PAPER

Sentence Similarity Computational Model Based on Information Content

Hao WU[†a)], *Member and* **Heyan HUANG**[†b)], *Nonmember*

SUMMARY Sentence similarity computation is an increasingly important task in applications of natural language processing such as information retrieval, machine translation, text summarization and so on. From the viewpoint of information theory, the essential attribute of natural language is that the carrier of information and the capacity of information can be measured by information content which is already successfully used for word similarity computation in simple ways. Existing sentence similarity methods don't emphasize the information contained by the sentence, and the complicated models they employ often need using empirical parameters or training parameters. This paper presents a fully unsupervised computational model of sentence semantic similarity. It is also a simply and straightforward model that neither needs any empirical parameter nor rely on other NLP tools. The method can obtain state-of-the-art experimental results which show that sentence similarity evaluated by the model is closer to human judgment than multiple competing baselines. The paper also tests the proposed model on the influence of external corpus, the performance of various sizes of the semantic net, and the relationship between efficiency and accuracy.

key words: *sentence semantic similarity, information content, inclusion-exclusion principle, natural language processing, information retrieval*

1. Introduction

Sentence similarity is a core and complicated task in natural language processing (NLP). Nowadays it is becoming an increasingly important text-related research hotspot [1]–[4]. Its applications span a multitude of areas, including information retrieval [5], text summarization [6], [7], text classification [8], text reuse detection [9], automatic machine translation evaluation [10], paraphrase recognition [11], Twitter search [12], image retrieval by captions [13], word sense disambiguation [14] and so on. These tasks all rely on a measure of textual semantic similarity. The computation techniques of sentence semantic similarity can also help these applications to improve the effectiveness. For examples, in web page retrieval, the employment of sentence similarity can significantly improve retrieval effectiveness, and in example-based machine translation (EBMT), the performance can also be enhanced by using the techniques of sentence similarity.

Traditional measures for sentence similarity are adapted from the methods for long texts (documents) [15], [16]. In such methods, words of sentences are treated as meaningless symbols, the meanings of the words are discarded. These measures can not achieve a desirable effect, for sentences generally have too few words for overlaps thus the information conveyed from sentences becomes important for the similarity calculation between the sentences. To solve this problem, hybrid measures [17]–[20] have emerged, which use word similarity as an essential part in various means to deal with similarity calculation between sentences. But these complicated models often need training parameters or use of empirical parameters to adjust the similarity scores. A major drawback of these methods is that they are dependent on the training datasets or subjective experiences.

From the viewpoint of information theory, the essential attribute of natural language is that the carrier of information and the capacity of information can be measured by information content (IC) [21]. IC has been successfully used for similarity computation between words [21]–[23] in simple and unsupervised ways. But how to use IC for multiword poses a computational challenge. This paper presents a novel and simple computational model of sentence semantic similarity by using IC. In the model, sentence similarity is measured by the degree of the overlap of information provided by both sentences. The model uses the hierarchy of semantic nets and corpus statistics as external resources, and employs the inclusion-exclusion principle from combinatorics to solve the computational challenge of sentence IC.

This paper makes the following foremost contributions:

- It presents a simple computational model for sentence semantic similarity by using IC which is mostly used for word similarity computation, and introduces inclusion-exclusion principle in combinatorial mathematics to overcome the computational challenge.
- It proves the model outperforms other four excellent hybrid methods through the experimental results, and finds: (1) to what extent do the databases influence the model, including the kind of external corpus and the size of the semantic net (2) the model can achieve consistent results in larger datasets, and (3) the relationship between the accuracy and the efficiency of the model.

2. Related Work

Pioneer methods on similarity computation between sen-

Manuscript received November 24, 2015.
Manuscript publicized March 14, 2016.

[†]The authors are with Beijing Engineering Research Center of High Volume Language Information Processing and Cloud Computing Applications, School of Computer Science, Beijing Institute of Technology, Beijing, 10081, China.
a) E-mail: wuhao123@bit.edu.cn
b) E-mail: hhy63@bit.edu.cn (Corresponding author)

DOI: 10.1587/transinf.2015EDP7474

tences or very short texts are based on adaptations of similarity measures between documents or long texts. Such work can roughly be categorized into word co-occurrence measures, TF-IDF measures and corpus based measures. The first two methods are easy to understand, while well-known corpus based similarity method is latent semantic analysis (LSA), which is actually a vector method. The aforementioned measures work well for long texts thanks to the adequate words for manipulation. When computing the similarity between sentences that are typically 10-20 words long, experimental results show that these methods are less suitable. Another major reason is that these methods treat words of sentences as meaningless symbols, which are not accurate as word co-occurrence may be much rare. In addition, vector methods in high-dimensional orthogonal space require no similarities between words in each dimension. That is the contradiction with common sense.

