativeness of word embeddings. A famous example is "man : woman :: father : w_i". In the ideal case, the word vector of w_i should be obtained by a calculation on the other three given words, i.e., $V(w_i)$ = V("woman") - V("man") + V("father"). For evaluation on analogy tasks, an ideal answer for w_i is given in advance (e.g., "mother" in this example), and the cosine similarity between the vector calculated by the upper formula and that of the ideal answer is computed to represent the performance on this task.

Chen et al[35] provided a dataset on analogy task https://github.com/Leonard-Xu/CWE, which includes 953 tuples divided into 3 types: 1) 506 tuples related to capitals and countries; 2) 175 tuples related to states and cities; 3) 272 tuples related to relationships. Since some words in this dataset are not included in CiLin, we finally used 921 tuples as the evaluation set: 1) 506 tuples related to capitals and countries; 2) 175 tuples related to states and cities; 3) 240 tuples related to relationships.

1.5 Word similarity measurement

Word similarity measurement is fundamental to synonyms detection, disambiguation, information extraction and so on. There are 2 approaches to this task[36]: 1) corpora-based statistic analysis methods, in particular the analysis of word frequency and word distribution[37]. This approach is largely dependent on the property of the corpora[38]. Presently the most frequently used method is to calculate the cosine similarity among the word embeddings trained by neural network. 2) Knowledge-based concepts analysis [39]. This approach can be divided into 4 types: path-similarity, feature-similarity, information-content-similarity, and concept-annotation-similarity[40].

In Chinese, MC30 https://github.com/huyingxi/Synonyms/blob/master/VALUATION.md and wordsim297 https://github.com/thunlp/SE-WRL/blob/master/datasets/wordsim-297.txt are two popular datasets for task evaluation, consisting of word pairs and human rating on similarity for each pair. For evaluation, the Pearson correlation r × 100 between the predicted similarity by the model and the human rating is calculated to determine the performance.

2 CiLin Embeddings

2.1 Adjustment on CiLin structure

The original CiLin hierarchical structure has no specific word groups for intermediate nodes, but each layer sets more semantic constraints to the codes as well as to the word groups. Therefore, we can view the new constraints in each layer as a new sememe added to the semantic content of the upper node, and thus each lower node contains all the sememes contained in its upper nodes, and the word meaning can be represented as a set of sememes. Besides, all the words at the leaf nodes in CiLin have the same semantic code, which doesn't reflect the language facts. In fact, the semantic granularity varies across the lexicon bank according to the level of semantic abstraction, and the position of the words on the hierarchical tree should be determined by their semantic granularity. Therefore, we first adjust the structure of CiLin based on this consideration.

