JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 21, Number 1, 2014

Pp. 80–88
DOI: 10.1089/cmb.2012.0009

Interaction Relation Ontology Learning

CHUAN-XI LI,[1,3,4] RU-JING WANG,[3,4] PENG CHEN,[2]
HE HUANG,[3] and YA-RU SU[3,4]

ABSTRACT

Ontology is widely used in semantic computing and reasoning, and various biomedicine ontologies have become institutionalized to make the heterogeneous knowledge computationally amenable. Relation words, especially verbs, play an important role when describing the interaction between biological entities in molecular function, biological process, and cellular component; however, comprehensive research and analysis are still lacking. In this article, we propose an automatic method to build interaction relation ontology by investigating relation verbs, analyzing the syntactic relation of PubMed abstracts to perform relation vocabulary expansion, and integrating WordNet into our method to construct the hierarchy of relation vocabulary. Five attributes are populated automatically for each word in interaction relation ontology. As a result, the interaction relation ontology is constructed; it contains a total of 963 words and covers the most relation words used in existing methods of proteins interaction relation.

Key words: interaction relation, ontology learning, relation words, text mining.

1. INTRODUCTION

Ontology is widely used for semantic computing and reasoning, and various biomedicine ontologies have become institutionalized to make the heterogeneous knowledge computationally amenable. As building ontology by domain experts manually is labor-intensive and hard scalable, learning-based methods are proposed naturally. Bodenreider and Stevens (2006) explored the application of ontologies in bioinformatics and medical informatics, described the gene ontology and the OBOization of bio-ontology according to the spectrum of genotype to phenotype, and presented some possible directions of the bio-ontology process. OntoUSP (Poon and Domingos, 2010) induced and populated a probabilistic ontology by parsing the syntactic structure of a sentence, which learns the ISA hierarchy by clusters of logical expression. Liu et al. (2011) reviewed and discussed the existing methods of biomedical ontology learning from free texts, which concern natural language processing, information extraction, and machine learning.

In hierarchy learning of ontology, Biemann (2005) presented a survey of learning ontology or ontology-like structures from unstructured text, and the comparisons of a clustering-based method and a classification-based method were analyzed in detail. Lin (1998) defined word similarity based on the distribution pattern

[1]National Science Library, Chinese Academy of Sciences, Beijing, P.R. China.
[2]Institute of Health Sciences, Anhui University, Hefei, Anhui 230601, China.
[3]Institute of Intelligent Machines, Chinese Academy of Sciences, Hefei, P.R. China.
[4]School of Information Science and Technology, University of Science and Technology of China, Hefei, P.R. China.

by parsing the dependency relation of sentences, and found similar words by different similarity measures. Our hierarchy of words is partly inspired by those works. The word class methods, N-gram model and statistical algorithm, were discussed by Brown et al. (1992). Caraballo (1999) created the semantic lexicon and its hierarchy of nouns based on heuristic pattern. Cimiano et al. (2004) learned the taxonomies automatically by concept clustering based on Formal Concept Analysis, compared it with hierarchical agglomerative clustering and hierarchical divisive clustering, and represented the hypernym as a description label of the abstract concept by a verb-like identifier. Ushioda (1996) presented a greedy algorithm that tries to minimize average loss of mutual information of adjacent classes for hierarchical clustering of words using large texts, and pointed out that the hierarchy learning methods based on clustering lack significant class labels of clustering results.

Relation words, especially verbs, play important roles when describing the interaction between biological entities (protein, gene, etc.). The work of Rebholz-Schuhmann et al. (2010) provided a survey on the relation verbs, which was referred to by different research teams, and described the prediction capacity of different verbs for protein interaction extraction on the existing corpora. Observed by the online service iHOP (Information Hyperlinked Over Proteins) (Hoffmann and Valencia, 2005), about 90% of active relationships of proteins were expressed syntactically as ''protein verb protein,'' highlighting the importance of interaction verbs at relation navigation networks. Levin (1993) classified over 3,000 English verbs in a lexicon according to the rule with shared meaning and behavior, which assumed that the meaning of the verb influences its syntactic behavior, and then integrated it into a powerful tool. VerbNet (Schuler, 2005; Schuler et al., 2009), the largest on-line hierarchical verb lexicon, contains explicit syntactic and semantic information for classes of verbs with mappings to other lexical resources such as WordNet, Xta, and FrameNet. Each verb class in VerbNet is completely described by thematic roles, selectional restrictions of the arguments. Differentiating with the open verb research, we aim at interaction relation verbs between biological entities and extract them from PubMed abstracts, which locate the verbs in coordination relation, and filter the irrelevant by mutual information.

