Journal of Biomedical Informatics 56 (2015) 94–102

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin

Identifying synonymy between relational phrases using word embeddings

Nhung T.H. Nguyen [a,b,*,1], Makoto Miwa [c], Yoshimasa Tsuruoka [d], Satoshi Tojo [b]

[a] University of Science, Vietnam National University, Ho Chi Minh City, 227 Nguyen Van Cu St., Ward 4, Dist. 5, Ho Chi Minh City, Viet Nam
[b] Japan Advanced Institute of Science and Technology, 1-8 Asahidai, Nomi-shi, Ishikawa 923-1292, Japan
[c] Toyota Technological Institute, 2-12-1 Hisakata, Tempaku-ku, Nagoya 468-8511, Japan
[d] The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

ARTICLE INFO

Article history:
Received 7 October 2014
Revised 12 May 2015
Accepted 15 May 2015
Available online 22 May 2015

Keywords:
Word embeddings
Synonym resolution
Relational phrase clustering
Topic modeling

ABSTRACT

Many text mining applications in the biomedical domain benefit from automatic clustering of relational phrases into synonymous groups, since it alleviates the problem of spurious mismatches caused by the diversity of natural language expressions. Most of the previous work that has addressed this task of synonymy resolution uses similarity metrics between relational phrases based on textual strings or dependency paths, which, for the most part, ignore the context around the relations. To overcome this shortcoming, we employ a word embedding technique to encode relational phrases. We then apply the *k*-means algorithm on top of the distributional representations to cluster the phrases. Our experimental results show that this approach outperforms state-of-the-art statistical models including latent Dirichlet allocation and Markov logic networks.

1. Introduction

Many of the robust text mining systems in the biomedical domain allow end-users to browse and retrieve information from their databases [1–3]. Implementing such retrieval functionality is usually not so difficult if the system is only concerned with a specific type of information, such as protein–protein interaction and gene-disease association, since they can apply some matching techniques to the input entities to extract the answers. However, the problem becomes much more difficult when the system is designed to cover unrestricted types of relations, which requires the relation in a query to be specified using a natural language expression, such as 'be induced by' or 'result in'. Such relational phrases expressed in natural language often cause spurious mismatches between the user's query and the textual data in the underlining database. For example, given the input query "What genes are essential for cell survival?", the system can fail to return the result <stat1, *be critical for*, cell survival> due to the string-level mismatch between *be essential for* and *be critical for*. In most situations, *be essential for* is equivalent to *be critical for*, i.e., they form a pair of synonyms, which can be used for alleviating the mismatch problem. Therefore, the major objective of this work is to identify synonymy between relational phrases in biomedical relations, which should be beneficial for many text mining applications in the domain, such as question answering, event extraction, and entailment detection [4,5].

Identifying synonymy between relational phrases can be seen as clustering synonymous phrases that represent identical or similar relationships between entities. Since this task is performed on top of a relation extraction system, the performance of clustering can be affected by the performance of the extraction system. Another difficulty of the task is the polysemy of natural language, i.e., a relational phrase can have multiple senses. This problem could be addressed by using a soft clustering approach, but we leave it for future work and assume that a relation phrase belongs to a single cluster.

Previous work that tackled this task employed similarity metrics based on textual strings [6] or dependency paths [7–9] of the two relational phrases. Kok and Domingos [10] proposed a probabilistic model based on two Markov logic networks (MLNs) [11] to simultaneously cluster objects and relations. Nebot and Berlanga [12] used a probabilistic model inspired by statistical machine translation to cluster relations in biomedical documents. These models are unsupervised in the sense that no manual

* Corresponding author at: University of Science, Vietnam National University, Ho Chi Minh City, 227 Nguyen Van Cu St., Ward 4, Dist. 5, Ho Chi Minh City, Viet Nam.

E-mail addresses: nthnhung@jaist.ac.jp (N.T.H. Nguyen), makoto-miwa@toyota-ti.ac.jp (M. Miwa), tsuruoka@logos.t.u-tokyo.ac.jp (Y. Tsuruoka), tojo@jaist.ac.jp (S. Tojo).

[1] This work was carried out while the first author was a doctoral student at JAIST.

http://dx.doi.org/10.1016/j.jbi.2015.05.010
1532-0464/

labeling of clusters by human is needed. One of the major shortcomings of their approaches, however, is that they only focus on the textual surface of arguments of a relation to estimate the synonymy probability and cannot effectively capture other features, such as the context around the relations.