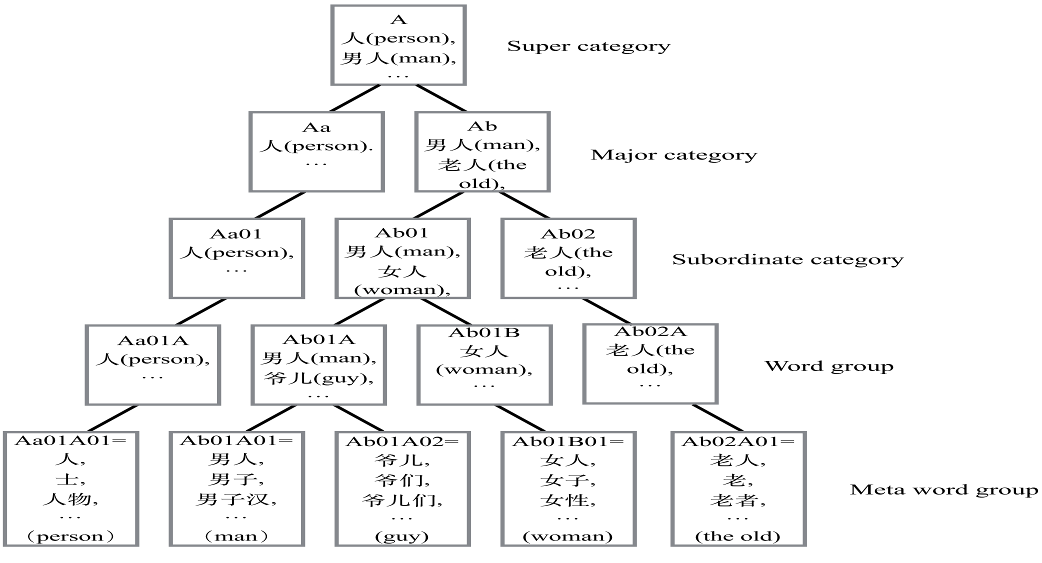

Fig. 2 The adjusted structure of CiLin

Since the first word of each word group is usually at higher level of abstraction and granularity and can represent the meaning of the whole group, we make the following adjustment to CiLin: in bottom-up direction, we extract the first word of each word group and hang it to its father node. In this way, all the intermediate nodes have their own word groups as well, and the generalization of each word group reflects the sememe information encoded in the group code. Besides, the semantic description of each word now varies according to its level of abstraction and granularity, and can better reflect the language fact. The adjusted structure is shown in Figure 2.

2.2 Pseudo-corpora created from CiLin

Since neural networks encode the words into embeddings according to their distribution in the corpora, we need to convert the codes in CiLin into a format reflecting the distributional relationship between words and sememes that can be applied to neural networks.

The adjusted CiLin structure has 23,570 nodes, each representing a different concept. We create 3 types of pseudo-corpora based on the hierarchical semantic code for each word: sememe-coded type, sememe-coded extensive type, and word-coded type. Since nodes at lower levels has more specific codes and are closer to the real meaning of the words, we decided the distance between each code and the core word in the pseudo-corpora according to their distribution in the hierarchical tree—codes at lower levels are closer to the core words—to incorporate the distributional relationship between words and their sememes. Then we train word embeddings and sememe embeddings with word2vec model. Table 2 presents some examples of sentences in pseudo-corpora.

Tab. 2 Examples of different sentence templates for CiLin

Sememe-coded type	<BOS> A A A A A 人(person) A A A A A <EOS> <BOS> A Aa Aa Aa Aa 人(person) Aa Aa Aa Aa A <EOS> <BOS> A Aa Aa01 Aa01 Aa01 人(person) Aa01 Aa01 Aa01 Aa A <EOS> <BOS> A Aa Aa01 Aa01A Aa01A 人(person) Aa01A Aa01A Aa01 Aa A <EOS> <BOS> A Aa Aa01 Aa01A Aa01A01= 人(person) Aa01A01= Aa01A Aa01 Aa A <EOS> ……
Sememe-coded extensive	<BOS> A A A A A 人(person) A A A A A <EOS> <BOS> A A A A A 士(scholar) A A A A A <EOS> …… <BOS> A A A A A 翁(senior) A A A A A <EOS> <BOS> A Aa Aa Aa Aa 人(person) Aa Aa Aa Aa A <EOS> <BOS> A Aa Aa Aa Aa 士(scholar) Aa Aa Aa Aa A <EOS> …… <BOS> A Aa Aa Aa Aa 翁(senior) Aa Aa Aa Aa A <EOS> ……
Word-coded type	<BOS> 人(person) 男人(man) 高个儿(tall man) 居民(resident) 职工(worker) 劳动者(labor) 健康人(healthy man) 亲戚(relative) 鼻祖(ancestor) 朋友(friend) 英雄(hero) 知识分子(intellect) 教徒(believer) 反动派(rebel) 人(person) 反动派(rebel) 教徒(believer) 知识分子(intellect) 英雄(hero) 朋友(friend) 鼻祖(ancestor) 亲戚(relative) 健康人(healthy man) 劳动者(labor) 职工(worker) 居民(resident) 高个儿(tall man) 男人(man) 人(person) <EOS> <BOS>人(person) 男人(man) 高个儿(tall man) 居民(resident) 职工(worker) 劳动者(labor) 健康人(healthy man) 亲戚(relative) 鼻祖(ancestor) 朋友(friend) 英雄(hero) 知识分子(intellect) 教徒(believer) 反动派(rebel) 人(person) 我(me) 你(you) 他(him) 自己(self) 谁(whom) 人(person) 谁(whom) 自己(self) 他(him) 你(you) 我(me) 人(person) 反动派(rebel) 教徒(believer) 知识分子(intellect) 英雄(hero) 朋友(friend) 鼻祖(ancestor) 亲戚(relative) 健康人(healthy man) 劳动者(labor) 职工(worker) 居民(resident) 高个儿(tall man) 男人(man) 人(person) <EOS> ……