Syntactic patterns were widely exploited in information extraction. The entity relation oriented open domain was involved in Open Information Extraction (Etzioni et al., 2008, 2011) and StatSnowBall (Zhu et al., 2009) on large-scale corpora. By analyzing and summarizing the syntactic patterns, Etzioni et al. employed a rule-based method to extract relations and arguments. In our method, the syntactic patterns were used to explore the sentences of PubMed abstracts that contain candidate interaction relation words. The purpose of our work lies in exploring and organizing the verbs that express an interaction relation between biomedical entities. We present an interaction relation ontology learning method that consists of relation lexicon learning and hierarchy relation learning. In the first step, relation lexicon learning, the syntactic patterns are used to construct query expressions to search PubMed abstracts for retrieving the relevant sentences. Then syntactic analysis is performed to extract the candidate interaction relation words, and these words are assembled into the relation lexicon. In the second step, hierarchy relation learning, the words in the relation lexicon are organized by combing the WordNet hierarchy to form the interaction relation ontology; meanwhile, five attributes are populated for each word automatically. The meaning of the lower-level words in the relation ontology is more general than those in the higher level, and thus the lower the level in which the word is located, the more specific the relations expressed between biological entities.

2. METHOD

In this article, we explore the relation verbs and propose an automatic method to build interaction relation ontology, which consists of two steps, relation lexicon learning and hierarchy relation learning. The first step extracts relation verbs from the sentences of PubMed abstracts (www.ncbi.nlm.nih.gov/pubmed/) by analyzing the coordination relation of syntactic structure and then from a relation lexicon. The second one builds the hierarchy of relation lexicon by integrating the WordNet. Finally, five attributes of the relation word in interaction relation ontology are populated automatically.

2.1. Relation lexicon learning

This step performs the expansion of seed relation verbs from PubMed abstracts and constructs the relation lexicon, in which the seed verbs are selected from a particular corpus manually. At first, query expressions are submitted to PubMed for retrieving the relevant sentences. Then syntactic parsing of the sentence is

Table 1. The Selected Coordination Relations in Stanford Typed Dependencies

conj_and	conj_but	conj_or	conj_negcc	conj_plus	conj_yet
conj_nor	conj_less	conj_so	conj_of	conj_so	conj_as
conj_only	conj_+	conj_et	conj_vs	conj_plus	

performed to discover the coordination relation of the seed words. The iterative procedure terminates when either the iterative count reaches a threshold or the number of new generated words is less than a threshold.

Coordination constructions (Haspelmath, 2004) in linguistics can be categorized based on their meaning: conjunctive, additive, coordinative, cumulative (''and''), disjunctive (''or''), adversative (''but''). In our method, the syntactic pattern ''verb + coordination word'' is taken to capture coordination relation, which forms the query expressions such as ''seed word + and,'' ''seed word + or,'' and ''seed word + but'' of the PubMed search, and extracts 200 sentences for each query submission. As there may be repeated retrieved sentences, we eliminate the redundancy by using Levenshtein distance for the efficiency of sentence parsing. The Stanford parser (de Marnee and Manning, 2010) provided syntactic analysis for analyzing the coordination words of the seed words. We select *typed dependencies with collapsed* as our parsing style and collect 17 expression styles of coordination relation as listed in Table 1.