To change the unsuitability of similarity between sentences or very short texts, hybrid methods emerge. These methods use more than one method to compute sentence similarity. Li et al. [17] present an algorithm that takes account of semantic information and word order information implied in the sentences. They form the word vector dynamically based entirely on the words in the compared sentences instead of high-dimensional space. The semantic similarity of two sentences is calculated using information from a structured lexical database and from corpus statistics. The use of a lexical database enables their method to model human common sense knowledge and the incorporation of corpus statistics allows the method to be adaptable to different domains. Liu et al. [18] take into account the semantic information, word order, the contribution of different parts of speech in a sentence, and use Dynamic Time Warping (DTW) which is a speech recognition technique. Islam and Inkpen [19] present a method for measuring the semantic similarity of texts using a corpus-based measure of semantic word similarity and a normalized and modified version of the Longest Common Subsequence (LCS) string matching algorithm. Oliva et al. [20] captures and combines syntactic and semantic information to compute the semantic similarity of two sentences. Semantic information is obtained from a lexical database. Syntactic information is obtained through a deep parsing process that finds the phrases in each sentence. Psychological plausibility is added to the method by using previous findings about how humans weight different syntactic roles when computing semantic similarity.

Although hybrid methods have achieved good experimental results on the test set, they have the limitation that all the above measures don't treat the information contained by the sentences as the kernel attribute of natural language from the viewpoint of information theory. That results in most of the measures need training parameters on the corresponding dataset or use empirical parameters. Our model addresses the limitation of above methods by using IC as the central factor, which is successfully used to compute the similarity between words [21]–[23], and establishing the definition of sentence similarity on the basis of the principle of Jaccard Coefficient. Different from hybrid methods, our model can obtain state-of-the-art experimental result without training parameters, and our straightforward measure don't need tools of natural language processing tools such as part of speech tagging, syntactic analysis, word sense disambiguation and so on, all of which may add median error of the similarities.

3. The Proposed Method

The proposed method derives sentence similarity from the overlap of IC contained in the compared sentences. A sentence is considered to be the capacity of IC. The words employing their senses make a sentence convey a specific meaning.

Figure 1 illustrates the process for sentence semantic similarity computation between sentences. Different from existing methods that use empirical parameters or train parameters on the training set, we directly use sentence information content to measure sentence semantic similarity. IC of sentences can be calculated respectively by making use of lexical database and the corpus. We subsequently obtain the intersection of IC of the candidate sentences. Finally, the sentence similarity is derived by the proportion of the intersection and the union of IC. The following sections describe an detailed procedure of the model.

3.1 Semantic Information Space

To illustrate the proposed model, we select a segment from semantic nets (see Fig. 2 left one), and obtain concept relations among concepts (see Fig. 3 left one). To simplify the description of the problem, Fig. 3 omits entity, man, woman concepts.

From the viewpoint of information theory, information is used to eliminate the uncertainty. The more top-level the concept, the greater the uncertainty and the less information provided. For example, if sentences expresses the information of male or female, it must imply the person information. That is, from the perspective of semantic information rela-

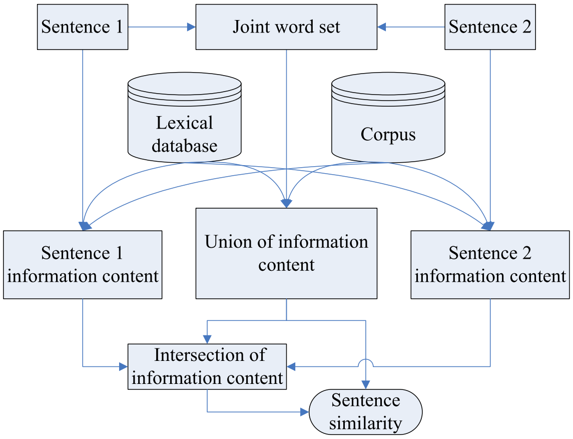
Sentence 1
Joint word set
Sentence 2
Lexical database
Corpus
Sentence 1 information content
Union of information content
Sentence 2 information content
Intersection of information content
Sentence similarity

Fig. 1 Sentence similarity computation diagram.

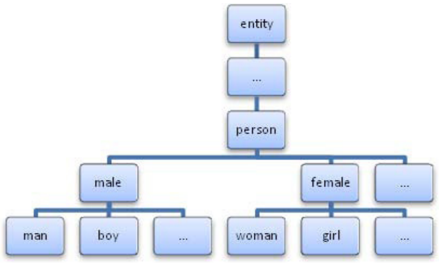

Fig. 2 Hierarchical semantic knowledge base.

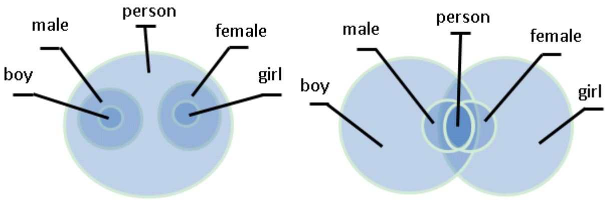

Fig. 3 Relations among concepts: the left shows concepts inclusion, and the right shows information inclusion in SIS.

tions, person is contained by male and female respectively. Thus, we obtain the relationship between the semantic information (see Fig. 3 right one). The space which uses IC to describe the spatial relationship is that we called Semantic Information Space (SIS). SIS isn't a traditional space which uses orthogonality multidimensional to construct, while it applies the inclusion relationship of the information to represent.

3.2 The Intersection of Information among Concepts

Following the standard argumentation of information theory, Resnik [21] proposed the definition information content (IC) of a concept as follows:

$$IC\,(c) = -\log P(c), \tag{1}$$

where $P(c)$ refers to statistical frequency of concept c. Here we convert concept c to specific space in SIS, and the size of the space can be calculated by Eq. (1).