To address the above shortcoming, we apply the continuous bag-of-words (CBOW) model, a deep-learning technique proposed by Mikolov et al. [13], to represent our relational phrases. A relation in the format of <entity 1, relational phrase, entity 2> is identified in a sentence, and each of the two entities and relational phrase is regarded as a newly defined *word*. We thus treat the entities and the phrase differently from the other words depending on their corresponding roles in the relation. The CBOW model then learns the distributional representations of the relational phrases through a feed-forward neural network language model [14], which allows us to capture the context around a relational phrase when learning its representation.

Sun and Korhonen [15] also used the context around verbs for the task of verb classification by introducing a rich set of semantic features. The features include collocations of verbs, prepositional preference, and lexical preference in subject, object and indirect object relations. The key difference between their work and ours is that we cluster verbs and verb phrases that compose biomedical relations while they only focus on single verbs.

We have compared our approach with three unsupervised methods: bag-of-words (BOW), latent Dirichlet allocation (LDA) [16], and Semantic Network Extractor (SNE) [10]. Regarding BOW and LDA, we treat a relational phrase as a *document* (in LDA terms) and entities that share the same phrase as *words* in the document. The BOW model represents each relational phrase as a sparse vector of occurrence counts of entities. LDA-SP [17], which is developed from LinkLDA [18] to model selectional preferences, simultaneously models two sets of distributions for two entities of a relation. Each entity is drawn from a hidden topic. LDA-SP assigns a higher probability to the state in which the two hidden topics are equal. For each relational phrase, the model outputs a vector of the prior topic distribution. We then apply the *k*-means algorithm on top of vector representations to cluster phrases into synonymous groups.

SNE tackles the task of clustering relational phrases by a probabilistic model trained on two MLNs. Unlike the other methods, SNE performs clustering on a database of relations, i.e., it does not consider the context or the frequency of relations. However, SNE can automatically identify the best number of clusters and simultaneously cluster objects and relational phrases.

We have conducted experiments using a large set of biomedical relations extracted from MEDLINE by PASMED, a pattern-based open information extraction (Open IE) system [19,20]. The results show that word embeddings significantly outperform BOW, LDA-SP and SNE. They can boost the performance of clustering by 9% of F-score compared with the other methods. In addition, we demonstrate how the obtained clusters of relational phrases could be used to improve the performance of high-level text-mining applications such as question answering and entailment detection.

The main contribution of this article is that we have applied LDA-SP and CBOW models to the task of identifying synonymy between relational phrases. For the CBOW model, we have introduced a simple but effective representation of relations. Because the representation can exploit various information relevant to relations, e.g., the textual surface of the two entities, the context around a relation in its sentence, and the corresponding role of each component in a relation, the performance of CBOW is boosted significantly.

2. Clustering relational phrases

We first encode our relational phrases into vector format by using three different unsupervised techniques: bag-of-words, topic model and word embeddings. Next, we apply the *k*-means algorithm on top of these vector representations to cluster relational phrases into synonymous groups. In addition to vector representations, we have also employed SNE, a Markov logic network-based system, to identify synonymous relational phrases. An overview of our working flow is shown in Fig. 1.

2.1. Word embeddings

Mikolov et al. [13] introduced two effective techniques for learning vector representations of words from large amounts of unstructured text data: the Continuous Bag-Of-Word (CBOW) model and the continuous Skip-gram model.

The CBOW model is similar to the feed-forward neural network language model [14], where there is no hidden layer and the projection layer is shared for all words. Unlike the BOW model, this model predicts a word by using the continuous context around that word. Given a sequence of training words $w_1, w_2, w_3 \ldots w_T$, the objective of this model [21] is to maximize the average log probability as shown in Eq. (1), where C_t is words in the context of w_t within a window size of c, $C_t = w_{t-c}, w_{t-c+1} \cdots w_{t-1}, w_{t+1} \cdots w_{t+c}$.

$$\frac{1}{T}\sum_{t=1}^{T} \log p(w_t|C_t) \tag{1}$$

The probability of $p(w_t|C_t)$ is estimated by using the softmax function:

$$p(w_t|C_t) = \frac{\exp\left(v'^{\top}_{t} v_{C_t}\right)}{\sum_{i=1}^{V} \exp\left(v'^{\top}_{i} v_{C_t}\right)} \tag{2}$$

where v and v' are the *input* and *output* vector representation of a word w, and V is the number of words in the vocabulary. In contrast with the CBOW model, the Skip-gram model receives the current word and predicts words within a certain window.