As an example, for the sentence ''*Here we report that in vivo IκBβ serves both to inhibit and facilitate the inflammatory response*,'' the following typed dependencies are obtained by the Stanford parser: ''*dep(report-3, Here-1); dep(report-3, we-2); root(ROOT-0, report-3); nsubj(serves-8, report-3); nn(IκBβ-7, vivo-6); prep_in(serves-8, IκBβ-7); rcmod(report-3, serves-8); dep(inhibit-11, both-9); aux(inhibit-11, to-10); ccomp(serves-8, inhibit-11); xcomp(serves-8, inhibit-11); ccomp(serves-8, facilitate-13); xcomp(serves-8, facilitate-13); conj_and(inhibit-11, facilitate-13); det(response-16, the-14); amod(response-16, inflammatory-15); dobj(inhibit-11, response-16).*'' As shown in Table 1, the selected typed dependencies of coordination relations contain ''*conj_and(inhibit-11, facilitate-13)*,'' in which the word ''*facilitate*'' can be further retrieved as a candidate relation word.

The obtained coordination words are lemmatized by MorphAdorner (morphadorner.northwestern.edu/), and words with POS adjective or adverb in this stage are discarded. If one word has the same root verb or can be used as verb, its root verb or itself is regarded as a candidate word; otherwise, discard it. In addition, the stop words (snowball.tartarus.org/algorithms/english/stop.txt) do not express interaction relation generally and are also removed. Algorithm 1 shows the pseudo codes of the relation lexicon learning. In lines 5 to 16, the candidate relation words are retrieved by querying PubMed abstracts and analyzing the coordination relation of seed words. In lines 17 to 22, the point-wise mutual information between seed relation words and the candidate relation words is computed based on the typed dependencies according to Formula 1, as used by Hindle (1990). In lines 23 and 24, new retrieved words are taken as seed words instead of the original seed words, and the process is repeated until the number of new generated words is less than a threshold or the loop count reaches a specified threshold.

Formula 1. Point-wise mutual information:

$$I(x;y) = log(p(x,y)/(p(x)^{*}p(y))).\ x, y \in \ relation\ words$$

Algorithm 1. Relation Lexicon Learning

Input: Seed relation words *seedRelationWordSet,* the parameter of mutual information threshold t_{MI}, iterative number threshold t_{iter}, and the number of new candidate words threshold t_{new}
Output: Relation Lexicon *RelationLexicon*
Function relationLexiconLearning(*seedRelationWordSet,* t_{MI}, $t_{iter,}$ t_{new})

1. *typedDependenciesSet = empty;candidateRelationWordSet = empty;newRelationWordSet = empty;*
2. *iterativeCount* = 0;
3. *Lexicon*.add(*seedRelationWordSet*);
4. Do
5. For each word in *seedRelationWordSet*
6. *typedDependenciesSet = empty; SentenceSet = empty;*
7. Form the query expressions of the word *queryExpressions*
8. For each *queryExpression* in *queryExpressions*

```
9.      Submit the queryExpressions to PubMed search and add the relevant sentences into SentenceSet
10.   Remove the redundant sentences of SentenceSet with Levenshtein distance
11.   Parse the sentence typed dependencies in SentenceSet and add it into typedDependenciesSet
12.   Retrieve words in coordination relation of typed dependencies and add it into candidateRelationWordSet
13.   Filter out adjective, adverb from candidateRelationWordSet
14.   Change the noun to its verb form if exist, or else discard the noun
15.   Remove the stop word from candidateRelationWordSet
16. End for
17. For each srw in seedRelationWordSet
18.   For each crw in candidateRelationWordSet
19.     Calculate the point-wise mutual information of word pair I(srw;crw)
20.     If (I(srw;crw) > t_MI) add the crw into newRelationWordSet
21.   End for
22. End for
23. RelationLexicon.add(newRelationWordSet);
24. seedRelationWordSet = newRelationWordSet;
25. iterativeCount = iterativeCount + 1;
26. While( newRelationWordSet.size < t_new or iterativeCount < t_n)
27. Return RelationLexicon;
End function
```