IC of a word is derived from its probability in a corpus (see Sect. 4.2 for details). Quantifying IC in this way makes intuitive sense: as probability increases, informativeness decreases, so the more abstract a concept, the lower its IC. In addition, if there is a unique top concept, its IC is 0. We use IC to measure the quantity of the information of concepts.

We define the quantity of common information of two concepts, that is, the size of the intersection of c_1 and c_2 in SIS is as follows:

$$\begin{aligned} &commonIC\,(c_1, c_2) = IC\,(c_1 \cap c_2) \\ &= \max_{c \in subsum(c_1, c_2)} [-\log P(c)], \end{aligned} \tag{2}$$

where $subsum\,(c_1, c_2)$ is the set of concepts that subsume both c_1 and c_2 in the hierarchy of semantic nets in cases of multiple inheritance. Concepts in this set is called subsumers. $sim\,(c_1, c_2)$ is also equal to the maximum IC of all the subsumers of c_1 and c_2.

We extensionally define the quantity of common information of n-concepts in SIS:

$$\begin{aligned} &commonIC\,(c_1, c_2, \cdots, c_n) = IC\,(c_1 \cap \cdots \cap c_n) \\ &= \max_{c \in subsum(c_1, \cdots, c_n)} [-\log P(c)], \end{aligned} \tag{3}$$

where $subsum\,(c_1, \cdots, c_2)$ is the set of the concepts which subsume all the concepts of $c_1, \cdots c_2$.

Specially, when n is 1, Eq. (3) becomes IC of a single concept:

$$commonIC\,(c_1) = \max_{c \in c_1} [-\log P(c)] = IC\,(c_1)\,. \tag{4}$$

3.3 Semantic Similarity between Sentences

First, we describe the sentence of a, s_a, in SIS as follows:

$$s_a = \left\{ \mathrm{c_i^a} | \mathrm{i} = 1, 2, \ldots, \mathrm{n}; \mathrm{n} = |\mathrm{s_a}| \right\},$$

where $\mathrm{c_i^a}$ is the concept of the i-th word in s_a, $|\mathrm{s_a}|$ is the word count of s_a.

In SIS, each concept belongs to a specific part of the space, the size of the space can be also measured by IC. The total space contained by two concepts is either larger than each concept (The subsumer is not one of these two concepts), or equal to the concept with larger one (The subsumer is one of these two concepts). In general, the space contained by the two concepts should have the intersection unless the subsumer of the concepts is the single root in the semantic nets. Similarly, we can calculate the total space size among multiple concepts by making use of the inclusion-exclusion principle from combinatorics, that is, the union of the space among concepts can be obtained through the intersection of them. And the quantity of the information provided by s_a is:

$$\begin{aligned} &IC\,(s_a) = IC\left(\bigcup_{i=1}^{n} c_i^a\right) \\ &= \sum_{k=1}^{n} (-1)^{k-1} \sum_{1 \le i_1 < \cdots < i_k \le n} IC\left(c_{i_1}^a \cap \cdots \cap c_{i_k}^a\right), \end{aligned} \tag{5}$$

where n is the word count in s_a.

Similarly, the quantity of the information provided by s_b can be deduced by substituting a for b in Eq. (5).

We join all the words from both s_a and s_b into a set, and the set is regarded as the new sentence which includes all information of the two sentences. The total amount information provided by s_a and s_b is as follows:

$$\begin{aligned} &IC\,(s_a \bigcup s_b) = IC\left(\bigcup_{t=a,b}\left(\bigcup_{i=1}^{n_t} c_i^t\right)\right) \\ &= \sum_{k=1}^{n} (-1)^{k-1} \sum_{1 \le i_1 < \cdots < i_k \le k} IC\left(c_{i_1}^a \cap \cdots \cap c_{i_k}^b\right), \end{aligned} \tag{6}$$

where n_t is the word count in sentence t, and n is $n_a + n_b$.

The total space of all concepts in each sentence is regarded as a whole, and the quantity of the intersection of the

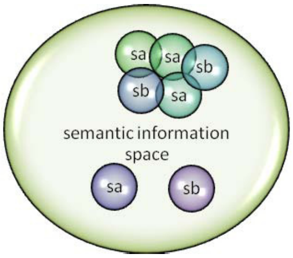

Fig. 4 The relationship of semantic information between two sentences.

information provided by the two sentences is as follows:

$$IC\,(s_a \cap s_b) = IC\,(s_a) + IC\,(s_b) - IC\,(s_a \cup s_b)\,. \tag{7}$$

Finally, we use IC to define the similarity of two sentences based on the principle of Jaccard Coefficient [24]:

$$sim\,(s_a, s_b) = \frac{IC\,(s_a \cap s_b)}{IC\,(s_a \cup s_b)}. \tag{8}$$

In Fig. 4, s_a and s_b present the concepts of sentence a and sentence b respectively. From the figure, we can intuitionally obtain the meaning of the similarity we defined. That is, sentence sematic similarity is the degree of the intersection of semantic information provided by both sentences.

4. Implementation

In this section, we briefly describe the databases used in the model, how to obtain the probability of a concept, and how to select the concept of a word in the sentence.