Recently, distributed representations have been shown to effectively improve the performance of many NLP tasks such as paraphrase detection [22], sentiment prediction [23], semantic relation classification [24], word alignment [25], entity mention tagging [26], and machine translation [27–29].

In this paper, we use the CBOW model[2] to estimate vector representations of our relational phrases. More specifically, for each relation in the format of <entity 1, relational phrase, entity 2>, which is given by an Open IE system, we retrieve the sentence that contains the relation from the original text database. We then identify the words or phrases that correspond to the entities and relational phrases, and create newly-defined *words* for them depending on their roles in the relation.

We introduce three different representations of a relation:

(i) *Relation*: treating a relation as a sentence, this representation uses the same information as BOW, LDA-SP, and SNE.
(ii) *Sentence*: embedding the relation in the sentence in which it appears and assigning a role to the relational phrase.
(iii) *Role*: embedding the relation in the sentence in which it appears and assigning corresponding roles to the relational phrase and its two entities.

For example, a relation of <parkinson's disease, treat with, dopaminergic drug> will be represented in three ways shown in

[2] The model is implemented in the word2vec tool: http://code.google.com/p/word2vec/.

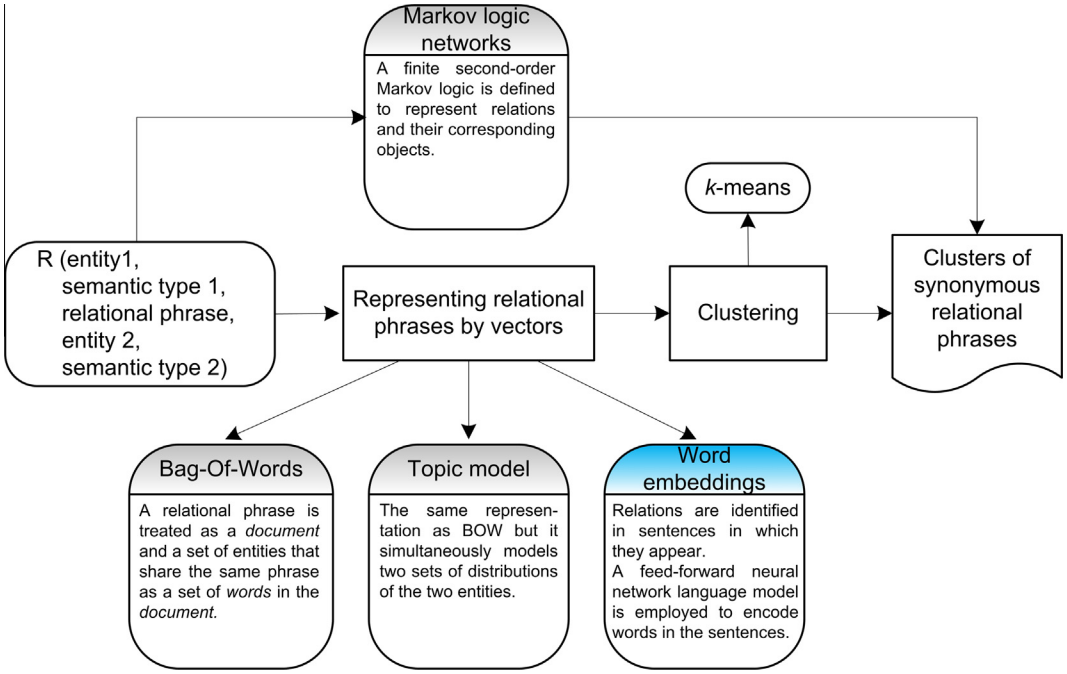

Fig. 1. An overview of our methods.

Table 1
Three ways of modeling a relation of <parkinson's disease, treat with, dopaminergic drug>.

Type	Representation
Relation	"parkinson's_disease treat_with dopaminergic_drug"
Sentence	"many patient with parkinson's_disease be treat_with@pred dopaminergic_drug"
Role	"many patient with parkinson's_disease@arg1 be treat_with@pred dopaminergic_drug@arg2"

Table 1. Note that multi-word terms are grouped with underscores and the roles in the relation are indicated by artificial suffixes such as '@arg1'.