2.2. Hierarchy relation learning

The hierarchy of relation ontology is constructed with integrating WordNet into our method, and the pseudo codes are described in Algorithm 2. The root node of relation ontology is set to *RelationEntity* in line 1. From line 2 to line 7, we construct the children of root node. At first, we search for the nodes whose parents do not exist in the lexicon by querying WordNet and add them as child nodes of root node, which form the second layer of the ontology. However, if the word parents exist in the relation lexicon and the distance between the word and its parents is less than a threshold t_{dist}, the word to the nodes of ontology is added as child correspondingly, which is performed from line 8 to line 14. From line 15 to line 24, the remaining words of the lexicon are regarded as child nodes that have the shortest distance with them. Here, if two or more nodes of the ontology have the same shortest distance with the word, the word is added as child of all of them. In the case of the word *act*, its parents do not exist in the relation lexicon, so it is added as child of the root node *RelationWords*. For example, the word *interact* has the same distance 0.3334 with *intervene* and *treat* in the lexicon; therefore, it is regarded as parent of *intervene* and *treat*.

Algorithm 2. Hierarchy Relation Learning

```
Input: Relation Lexicon RelationLexicon, RiTa.WordNet, and Words distance threshold t_dist
Output: Relation ontology RO
Function HierarchyRelationLearning(RelationLexicon, RiTa)
1.  RO.root = RelationWords
2.  For rw in RelationLexicon
3.    If (RiTa .getParents(rw) ∩ RelationLexicon) = = empty)
4.      RO.root.addChild(rw);
5.      RW.remove(rw);
6.    End if
7.  End for
8.  For rw in RelationLexicon
9.    If (RiTa. getParents(rw) ∩ RelationLexicon) is not empty)
10.     If(RiTa.getDistance(rw.parents,RO.getWord()) < t_dist)
11.       RO.getWord.addChild(rw);
12.       RelationLexicon.remove(rw);
13.   End if
14. End for
15. For rw in RelationLexicon
16.   minimizeDistance = 1;
17.   For ro in RO
```

```
18.     If(RiTa.getDistance(rw, ro) < minimizeDistance)
19.         word = ro;
20.     End if
21.   End for
22.   ro.addChildren(word);
23.   RW.remove(word);
24. End for
End function
```

The distance of two words is calculated as shown below. First, locate the common parent *cp* of two words with lemmatization. If it exists, check each sense of each lemma; otherwise, return 1, which means that their distance is immeasurable. Second, calculate the shortest path from either lemma to *cp*, *minDistToCommonParent*. Third, calculate the length of the path from *cp* to the root node of the WordNet, *distFromCommonParentToRoot*. Finally, the distance of two words is shown in Formula 2.

Formula 2.

$$\text{normalizedDistToCommonParent} = \frac{\text{minDistToCommonParent}}{\text{distFromCommonParentToRoot} + \text{minDistToCommonParent}}$$

2.3. Entity attributes population

After finishing the hierarchy learning of relation ontology, attributes of the relation words are created, which contain *wordContext*, *distanceWithParent*, *nounForm*, *presentParticiple*, and *pastParticiple*. The value of the attribute *wordContext* refers to the number of sentences in which the word occurs, which gives the instances of its usage. The value of the attribute *distanceWithParent* is the distance with its parent, which indicates the strength between them. Moreover, the attributes *nounForm*, *presentParticiple*, and *pastParticiple* populate the noun, present participle, and past participle of it, respectively, and the word form transformation is carried by MorphAdorner. A pseudo code of the attributes populating is described in Algorithm 3.

Algorithm 3. Attributes populating of relation ontology

```
Input: Relation ontology RO, RiTa.WordNet, MorphAdorner
Output: Relation ontology RO
Function AttributesPopulating (RO, RiTa)
 1.  For all entity in RO
 2.    If(entity.root is root node)
 3.      entity.setAttribute(distanceWithParent) = 0;
 4.    Else
 5.      entity.setAttribute(distanceWithParent) = RiTa.getDistance(entity, entity.parent);
 6.    End if
 7.  End for
 8.  For all entity in RO
 9.    entity.setAttribute(nounForm) = entity.NounForm;
10.    entity.setAttribute(wordContext) = sentences containing the entity extracted from PubMed;
11.    entity.setAttribute(presentParticiple) = MorphAdorner.getPresentParticiple(entity) ;
12.    entity.setAttribute(pastParticiple) = MorphAdorner.getPastParticiple(entity) ;
13.  End for
End function
```