4.1 The Databases

We use two kinds of databases to implement the computation of IC. One is WordNet [25] and the other is a corpus.

WordNet is a large lexical database of English developed in Princeton University. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. The most frequently encoded relation among synsets is the super-subordinate relation (IS-A relation). All noun hierarchies ultimately go up to the root node entity. This paper uses the IS-A relation of nouns in WordNet to build SIS and compute IC of words. The version of WordNet used is 2.1 for Windows. It includes 104,855 in all nodes (synsets) which contains single words and compound words, most of which are nominal nodes (81,426).

The corpus is used to calculate the frequency of occurrence of words, and then to compute IC along with WordNet. Although the proposed model requires only one corpus, in order to test corpus influence on the model, we compare 5 different corpora widely used in NLP from Natural Language Toolkit (NLTK) [26]: British National Corpus (BNC), Penn Treebank (Treebank), Brown Corpus (Brown), Complete Works of Shakespeare (Shaks) and SemCor Semantically Tagged Corpus (SemCor). Table 1 shows the contained annotation types in each corpus.

Table 1 Tagging types contained in each corpus.

	Automatic POS	Manual POS	Syntax	Semantics
BNC	x			
Treebank		x	x	
Brown	x			
Shaks				
SemCor		x		x

Table 2 The percentage of corpus words absent from WordNet.

	BNC	Treebank	Brown	Shaks	Semcor
Consider Frequency	8.3%	6.5%	6.8%	11.1%	4.9%
Desert Frequency	66.1%	29.1%	20.6%	23.9%	5.2%

There some words in target corpora did not exist in the hierarchy of the semantic net. Table 2 specifies the statistical value of the percentage of corpus words absent from WordNet. The first line of Table 2 considers the word frequency of the corpora and second line deserts it. From Table 2, we can see the word balance between WordNet and the corresponding corpus.

4.2 Obtaining the Probability of a Concept

We use 4 different counting schemes to compute the frequency of a word in a corpus as NLTK. They are standard counting (SC), standard counting with adding 1 smoothing (SC+1), Resnik counting (RC), and Resnik counting with adding 1 smoothing (RC+1).

SC means each concept receives a full count for each word types associated with it. RC denotes each concept associated with a word type receives an equal share of each count. For instance, if there are two senses of a word n, then when we observe n in a corpus, each of the concepts associated with each sense is updated by 0.5 in RC, while each of the two concepts will receive a count of 1 in SC. Word count of different counting schemes is as follows:

$$count(n) = \begin{cases} freq(n) & SC, SC+1 \\ freq(n)/sense(n) & RC, RC+1 \end{cases}, \tag{9}$$

where $freq(n)$ is the frequency of the word n in the corpus, and $sense(n)$ is the quantity of concepts/synsets that contain the word n in WordNet.

We use the method of Resnik [21] to define $P(c)$, the probability of a concept/synset c, for SC and RC as follows:

$$P\,(c) = \frac{\sum_{n \in words(c)} count(n)}{N}, \tag{10}$$

where $words\,(c)$ is the set of all the words contained in concept c and its sub-concepts in WordNet, N is the sum of frequencies all the concepts in the hierarchy of semantic net, which the frequency of one concept is the sum of all the words contained in the concept. For SC+1 and RC+1, each concept in WordNet starts with a count of 1 instead of 0. Table 3 shows the quantitative value of N in each corpus with each counting scheme from NLTK.

Table 3 The quantitative value of N in each corpus with each counting scheme.

	SC	SC+1	RC	RC+1
BNC	179,806,392	179,911,247	48,213,107	48,317,962
Treebank	2,214,723	2,319,578	611,392	716,247
Brown	1,829,957	1,934,812	485,236	590,091
Shaks	1,759,108	1,863,963	478,050	582,905
SemCor	359,821	464,676	93,668	198,523

4.3 Choosing the Concept of a Word

In practice, people need to measure similarity of sentences which are represented by words rather than concepts, so there is a realistic demand that realize the word/concept conversions. However, nearly half of words in WordNet are polysemous (i.e., a word having multiple meanings), more than one path may exist between the two words. Traditional similarity measures, such as Li et al. [17], usually use an alterable concept of a word to calculate max similarity. We use the certain concept of a word, so the results obtained by our method are more reliable and can be explained and understood.

Another difference is the way to calculate word similarity. Traditional methods use superficial factors, such as the shortest path in WordNet, to calculate max semantic similarity between words. From the perspective of information theory, the path length and subsumer depth are only some surface layer influence factors, however, the IC provided by common part of words are the essential.

The way we choose the concept of a word is as follows:

$$c_1 = \mathop{\arg\max}_{\substack{c \in subsume(c_1, c_2) \\ c_1 \in concept(word_1) \\ c_2 \in concepts(sentence_2)}} \{IC(c)\}, \tag{11}$$

where $concept(word_1)$ is the concept set of $word_1$ from $sentence_1$, and $concepts(sentence_2)$ is the concept set of all the words from $sentence_2$.

5. Experiments

Sentence semantic similarity is a subjective concept. Usually it can only reference to the subjective judgment of human. To evaluate our model, we use datasets of Li et al. [17] as benchmarks set. The average word number is 13.2 in sentence pairs. We design four experiments to test our model in this paper. The first experiment compares our model with the other four excellent hybrid methods. The last three experiments focus on the model itself.