Sememe-coded type: all the ancestor nodes of the each word contribute to its sememe set that can be used to form its sememe-coded pseudo-sentence. The distance between the sememe codes and the core words are determined by their distance on the hierarchical tree. The sentences all have a palindrome form with the core word in the middle, 5 codes in the prefix, and 5 codes in the suffix. If the ancestor codes are less than 5, then the closest code is repeated to cover the empty positions. This form guarantees the fixed length and the symmetry of the sentences, which suit both CBOW model and Skip-Gram model.

Sememe-coded extensive type: based on previous algorithms for word similarity measurement, we extracted similar words whose similarity is above certain threshold, and then replaced the core word in each sememe-coded sentence with its similar words. In this way, the pseudo-corpora is expanded largely. This type of pseudo-corpora leverages the previous rational methods to improve the distributional similarity among the similar words and therefore the obtained embeddings can better incorporate the similarity relationship. In this research, we applied the methods provided by Tian[1], Lv[2], and Zhu[3] to create sememe-coded extensive pseudo-corproa. Take "人"(person) as an example. According to Tian[1]'s algorithm, given certain similarity threshold, the similar word set of "人"(person) includes {人(person), 士(scholar), 人物(figure), 人士(personage), 人氏(family), 人选(candidate), 人类(human), 生人(foreigner), 全人类(mankind), 人口(population), 口(resident), 食指(thumb), 翁(old man)}.

Word-coded type: by replacing all the sememe codes in the sememe-coded pseudo-sentences with the corresponding words in the group, we can obtain word-coded pseudo-corpora. This is based on the hypothesis that the generalization of each word group reflects the sememe information in the group code, and therefore we can use the words in the group as a substitute for the code. Word-coded pseudo-sentences have no fixed length, but are also symmetric.

2.3 Training CiLin embeddings

We use Word2vec to train CiLin embeddings. There are two typical models in Word2vec—CBOW and Skip-Gram, which can directly obtain word embeddings on unlabeled corpora. CBOW model predicts the core word w_i according to its context embeddings' sum or average, while Skip-Gram model predicts the context embeddings based on the w_i embedding.

In particular, we apply CBOW and Skip-Gram models provided in gensim word2vec package https://github.com/RaRe-Technologies/gensim to train embeddings on three pseudo-corpora, without using any real corpora in addition. We also compare the performance generated by different window sizes in the training process.

Since the sememe codes also exist in the sememe-coded and sememe-coded extensive pseudo-corpora, we can obtain sememe embeddings from these two corpora as well, and their distributional relationship with the core words should be encoded in the embeddings given that we create the corpora to incorporate such relationships. We also train word embeddings on Chinese wikipedia corpora https://dumps.wikimedia.org to make a comparison with our models in the evaluation section.