3. EXPERIMENTS EVALUATION

3.1 Relation lexicon learning

The relation verbs of protein interaction are extracted and summarized from the work of Chowdhary et al. (2009) as seed words. The total seed words retrieved from Chowdhary contain 293 words. After

Table 2. Seed Relation Words

acetylate	carbamoylated	dehydrate	inhibit	up-regulate	transactivate
accept	cleave	down-regulate	interact	repress	modulate
activate	conjugate	ethylate	methylate	increase	
associate	deacetylate	formylate	phosphorylate	promote	
attach	deaminate	heterodimer	regulate	stimulate	
bind	decarboxylate	homodimer	transaminate	disassemble	

lemmatization and normalization, 32 words are selected as shown in Table 2. In Algorithm 1, the parameters of mutual information threshold t_{MI}, iterative threshold t_{iter}, and the threshold number of new candidate words t_{new} are empirically set to 0.05, 10, and 10, respectively.

In Algorithm 1, the terminal conditions of the iterative contain the iterative threshold number and the threshold number of new candidate words. After four iterations, the algorithm terminates as the number of new generated words is less than the threshold t_{new}. The experimental results for four interactions are given in Table 3. Each row represents the results of one iteration. The column *Generated candidate words* represents the number of the generated coordination words without any processing, as described in lines 13 to 15 in Algorithm 1. The column *Processed words* represents the number of words after lemmatization and normalization. As a result, the learned relation lexicon contains 963 verbs.

The performance of the sentence parser influences the extraction of coordination relation words, but this is not always the case for some incorrect parsing. For example, in the sentence ''*Specifically, despite receiving the same mechanical perturbation causing muscle stretch, the strongest responses were produced when the contralateral arm was perturbed in the opposite direction (large tray tilt) rather than in the same direction or not perturbed at all*,'' two typed dependencies of coordination relation are generated by the Stanford parser, that is, ''*conj_or(perturbed-23, not-40)*'' and ''*conj_negcc(direction-27, direction-38)*,'' and the two extracted candidate relation words, ''*not*'' and ''*direction*,'' are all discarded in the next step in that the word ''*not*'' does not have a corresponding verb form and the word ''*direction*'' is a duplicate.

3.2. Hierarchy relation learning and attribute populating

Relation ontology hierarchy learning forms the hierarchy of relation lexicon by combining the WordNet hierarchy into Algorithm 2. The parameter t_{dist} that controls the distance between biological entities is set to 0.5. Five attributes are set automatically in experiments. The attribute *wordContext* populates the sentences in which the word occurs in the top five sentences of PubMed abstracts. The attribute *distanceWithParent* populates the distance between the word and its parent, which is calculated by Formula 2. The attributes *nounForm*, *presentParticiple*, and *pastParticiple* populate the noun, the present participle, and the past participle of the verb word by MorphAdorner. The snippet of the constructed relation ontology is generated by protégé (protege.stanford.edu) and presented in Figure 1. As an example, the five attributes of the relation word *intervene* are presented in Figure 2, in which the attribute *wordContext* can have more than one value.

3.3. Coverage evaluation

We compare our relation ontology with the protein interaction relation words that are extracted from corpora BioInfer (Pyysalo et al., 2007), BioCreAtIvE-PPI (Plake et al., 2005), LLL05 (Hakenberg et al., 2005), Hakenberg (Hakenberg et al., 2006), RelEx (Fundel et al., 2007), Temkin (Temkin and Gilder, 2003), and Kabiljo (Kabiljo et al., 2009), in which the singular and plural of verb and noun are ignored. As in Table 4, the columns *Extracted relation words, Ignored,* and *Recall* represent the total number of

Table 3. Generated Words of Each Iteration

	Generated candidate words	*Processed words*
First iteration	2,027	442
Second iteration	5,093	486
Third iteration	1,454	35
Fourth iteration	279	7