5.1 Compared with Other Methods

In this experiment, BNC is used for our model based on the idea of max word counts in the five corpora, and standard counting (SC) is as our counting scheme described in Sect. 4. Relevant evaluations employ Pearson correlation coefficient[†] (PCC, Pearson's r), Spearman rank correlation coefficient[††] (SRCC, Spearman's ρ) and Kendall rank correlation coefficient[†††] (KRCC, Kendall's τ). Table 4 details the similarity scores of each sentence pair obtained from the mean of human ratings, the four benchmarks, and our measure.

Table 4 The similarity of 30 pairs of sentence scores by different measures (range of 0.0-1.0).

No.Sentence Pair	Human Ratings	Li's	Liu's	Islam's	Oliva's	Ours
1.Cord-smile	.01	.33	.03	.06	.32	.27
5.Autograph-shore	.01	.29	.00	.11	.28	.18
9.Asylum-fruit	.01	.21	.00	.07	.27	.05
13.Boy-rooster	.11	.53	.12	.16	.27	.22
17.Coast-forest	.13	.36	.02	.26	.42	.19
21.Boy-sage	.04	.51	.14	.16	.37	.12
25.Forest-graveyard	.07	.55	.18	.33	.53	.21
29.Bird-woodland	.01	.33	.01	.12	.31	.09
33.Hill-woodland	.15	.59	.47	.29	.43	.26
37.Magician-oracle	.13	.44	.05	.20	.23	.15
41.Oracle-sage	.28	.43	.16	.09	.38	.21
47.Furnace-stove	.35	.72	.06	.30	.24	.18
48.Magician-wizard	.36	.65	.22	.34	.42	.34
49.Hill-mound	.29	.74	.45	.15	.39	.42
50.Cord-string	.47	.68	.16	.49	.35	.25
51.Glass-tumbler	.14	.65	.16	.28	.31	.19
52.Grin-smile	.49	.49	.18	.32	.54	.30
53.Serf-slave	.48	.39	.18	.44	.52	.34
54.Journey-voyage	.36	.52	.19	.41	.33	.18
55.Autograph-signature	.41	.55	.33	.19	.33	.28
56.Coast-shore	.59	.76	.46	.47	.43	.51
57.Forest-woodland	.63	.70	.39	.26	.50	.75
58.Implement-tool	.59	.75	.34	.51	.64	.40
59.Cock-rooster	.86	1.0	.85	.94	1.0	1.0
60.Boy-lad	.58	.66	.69	.60	.63	.55
61.Cushion-pillow	.52	.66	.45	.29	.39	.40
62.Cemetery-graveyard	.77	.73	.65	.51	.75	.57
63.Automobile-car	.56	.64	.38	.52	.78	.39
64.Midday-noon	.96	1.0	1.0	.93	1.0	1.0
65.Gem-jewel	.65	.83	.60	.65	.36	.39

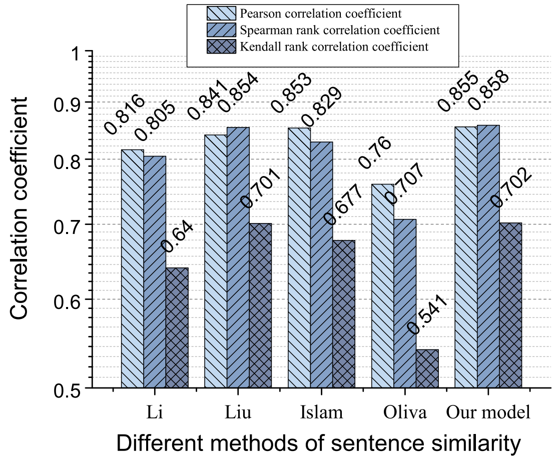

Fig. 5 Correlation coefficient between human ratings and program results by different sentence similarity measures.

From Fig. 5 we can see our model outperform other four measures on PCC, SRCC and KRCC. This means sentence similarities gained from the model are closer to human subjective judgments whether on the linear correlation or on

[†] http://en.wikipedia.org/wiki/Pearson_product-moment _correlation _coefficient

[††] http://en.wikipedia.org/wiki/Spearman's_rank_correlation _coefficient

[†††] http://en.wikipedia.org/wiki/Kendall_tau_rank_correlation _coefficient

Table 5 PCC on different corpora and counting schemes.

	SC	**SC+1**	**RC**	**RC+1**	**av.**
BNC	.854	.854	.842	.843	.848
Treebank	.860	.852	.829	.845	.847
Brown	.857	.841	.842	.838	.845
Shaks	.845	.853	.832	.842	.843
SemCor	**.861**	.840	.852	.841	**.849**
av.	**.855**	.848	.839	.842	

Table 6 SRCC on different corpora and counting schemes.

	SC	SC+1	RC	RC+1	av.
BNC	.858	.858	.842	.848	.852
Treebank	**.877**	.872	.858	.870	**.869**
Brown	.870	.860	.860	.847	.859
Shaks	.853	.874	.825	.863	.854
SemCor	.855	.853	.841	.848	.849
av.	**.863**	**.863**	.845	.855	

Table 7 KRCC on different corpora and counting schemes.