3 Evaluation on CiLin embeddings

3.1 Semantic compositionality

Since each sememe code in CiLin includes all levels of sememe information, and the semantic meaning of each word can be represented by a set of sememes, ideally, a word embedding can be replaced by the normalized sum of its sememe embeddings. The cosine similarity between the original word embedding and its sememe-composed embedding can be computed to represent the performance of CiLin embeddings on semantic compositionality task. We used the following formula to calculate the sememe-composed word embeddings:

$$w_1 = \Sigma(s_1, ..., s_i, ..., s_n)$$
$$V(w_1) = \Sigma(\alpha_1 V(s_1), ..., \alpha_i V(s_i), ..., \alpha_n V(s_n))$$
$$\Sigma(\alpha_1, ..., \alpha_i, ..., \alpha_n) = 1$$

w_1 is the target word. s_i refers to its sememe$_i$. V(x) refers to the embeddings(vectors) of words or sememes. α_i is the weight of s_i. Here, we calculate α_i according to the level of its s_i on the hierarchical tree, with two candidate formulas: $\alpha_{i+1} = 0.5\alpha_i$ or $\alpha_{i+1} = 2\alpha_i$.

Since some words are polysemes and have multiple candidate sememe-composed embeddings, we select the one that can render the highest performance in the evaluation. The results are shown in Table 3.

Tab. 3 Evaluation results on semantic compositionality task: cosine similarty between sememe-vector-combined word vector and original word vector %

Model	CBOW					Skip-gram				
	3	4	5	6	7	3	4	5	6	7
Sememe-coded type	85.62	87.70	**88.31**	87.66	87.24	95.01	**95.84**	95.82	95.62	95.37
Sememe-coded extensive[1]	**73.61**	70.67	69.25	68.29	65.73	**82.21**	80.71	78.11	77.86	77.14
Sememe-coded extensive [2]	**64.31**	63.32	60.95	58.98	58.29	**78.62**	76.78	75.01	74.29	74.33
Sememe-coded extensive [3]	62.86	**63.29**	60.68	58.83	58.46	**79.60**	78.02	75.73	75.39	75.01

* 3~7 in the second row state the size of ngram window in the training process; [1]、[2]、[3] respectively refer to the methods for word similarity measurement in reference [1]、[2]、[3]. The same applies to the following tables。

Here, for sememe-coded extensive pseudo-sentences, the weight formula for the best models is $\alpha_{i+1} = 0.5\alpha_i$. The similarity thresholds ρ for the three rational methods are respectively: $\rho_{[1]}$=0.89，$\rho_{[2]}$=0.65，$\rho_{[3]}$=0.84. The parameters of the word2vec models are: iteration=5, dimension=300, min-count=0. The other parameters are default.

According to the results, Skip-Gram models have better performance than CBOW. The best window size is 3-4. Sememe-coded pseudo-corpora are the best type of pseudo-corpora, reaching a score of 95.84%. The results suggest that the sememe information has been implanted into the model, and sememe-composed word embeddings can successfully replace the original word embeddings. This further proved that CiLin hierarchical structure can properly represent semantic meaning and the pseudo-corpora have maintained such property. In a word, CiLin knowledge can facilitate semantic compositionality task.

3.2 Analogy

The performance on analogy task is shown in Table 4. Here, the weight formula for sememe-coded extensive pseudo-corpora is $\alpha_{i+1} = 0.5\alpha_i$. The similarity thresholds ρ and the parameters for the models are same as before, except that for the model on Chinese wikipedia, the min-count is 3.

Tab. 4 Evaluation results on analogical reasoning task: cosine similarty between analogical word vector and correct word vector %

Model	CBOW					Skip-gram				
Window	3	4	5	6	7	3	4	5	6	7
Wikipedia corpora	80.83	81.39	81.67	81.91	**81.93**	80.6	80.99	81.42	81.58	**81.84**
Sememe-coded type	64.62	65.04	**66.24**	63.46	64.65	92.99	92.89	92.93	**93.32**	93.16
Sememe-composed embeddings	**95.12**	93.76	92.87	93.28	93.74	**94.37**	92.95	93.07	92.43	92.43
Sememe-coded extensive[1]	57.2	60.54	56.92	**61.14**	58.56	79.43	79.28	80.27	81.04	80.89
Sememe-coded extensive [2]	**85.23**	83.8	81.24	80.93	81.27	92	**92.58**	92.04	91.97	92.05
Sememe-coded extensive [3]	83.36	**86.18**	84.19	84.89	85.09	93.3	93.4	93.33	93.07	**93.46**