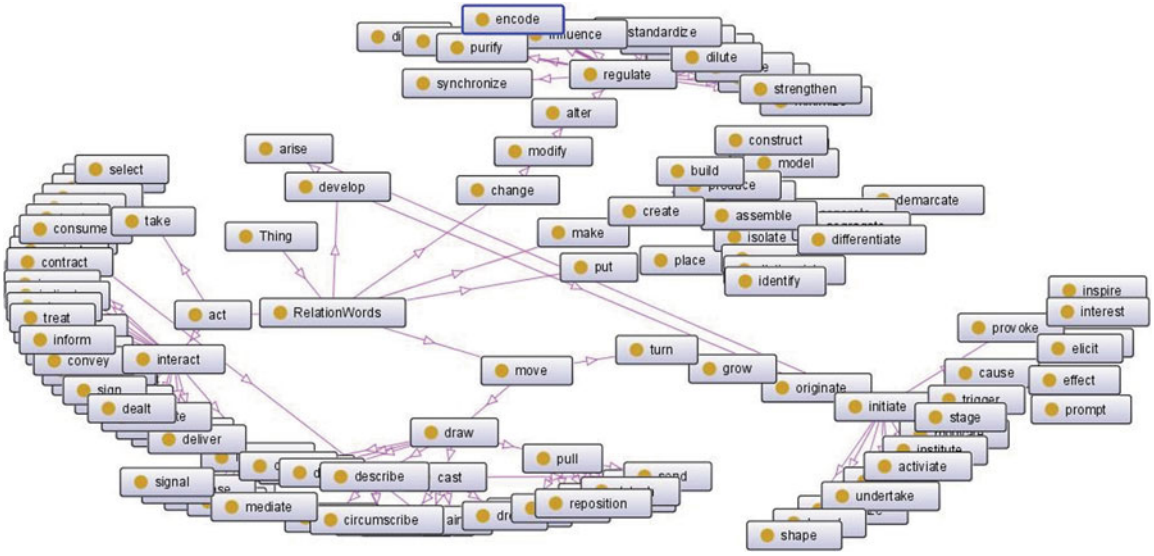

FIG. 1. Interaction relation ontology snippet.

extracted relation words, the number of omitted words by our method, and the recall of our ontology that is computed in Formula 3, respectively.

Formula 3. $Recall = (Extracted\ relation\ words - Error)/Extracted\ relation\ words$

In Table 4, the constructed relation ontology can cover most of the relation words of the selected corpora. In BioInfer, the compound word ''*CROSS-LINK-AP*'' is missed, because we do not consider this word style. The words ''*ligand-independent*'' and ''*formalin-fixed*'' in BioCreAtIvE are missed for the same reason in BioInfer, and the word ''*enhancer's*'' is ignored as we could not lemmatize the word correctly. Adjective and adverb are discarded in our ontology, which induces the words ''*misc*,'' ''*actively*,'' and ''*inducible*'' missed in LLL05. In corpus (Hakenberg et al., 2006), all words can be covered by our ontology, except for ''*see*,'' ''*be*,'' and ''*give, make*,'' which are filtered out as stop words, after they are lemmatized and presented in the style of past participle. In RelEx, the lemmatization of 175 words is presented, and all the words are captured except for ''*ligand*'' and ''*use*'' when lemmatizing the word in our ontology, where ''*ligand*'' as a noun without corresponding verb and ''*use*'' as stop words are ignored. The words used in Temkin (Temkin and Gilder, 2003) and Kabiljo (Kabiljo et al., 2009) are all covered by our ontology. As the discussed hierarchy in WordNet, the meaning of the lower-level word in the relation ontology is more general than that of the corresponding higher-level one, *i.e.*, the lower the level where the