	SC	**SC+1**	**RC**	**RC+1**	**av.**
BNC	.702	.702	.700	.700	.701
Treebank	**.733**	.730	.710	.730	**.726**
Brown	.719	.713	.713	.706	.713
Shaks	.697	.728	.669	.716	.703
SemCor	.696	.696	.678	.697	.692
av.	.709	**.714**	.694	.710	

Table 8 PCC influenced by the size of semantic net.

WordNet Proportion	**Synset Amount**	**Least N**	**PCC**
10%	8143	5532	.257
15%	12213	2053	.794
20%	16284	895	.834
25%	20362	417	.861
30%	24432	213	.853
50%	40795	23	.858

Table 9 Performs on the training set (range of 0.0-4.0).

No.Sentence Pair	**Human Ratings**	**Ours**	**No.Sentence Pair**	**Human Ratings**	**Ours**
2.Rooster-voyage	.02	.07	3.Noon-string	.05	.06
4.Fruit-furnace	.19	.16	6.Automobile-wizard	.08	.08
7.Mound-stove	.02	.15	8.Grin-implement	.02	.13
10.Asylum-monk	.15	.06	11.Graveyard-madhouse	.09	.25
12.Glass-magician	.03	.09	14.Cushion-jewel	.21	.11
15.Monk-slave	.18	.18	16.Asylum-cemetery	.15	.08
18.Grin-lad	.05	.07	19.Shore-woodland	.33	.19
20.Monk-oracle	.45	.09	22.Automobile-cushion	.08	.16
23.Mound-shore	.14	.11	24.Lad-wizard	.13	.11
26.Food-rooster	.22	.32	27.Cemetery-woodland	.15	.31
28.Shore-voyage	.08	.10	30.Coast-hill	.40	.39
32.Crane-rooster	.08	.14	31.Furnace-implement	.20	.09
34.Car-journey	.29	.10	35.Cemetery-mound	.23	.14
36.Glass-jewel	.43	.16	38.Crane-implement	.74	.07
39.Brother-lad	.51	.29	40.Sage-wizard	.61	.11
42.Bird-crane	.14	.25	43.Bird-cock	.65	.21
44.Food-fruit	.97	.22	45.Brother-monk	.18	.46
46.Asylum-madhouse	.86	.18			

rank correlation. In addition, Li et al. [17] also claims that the average PCC between scores by a single volunteer and all volunteers is 0.825 on his data set, and the max PCC between them is 0.921. Our experimental PCC score is higher than average PCC and lower than highest PCC, and this suggests that judgement ability on sentence similarity by our model is above most people abilities, and that the score is below the upper limit indicates our model is reliable.

5.2 The Model Influenced by the Databases

This section we focus on the model influenced by the external corpus and the size of the semantic net.

In the aspect of the external corpus, we use 5 English corpora and each with 4 different word counting schemes (see Sect. 4.1 and 4.2 for details). Standard deviations of the scores in Tables 5, 6, and 7 are 0.01, 0.01 and 0.02, respectively, which manifests the model has a small dependence on frequently-used corpora, and the method could achieve consistent results in various corpora. From Tables 5, 6, and 7, we can see the following phenomena:

- The highest PCC score are obtained by SemCor with SC, which corpus has the minimum size and the best manual semantic tagging for WordNet word sense.
- All highest rank related correlation coefficient (SRCC and KRCC) scores are gained by using Treebank with SC, which corpus owns the better balance of the size and the high quality manual tagging.
- Both Treebank and SemCor have manual tagging from Table 1, and lower percentages of corpus word absent from WordNet considered the word frequency of the corpora from Table 2.

From corpus tagging perspective alone, we may conclude that the manual annotation of the corpus is extremely important for IC computation. From word counting schemes angle alone, SC gets the highest scores while RC acquires the worst on all correlation coefficients. RC assumes that that the each word sense has an equal probability of occurrence is not real.

In the aspect of the semantic net, we test the performance of the model influenced by the size of the semantic net. We resize WordNet according to N (see Eq. (10) for reference), the sum of frequencies of all the concepts in the hierarchy of semantic net, based on the following consideration: The lower frequency of a concept, the smaller possibility of construction in a semantic net. The corpus used here is BNC based on the idea of real distribution of words. The proportion of WordNet synsets, synset amount, the least N in BNC and PCC are listed in Table 8.

From Table 8, we may deduce: The least synset amount in the semantic net should be more than 12 thousand (PCC score above 0.8 is considered highly linear correlation); to achieve desirable results, the minimal semantic net should contain 20 thousand concepts, a quarter size of WordNet 2.1.

5.3 Experiments on More Data

To see whether the model can achieve consistent results, we test the model on all the 65 sentence pairs, including extra 35 sentence pairs. Table 9 shows the rating scores of human and the model on the extra 35 sentence. Note that similarity scores are in the range of 0.0-4.0 in Table 9 as the original literature. From the definition of PCC, we know the normal-

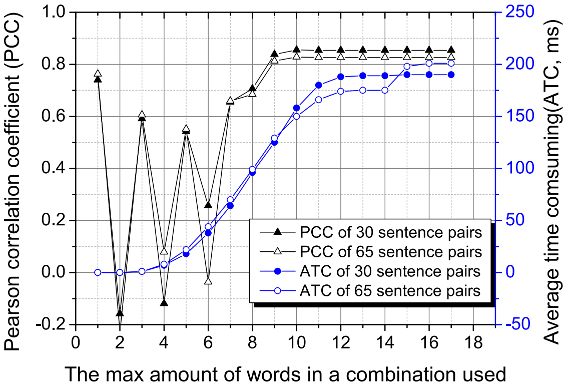

Fig. 6 Relations between PCC and ATC in the model.

ization of scores does not affect PCC results.