Sememe-composed embeddings[1]	91.02	**91.92**	91.74	89.53	90.95	92	**92.24**	91.96	91.7	92.05
Sememe-composed embeddings [2]	**92.64**	89.85	89.58	88.66	91.78	91.06	91.93	**92.39**	91.53	92.31
Sememe-composed embeddings [3]	**90.31**	90.3	88.76	88.73	89.5	92.43	91.5	92.37	92.54	**92.67**
Word-coded type	77.88	78.19	**79.89**	78.04	76.64	74.66	76.12	74.88	76.6	**77.04**

According to the results, Skip-Gram models are better. The best pseudo-corpora are the sememe-coded ones, as the sememe-composed embeddings out of this type achieve a score of 94.37%, higher than the original embeddings. The results, again, prove that the semantic compositionality is successful and the sememe-composed embeddings can be further applied to other tasks. Besides, under the same training iteration, the embeddings trained on pseudo-corpora generate better performance on the task than that trained on wikipedia. The possible reason behind this is that sememes have no ambiguity, compared to words, and meantime can cover all the meaning conveyed by word. In addition, CiLin has a rigorous sememe code format, which is incorporated in the pseudo-corpora, and we can manually control the distribution of words and sememes in the pseudo-corpora to decrease the noise, while the real corpora can't avoid such noise. Another thing to mention is that the performance achieved by the new method is better than the best performance reported by Chen[35], which is 72.99%.

3.3 Word similarity measurement

The performance on word similarity measurement task is shown in Table 5 (MC30) and Table 6 (wordsim297). Here, for sememe-coded extensive pseudo-corpora, the weight formula for the best models is $\alpha_{i+1}= 2\alpha_i$. The similarity thresholds ρ and the parameters for the models are same as those for analogy task. Since CiLin only contains 277 word pairs out of wordsim297, the final evaluation dataset of wordsim297 is cut to those 277 word pairs.

Tab. 5 Evaluation results on word similarity measurement task (MC30): Pearson r × 100

Model	CBOW					Skip-gram				
Window	3	4	5	6	7	3	4	5	6	7
Wikipedia corpora	65.99	64.24	64.41	**67.09**	65.72	**67.08**	65.50	66.09	65.07	66.20
Sememe-coded type	22.31	11.02	27.88	24.13	**39.19**	42.88	60.82	70.92	**74.39**	70.71
Sememe-composed embeddings	67.50	59.47	62.64	67.43	**69.99**	74.35	70.36	76.10	**77.65**	77.02
Sememe-coded extensive[1]	20.12	2.63	**33.32**	14.95	-0.05	63.83	63.17	77.05	**75.60**	77.42
Sememe-coded extensive [2]	**46.23**	42.03	22.16	19.16	32.72	**73.20**	65.09	62.70	70.88	70.41
Sememe-coded extensive [3]	26.08	29.83	**40.52**	18.80	13.82	77.97	**84.73**	80.60	82.89	79.18
Sememe-composed embeddings[1]	74.59	70.47	76.65	73.83	**77.39**	84.86	81.39	84.60	82.04	**84.95**
Sememe-composed embeddings [2]	75.26	62.01	64.13	74.62	**76.46**	81.23	77.88	73.19	80.59	**82.08**
Sememe-composed embeddings [3]	67.41	78.59	63.38	71.64	**79.27**	68.84	82.00	81.99	81.19	**82.50**
Word-coded type	13.59	37.83	38.49	36.86	**44.70**	63.11	69.18	**78.85**	75.54	69.61