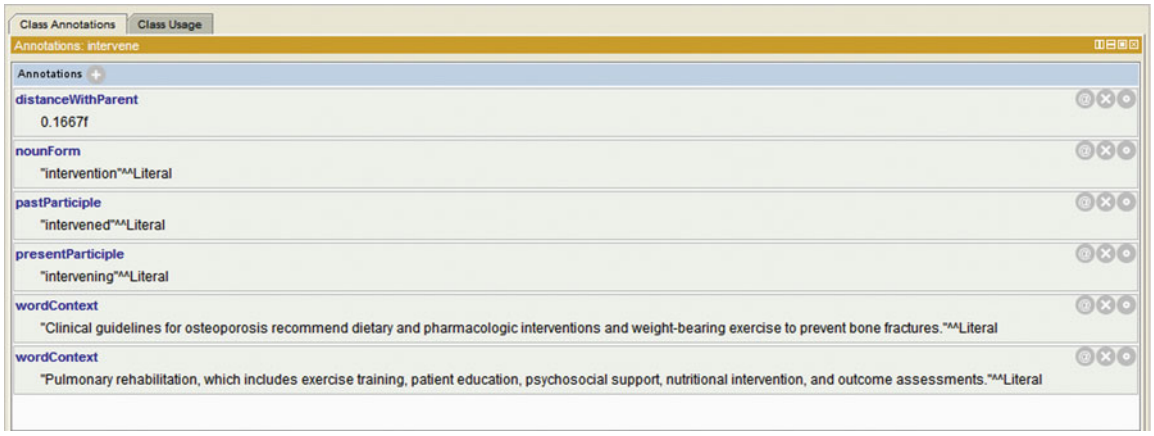

FIG. 2. The attributes of relation word *intervene*.

Table 4. The Performance of Relation Ontology

	Extracted relation words	*Ignored*	*Recall*
BioInfer	34	1	0.9706
BioCreAtIvE-PPI	158	3	0.9810
LLL05	102	3	0.9706
Hakenberg	419	4	0.9905
RelEx	175	2	0.9886
Temkin	72	0	1
Kabiljo	110	0	1

word is located, the more specific the relations expression between biological entities the word has. For instance, the word *mediate* is used to express the interaction relation more frequently and specifically than its parent *draw* in experiments.

4. CONCLUSIONS AND FUTURE WORKS

In this article, we propose an automatic method to build an interaction relation ontology, which explores the relation words that represent the interaction between biological entities, by analyzing the syntactic relation of PubMed sentences to learn the vocabulary. The lexicon hierarchy is formed with the help of WordNet, whose word position is determined by the distance between the word and its parents. The constructed interaction relation ontology contains a total of 963 verbs. By experimental evaluation, the ontology covers the most relation words used in existing methods of protein interaction relation extraction. On the ontology hierarchy, the words in the lower level tend to express relations between biological entities more specific than those in the higher level. At the same time, five attributes, *wordContext*, *distanceWithParent*, *nounForm, presenParticiple,* and *pastParticiple,* are populated automatically. In the future, we will explore the detailed usage of the interaction words in a particular aspect of molecular function, biological process, and cellular component.

ACKNOWLEDGMENTS

This research was supported by the National Natural Science Foundation of China (Nos. 61300058 and 61374181). This work was also supported in part by the National Key Technology R&D Program under grant no. 2011BAH10B03.

DISCLOSURE STATEMENT

The authors declare that no competing interests exist.

REFERENCES

Biemann, C. 2005. Ontology learning from text: a survey of methods. *LDV Forum* 20, 75–93.

Bodenreider, O., and Stevens, R. 2006. Bio-ontologies: current trends and future directions. *Brief. Bioinform.* 7, 256–274.

Brown, P.F., deSouza, P.V., Mercer, R.L., et al. 1992. Class-based n-gram models of natural language. *Comput. Linguist.* 18, 467–479.

Caraballo, S.A. 1999. Automatic construction of a hypernym-labeled noun hierarchy from text. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics on Computational Linguistics*. Association for Computational Linguistics, College Park, MD; pp. 120–126.

Chowdhary, R., Zhang, J., and Liu, J.S. 2009. Bayesian inference of protein–protein interactions from biological literature. *Bioinformatics* 25, 1536–1542.

Cimiano, P., Hotho, A., and Staab, S. 2004. Comparing conceptual, divisive and agglomerative clustering for learning taxonomies from text. In *Proceedings of the European Conference of Artificial Intelligence*.

de Marnee, M.-C., and Manning, C. 2010. Stanford typed dependencies manual. Available at nlp.stanford.edu/downloads/dependencies_manual.pdf. Accessed Novemeber 27, 2013.