In Table 9, there are 27 sentence pairs that human ratings are less than 0.1 (normalization to the range of 0.0-1.0), and this accounts for 77% of the total 35 extra sentence pairs. Excessive concentration to lower similarity scope increases bias in the frequency distribution, and makes it hard achieve a good PCC. The model obtains the PCC of 0.826 for all 65 sentence pairs. Results decrease less than 0.03 compared to 30 sentence pairs, which is still higher than the baseline of 0.816 (PCC obtained by Li et al. [17]). This shows our method can achieve consistent results, and be applied on a larger range of dataset to compute sentence similarity.

5.4 Accuracy and Efficiency

The model does not need to train parameters, and the complexity is mainly from the algorithm of computing the union of IC. From Eq. (5) and Eq. (6), we can see the model needs to compute all IC of the word combinations from one word to all words in sentence pairs. The amount of the combinations to be computed is as follows:

$$C(n,1) + C(n,2) + \cdots + C(n,n) = 2^n - 1 \qquad (12)$$

where n is the amount of the words in the sentence pair, $C(n,1)$ is the number of 1-combinations from n-words. The amount of combinations is huge when n increases to some extent. We want to decrease the amount of combinations to compute. As we know, when the amount of words in a combination increases, the intersection of IC of these words decreases. In this experiment, we test the relation between accuracy of PCC and efficiency of time consuming, by decreasing the amount of combinations to compute. The average length of all sentence pairs is 28.2, and the max length of them is 54. BNC with SC is used.

Figure 6 illustrates the following conclusion: When the max amount of words in a combination is 1, the PCC (0.740 for 30 sentence pairs and 0.76 for 65) nearly reaches the score gained by Oliva et al. [20] (0.76 for 30), and the ATC is nearly zero (The minimum time interval obtained from the operating system is about 30ms, the ATC is less than it). When the max amount is 9, the PCC (0.838 for 30 and 0.813 for 65) exceeds the method of Li et al. [17] (baseline, 0.816 for 30), and the ATC is reduced to about 65% of the total ATC. When the max amount is 10, the PCC reaches the maximum value, and the ATC is reduced by about 20%.

6. Conclusion

This paper presented a model for computing the semantic similarity between sentences or short texts, based on information content. First, we use a simple method to choose the concept of each word in each sentence pair as a preprocessing procedure. Second, the IC of a concept and the common IC among multi-concepts are derived from a lexical knowledge base and a corpus. Third, the model applies inclusion-exclusion principle in combinatorial mathematics to obtain the IC of each sentence and the joint sentence of the two, and sentence similarity can be computed by information overlap between the compared sentences. To certify our model, we develop four experiments: the first experiment shows sentence similarity calculated by our model is a more significant correlation to human intuition than multiple competing baselines. The other three experiments test the proposed model on the influence of external corpus, the performance of various sizes of the semantic net, the adaptability to a larger database, and the relationship between efficiency and accuracy. In addition, the model is simple, straightforward and fully unsupervised, needs neither any parameter nor other NLP tools.

Further work will include using word sense disambiguation as a preprocessing procedure to improve the accuracy of the algorithm. Also, we will try to further improve the model on the efficiency for longer sentences on other datasets.

Acknowledgments

The work described in this paper was mainly supported by State Key Program of National Natural Science Foundation of China under Grant 61132009 and National Programs for Fundamental Research and Development of China (973 Program) under Grant 2013CB329303.

The authors would like to thank the editors and the anonymous reviewers for their insightful comments to the improvement in accuracy and integrity.

References

[1] E. Agirre, M. Diab, D. Cer, and A. Gonzalez-Agirre, "Semeval-2012 task 6: A pilot on semantic textual similarity," Proceedings of the First Joint Conference on Lexical and Computational Semantics-Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation, pp.385–393, Association for Computational Linguistics, 2012.

[2] E. Agirre, D. Cer, M. Diab, A. Gonzalez-Agirre, and W. Guo, "sem 2013 shared task: Semantic textual similarity, including a pilot on typed-similarity," In* SEM 2013: The Second Joint Conference on

Lexical and Computational Semantics. Association for Computational Linguistics, Citeseer, 2013.

[3] E. Agirre, C. Banea, C. Cardie, D. Cer, M. Diab, A. Gonzalez-Agirre, W. Guo, R. Mihalcea, G. Rigau, and J. Wiebe, "Semeval-2014 task 10: Multilingual semantic textual similarity," Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pp.81–91, 2014.

[4] E. Agirre, C. Banea, C. Cardie, D. Cer, M. Diab, A. Gonzalez-Agirre, W. Guo, I. Lopez-Gazpio, M. Maritxalar, R. Mihalcea, G. Rigau, L. Uria, and J. Wiebe, "Semeval-2015 task 2: Semantic textual similarity, english, spanish and pilot on interpretability," Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015), pp.252–263, June 2015.