Tab. 6 Evaluation results on word similarity measurement task (wordsim297): Pearson r × 100

Model	CBOW					Skip-gram				
Window	3	4	5	6	7	3	4	5	6	7
Wikipedia corpora	60.76	62.32	62.77	63.92	**64.53**	58.09	58.93	**59.86**	59.79	59.25
Sememe-coded type	7.22	10.38	14.71	13.97	**18.76**	26.22	32.03	32.60	37.72	**33.19**
Sememe-composed embeddings	32.00	**32.28**	29.71	28.03	29.44	32.74	32.28	32.97	**33.62**	32.53
Sememe-coded extensive[1]	**21.32**	5.40	8.64	4.38	2.05	30.80	34.07	**40.27**	38.65	38.29
Sememe-coded extensive [2]	**16.36**	15.86	12.18	1.04	5.74	37.07	39.22	**39.42**	36.57	39.05
Sememe-coded extensive [3]	**23.39**	16.88	6.06	3.77	7.74	37.85	**39.22**	38.09	33.01	38.29
Sememe-composed embeddings[1]	**35.34**	28.95	34.15	31.76	32.01	36.33	38.97	38.71	**41.75**	38.26
Sememe-composed embeddings [2]	36.25	32.70	38.26	38.23	**39.45**	40.66	42.01	40.71	**42.74**	41.91
Sememe-composed embeddings [3]	34.25	35.88	34.58	38.50	**39.40**	39.40	41.17	40.73	38.71	**43.48**
Word-coded type	27.70	27.23	28.17	25.18	**32.46**	34.61	36.40	39.02	**39.19**	38.62

According to the results, Skip-Gram models are better. The best window size is 7. Sememe-composed embeddings have better performance than original word embeddings. The highest score is 84.95, achieved on the sememe-coded extensive pseudo-corpora which leverage Tian[1]'s algorithm, suggesting that the rational methods can be successfully adopted by the new method. Compared to the best performance reported in references[1], [2], [3], which are respectively: 49.39, 74.03, 79.24 (on MC30), and 35.53, 34.11, 42.22 (on wordsim297), the new method has better performance, which further proves that the new method has an edge at maximizing the utility of CiLin knowledge base.

Under the same training iteration, the embeddings trained on pseudo-corpora generated better performance on the task than that trained on wikipedia on MC30, but the opposite results are obtained on wordsim297. However, considering the rational methods have the same decrease in performance on wordsim297 as the new method, the decrease may be caused by the speciality of MC30, which has a rather small sample size.

In conclusion, embeddings trained on real corpora have more stable performance than embeddings training on pseudo-corpora, but this might depends on the property of the knowledge base as well, e.g. CiLin contains certain mistakes in synonyms categorization.

4. Conclusion

This research presents a new method for creating pseudo-corpora with knowledge base and training embeddings on pseudo-corpora, using CiLin as ontology for a case study, We evaluated the performance of different models with different window sizes on three NLP tasks: semantic compositionality, analogy, and word similarity measurement. The results have shown that the word and sememe embeddings obtained by the new methods can achieve better performance on all these tasks than previous rational methods. In particular, the accuracy on semantic compositionality and analogy are above 90%, suggesting the great potential of the new method. We also release the codes and models for this research online https://github.com/ariaduan/CiLin2Vec, for other researchers to use.

This new method can effectively leverage the existent knowledge base by implanting the knowledge into the pseudo-corpora and word embeddings. It can also incorporate the previous rational methods for pre-processing of the knowledge base. Besides, this method increases the interpretability of the obtained embeddings by combining rational and empirical methods. It also reduces the length of training period by replacing the real corpora with small-sized pseudo-corpora.

In the future, we expect to combine the real-corpora embeddings and the pseudo-corpora embeddings to further improve the performance on NLP tasks. We also plan to study the possibility of generalizing this method to adapt to all kinds of knowledge base, and apply this method to evaluate the performance of different knowledge bases. This method will also support the development of Peking University COOL resources.