Etzioni, O., Banko, M., Soderland, S., and Weld, D.S. 2008. Open information extraction from the web. *Commun. ACM* 51, 68–74.

Etzioni, O., Fader, A., Christensen, J., et al. 2011. Open information extraction: the second ceneration. International Joint Conference on Artificial Intelligence.

Fundel, K., Kuffner, R., and Zimmer, R. 2007. RelEx—relation extraction using dependency parse trees. *Bioinformatics* 23, 365–371.

Hakenberg, J., Plake, C., Leser, U., et al. 2005. LLL'05 Challenge: genic interaction extraction with alignments and finite state automata. In *Proceedings of Learning Language in Logic Workshop (LLL05) at the 22nd International Conference on Machine Learning*. 38–45.

Hakenberg, J., Leser, U., Kirsch, H., and Schuhmann, D. 2006. Collecting a large corpus from all of Medline. In *Second International Symposium on Semantic Mining in Biomedicine, SMBM*.

Haspelmath, M. 2004. *Coordinating Constructions*. J. Benjamins Publishing, Amsterdam, The Netherlands.

Hindle, D. 1990. Noun classification from predicate-argument structures. In *Proceedings of the 28th Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, Pittsburgh, PA; pp. 268–275.

Hoffmann, R., and Valencia, A. 2005. Implementing the iHOP concept for navigation of biomedical literature. *Bioinformatics* 21 Suppl 2, ii252–ii258.

Kabiljo, R., Clegg, A., and Shepherd, A. 2009. A realistic assessment of methods for extracting gene/protein interactions from free text. *BMC Bioinformatics* 10, 233.

Levin, B. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago, IL.

Lin, D. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics—Volume 2*. Association for Computational Linguistics, Montreal, Quebec, Canada; pp. 768–774.

Liu, K., Hogan, W.R., and Crowley, R.S. 2011. Natural language processing methods and systems for biomedical ontology learning. *J. Biomed. Inform*. 44, 163–179.

Plake, C., Hakenberg, J., and Leser, U. 2005. Optimizing syntax patterns for discovering protein–protein interactions. In *Proceedings of the 2005 ACM Symposium on Applied Computing*. ACM, Santa Fe, NM; pp. 195–201.

Poon, H., and Domingos, P. 2010. Unsupervised ontology induction from text. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Uppsala, Sweden; pp. 296–305.

Pyysalo, S., Ginter, F., Heimonen, J., et al. 2007. BioInfer: a corpus for information extraction in the biomedical domain. *BMC Bioinformatics* 8, 50.

Rebholz-Schuhmann, D., Jimeno-Yepes, A., Arregui, M., and Kirsch, H. 2010. Measuring prediction capacity of individual verbs for the identification of protein interactions. *J. Biomed. Inform.* 43, 200–207.

Schuler, K.K. 2005. *VerbNet: A Broad-Coverage, Comprehensive Verb Lexicon*. University of Pennsylvania, Philadelphia, PA; p. 146.

Schuler, K.K., Korhonen, A., and Brown, S. 2009. VerbNet overview, extensions, mappings and applications. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Tutorial Abstracts*. Association for Computational Linguistics, Boulder, CO; pp. 13–14.

Temkin, J.M., and Gilder, M.R. 2003. Extraction of protein interaction information from unstructured text using a context-free grammar. *Bioinformatics* 19, 2046–2053.

Ushioda, A. 1996. Hierarchical clustering of words. In *Proceedings of the 16th Conference on Computational Linguistics—Volume 2*. Association for Computational Linguistics, Copenhagen, Denmark; pp. 1159–1162.

Zhu, J., Nie, Z., Liu, X., et al. 2009. StatSnowball: a statistical approach to extracting entity relationships. In *Proceedings of the 18th International Conference on World Wide Web*. ACM, Madrid, Spain; pp. 101–110.

Address correspondence to:
Dr. Peng Chen
Institute of Health Sciences
Anhui University
No. 111 Jiulong Road
Hefei, Anhui Province
230601, P.R. China

E-mail: bigeagle@mail.ustc.edu.cn