[5] E.-K. Park, D.-Y. Ra, and M.-G. Jang, "Techniques for improving web retrieval effectiveness," Information processing & management, vol.41, no.5, pp.1207–1223, 2005.

[6] C.-Y. Lin and E. Hovy, "Automatic evaluation of summaries using n-gram co-occurrence statistics," Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1, pp.71–78, Association for Computational Linguistics, 2003.

[7] R.M. Aliguliyev, "A new sentence similarity measure and sentence based extractive technique for automatic text summarization," Expert Systems with Applications, vol.36, no.4, pp.7764–7772, 2009.

[8] F. Sebastiani, "Machine learning in automated text categorization," ACM computing surveys (CSUR), vol.34, no.1, pp.1–47, 2002.

[9] P. Clough, R. Gaizauskas, S.S.L. Piao, and Y. Wilks, "Meter: Measuring text reuse," Proceedings of the 40th Annual Meeting on Association for Computational Linguistics, pp.152–159, Association for Computational Linguistics, 2002.

[10] D. Kauchak and R. Barzilay, "Paraphrasing for automatic evaluation," Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics, pp.455–462, Association for Computational Linguistics, 2006.

[11] B. Dolan, C. Quirk, and C. Brockett, "Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources," Proceedings of the 20th international conference on Computational Linguistics, p.350, Association for Computational Linguistics, 2004.

[12] B. Sriram, D. Fuhry, E. Demir, H. Ferhatosmanoglu, and M. Demirbas, "Short text classification in twitter to improve information filtering," Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval, pp.841–842, ACM, 2010.

[13] T.A.S. Coelho, P.P. Calado, L.V. Souza, B. Ribeiro-Neto, and R. Muntz, "Image retrieval using multiple evidence ranking," IEEE Trans. Knowl. Data Eng., vol.16, no.4, pp.408–417, 2004.

[14] H. Schütze, "Automatic word sense discrimination," Computational Linguistics, vol.24, no.1, pp.97–123, 1998.

[15] G. Salton, A. Wong, and C.S. Yang, "A vector space model for automatic indexing," Communications of the ACM, vol.18, no.11, pp.613–620, 1975.

[16] A.G. Maguitman, F. Menczer, H. Roinestad, and A. Vespignani, "Algorithmic detection of semantic similarity," Proceedings of the 14th international conference on World Wide Web, pp.107–116, ACM, 2005.

[17] Y. Li, D. McLean, Z.A. Bandar, J.D. O'shea, and K. Crockett, "Sentence similarity based on semantic nets and corpus statistics," IEEE Trans. Knowl. Data Eng., vol.18, no.8, pp.1138–1150, 2006.

[18] X. Liu, Y. Zhou, and R. Zheng, "Sentence similarity based on dynamic time warping," Semantic Computing, 2007. ICSC 2007. International Conference on, pp.250–256, IEEE, 2007.

[19] A. Islam and D. Inkpen, "Semantic text similarity using corpus-based word similarity and string similarity," ACM Transactions on Knowledge Discovery from Data (TKDD), vol.2, no.2, p.10, 2008.

[20] J. Oliva, J.I. Serrano, M.D. del Castillo, and Á. Iglesias, "Symss: A syntax-based measure for short-text semantic similarity," Data & Knowledge Engineering, vol.70, no.4, pp.390–405, 2011.

[21] P. Resnik, "Using information content to evaluate semantic similarity in a taxonomy," International Joint Conference on Artificial Intelligence (IJCAI), 1995.

[22] J.J. Jiang and D.W. Conrath, "Semantic similarity based on corpus statistics and lexical taxonomy," Proceedings of International Conference Research on Computational Linguistics (ROCLING X), 1997.

[23] D. Lin, "Using syntactic dependency as local context to resolve word sense ambiguity," Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics, pp.64–71, Association for Computational Linguistics, 1997.

[24] P. Jaccard, Nouvelles recherches sur la distribution florale, 1908.

[25] G.A. Miller, "Wordnet: a lexical database for english," Communications of the ACM, vol.38, no.11, pp.39–41, 1995.

[26] S. Bird, "Nltk: the natural language toolkit," Proceedings of the COLING/ACL on Interactive presentation sessions, pp.69–72, Association for Computational Linguistics, 2006.

Hao Wu received the master degree in Computer Science from the Beijing Institute of Technology. He is currently a doctoral candidate in the School of Computer Science at the Beijing Institute of Technology, and he is also an experimentalist in the School of Computer Science at the Beijing Institute of Technology. His research interests include sentence similarity, natural language processing and machine learning. He has involved in research projects and foundations including National Natural Science Foundation of China (NSFC), National Basic Research Program of China (973 Program), etc.

Heyan Huang is currently Professor and Dean of School of Computer Science and Technology in Beijing Institute of Technology of China. She received her PhD degree in Computer Science and Technology in 1989 from Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. Her current research interests include machine learning, information retrieval, and natural language processing. She has published over 90 research papers in reputed journals and conferences, such as Pacific Asia Conference on Language, Information and Computation, China Communications, and Journal of Software. She serves on the editorial boards of International Journal of Advanced Intelligence and Journal of Computer Research and Development. She has undertaked 20 more research projects including National 863 Project of China, National 973 Project of China, National Natural Science Foundation of China, etc.