References

[1] TIAN J L, ZHAO W. Word similarity algorithm based on Tongyici Cilin in semantic web adaptive learning system[J]. Journal of Jilin University(Information Science Edition), 2010, 28(6):602-608.

[2] LV L H, LIANG W W, RAN S Y. A Method for Measuring Word Similarity Based on Cilin[J]. Modern Computer, 2013, 1:3-6.

[3] ZHU X H, MA R C, SUN L, et al. Word Semantic Similarity Computation Based on HowNet and CiLin [J]. Journal of Chinese Information Processiong, 2016, 30(4):29-36.

[4] LIU D D, PENG C, QIAN L H, et al. The Effect of TongYiCi CiLin in Chinese Entity Relation Extraction [J]. Journal of Chinese Information Processiong, 2014, 28(2):91-99.

[5] XU Q, DUAN A G, LI A P, et al. Chinese entity relation extraction based on entity semantic similarity[J]. Journal of Shandong University(Engineering Science), 2015,45(6):07-15.

[6] LI G C, LV L, WANG R B, et al. Semantic Role Labeling Based on TongYiCi CiLin Derived Features [J]. Journal of Chinese Information Processiong, 2016, 30(1):101-108.

[7] WANG D, XIONG S H. Short text classification based on synonymy expansion[J]. Journal of Lanzhou University of Technology, 2015, 4:104-108.

[8] DEERWESTER S, DUMAIS S T, FURNAS G W et al. Indexing by latent semantic analysis[J]. Journal of the American Society for Information Science, 1990, 41(6):391– 407.

[9] SCHÜTZE H. Dimensions of meaning[J]. Proceedings of the 1992 ACM/IEEE Conference on Supercomputing. California:IEEE 1992:787-796

[10] LUND K, BURGESS C. Producing high-dimensional semantic spaces from lexical co-occurrence[J]. Behavior Research Methods, Instruments, & Computers, 1996, 28(2):203–208.

[11] COLLOBERT R, WESTON J. A unified architecture for natural language processing: Deep neural networks with multitask learning[C]// International Conference on Machine Learning. Helsinki:ACM, 2008:160-167.

[12] COLLOBERT R, WESTON J, BOTTOU L, et al. Natural language processing (almost) from scratch[J]. Journal of Machine Learning Research, 2011, 12(1):2493-2537.

[13] TURNEY P D. Domain and function: a dual-space model of semantic relations and compositions[J]. Journal of Artificial Intelligence Research, 2012, 44:533-585.

[14] PENNINGTON J, SOCHER R, MANNING C D. Glove: global vectors for word representation[C]// Conference on Empirical Methods on Natural Language Processing. Doha:Association for Computational Linguistics, 2014:1532-1543.

[15] BARTUSIAK R, AUGUSTYNIAK Ł, KAJDANOWICZ T, et al. WordNet2Vec: corpora agnostic word vectorization Method[J]. Neurocomputing. 2016:1-10.

[16] TISSIER J, GRAVIER C, HABRARD A. Dict2vec: learning word embeddings using lexical dictionaries[C]// Conference on Empirical Methods in Natural Language Processing. Copenhagen: Association for Computational Linguistics, 2017:254–263.

[17] ROTHE S, SCHÜTZE H. AutoExtend: extending word embeddings to embeddings for synsets and lexemes[J]. Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics, 2015:1793-1803.

[18] PANCHENKO A. Best of both worlds: making word sense embeddings interpretable[c]// Edition of the Language Resources and Evaluation Conference. Portorož:ELRA, 2016:2649-2655.

[19] YANG L, SUN M. Improved learning of Chinese word embeddings with semantic knowledge[M]// Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data. Switzerland: Springer, 2015:15-25.

[20] GOIKOETXEA J, SOROA, AGIRRE E. Random walks and neural network language models on knowledge bases[J]. Proceedings of the 2015 Annual Conference of the North American Chapter of the ACL, 2015:1434–1439.

[21] MEI J J. Tongyici Cilin[M]. Shanghai: Shanghai Lexicographic Publishing House, 1983.

[22] HARRIS Z. Distributional structure[J]. WORD, 1954,10(2):146-162.

[23] RUMELHART D E, MCCLELLAND J L. Parallel distributed processing: explorations in the microstructure of cognition(volume 1). Cambridge: MIT, 1986, 1:77–109.

[24] SUN F, GUO J F, LAN Y Y, et al. A survey on distributed word representation[J]. Chinese Journal of Computers, 2016, 39:1-22.

[25] CHOMSKY N. Three models for the description of language[J]. Transactions on information theory(volume IT-2 no.3). Institute of Radio Engineers, 1956:113–124.

[26] YESSENALINA A, CARDIE C. Compositional matrix-space models for sentiment analysis[C]// Conference on Empirical Methods on Natural Language Processing. Edinburgh: Association for Computational Linguistics, 2011:172-182.

[27] SOCHER R, HUVAL B, MANNING C D, et al. Semantic compositionality through recursive matrixvector spaces[C]// Conference on Empirical Methods on Natural Language Processing. Jeju Island:Association for Computational Linguistics, 2012:1201-1211.

[28] GREFENSTETTE E, DINU G, ZHANG Y Z, et al. Multi-step regression learning for compositional distributional semantics[EB/OL].(2013-01-29)[2018-04-01]. arXiv:1301.6939.

[29] PENNINGTON J, SOCHER R, MANNING C D. Glove: global vectors for word representation[C]// Conference on Empirical Methods on Natural Language Processing. 2014:1532-1543.

[30] FODOR J A, PYLYSHYN Z W. Connectionism and cognitive architecture: a critical analysis[J]. Cognition, 1988, 28(1/2):3–71.

[31] GERSHMAN S, TENENBAUM J B. Phrase similarity in humans and machines[C]// Proceedings of the 37th Annual Conference of the Cognitive Science Society. Cambridge:MIT, 2015:776-781.

[32] VAKULENKO S. (2005). The notion of sememe in the work of Adolf Noreen[J]. Henry Sweet Society for the History of Linguistic Ideas Bulletin. 2005(44):19–35.

[33] LYONS J. Linguistic Semantics[M]. Cambridge: Cambridge University Press, 1996.

[34] MIKOLOV T, YIH W T, ZWEIG G. Linguistic regularities in continuous space word representations[C]// Proceeding of the 2013 Conference of the North American Chapter of the ACL. Atlanta:Association for Computational Linguistics, 2013:746-751.

[35] CHEN X, XU L, LIU Z, et al. Joint learning of character and word embeddings[C]// Proceedings of IJCAI. Buenos Aires:AAAI, 2015:1236-1242

[36] GE B, LI F F, GUO S L, et al. Word’s semantic similarity computation method based on Hownet[J]. Application Research of Computers, 2010, 27(9):3329-3333.

[37] SHI J, WU Y F, QIU L K, et al. Chinese Lexical Semantic Similarity Computing Based on Large-scale Corpus[J]. Journal of Chinese Information Processiong, 2013, 27(1):1-6.

[38] LI Y, BANDAR Z A, MCLEAN D. An approach for measuring semantic similarity between words using multiple information sources[J]. IEEE Transactions on Knowledge and Data Engineering, 2003, 15(4):871-882.

[39] MEI L J, ZHOU Q, ZANG L, et al. Merge Information in HowNet and TongYiCi CiLin[J]. Journal of Chinese Information Processiong, 2005, 19(1):64-71.

[40] TAIEB M A H, AOUICHA M B, HAMADOU A B. Ontology-based approach for measuring semantic similarity[J]. Engineering Applications of Artificial Intelligence, 2014, 36:238–261.