Since each representation has its own definition of words, its corresponding size of vocabulary and the number of words in the training data are different from those by the others, as reported in Table 2. In case of *Relation* representation, the vocabulary and the number of words are substantially lower than the others because the representation does not take into account the context around a relation. Meanwhile, by assigning roles to entities, we increased the size of vocabulary and the number of words in the training data, which means that the data is sparser. In theory, sparse data may substantially affect the running time of the learning process. In our settings, however, the running times of the three representations are comparable.

After training the CBOW model, we extract the distributed feature vector v'_w associated with each relational phrase w and apply k-means clustering to them.

2.2. Bag-of-words model

The bag-of-words (BOW) model is a simple model commonly used in a variety of text processing tasks. In this model, a document is represented simply as a set of words without considering the syntax and even word order. Each word is represented by its index in the vocabulary and its frequency in the document. In our scenario, each relational phrase is treated as a *document* and a set of entities that share the same phrase as a set of *words* in the *document*. Consequently, a relational phrase has two bags of entities for the two corresponding arguments. The relational phrase are thus represented by a sparse vector of occurrence counts of entities, i.e., a sparse histogram over the vocabulary.

Table 2
Vocabulary size and number of words by each representation.

	Relation	*Sentence*	*Role*
Vocabulary size (K)	340	494	653
Number of words (M)	126	268	268

2.3. Topic model

We have employed LDA-SP [17], an extension of the LinkLDA model [18], to model our relations for clustering. LDA-SP considers a relational phrase as a *document*, and a set of entities that share the same relational phrase as a set of *words* in a *document*. The advantage of this model is that it simultaneously models two sets of distributions of the entities for each topic. The graphical representation of LDA-SP is shown in Fig. 2.

In this model, each argument a_i is drawn from a different hidden topic z_i; however, the z_i's are drawn from the same distribution θ_r for a given relation r. LDA-SP allows two arguments of a given relation to be generated from $|Z|^2$ possible pairs. Since z_1 and z_2 are drawn from the same distribution θ_r, the model assigns a higher probability to states in which $z_1 = z_2$. The output of this model is the prior topic distribution of each relational phrase in R. More specifically, a relation phrase r is represented as a vector whose elements are the probabilities that the phrase belongs to a topic $p(r|t)$.

We implemented the LDA-SP model by using collapsed Gibbs sampling [30] for inference.

2.4. Markov logic networks

Based on the output of TextRunner [31], Kok and Domingos [10] built a Semantic Network Extractor (SNE) to detect groups of

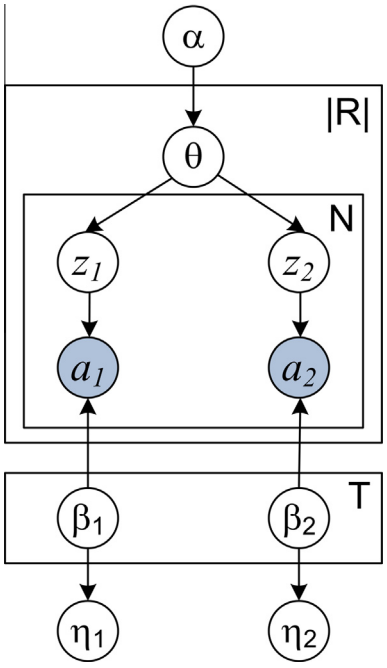

Fig. 2. The graphical representations of the LDA-SP [17] model.

entities and relational phrases of relations. Their model, which is enhanced by two Markov logic networks (MLN), can simultaneously cluster both entities and phrases. The model learns the log-posterior of each cluster assignment Γ as shown in Eq. (3), where R is the set of relations, K is the set of cluster combinations, and t_k and f_k are the empirical numbers of true and false atom in a cluster combination k, respectively.

$$\log P(\Gamma|R) = \sum_{k\in K}\left[t_k \log\left(\frac{t_k+\alpha}{t_k+f_k+\alpha+\beta}\right) + f_k \log\left(\frac{f_k+\beta}{t_k+f_k+\alpha+\beta}\right) - \lambda m_{cc} + \mu d + C\right] \quad (3)$$

Other parameters in Eq. (3) present the following meanings:

- α, β: Smoothing parameters used to estimate the MAP (maximum a posteriori) weight of an instance of the atom prediction rule.
- λ: A weight corresponds to the number of cluster combinations (m_{cc}) being formed.
- μ: A weight accompanies with the number of pairs of symbols that belong to different clusters (d).

3. Evaluation settings

3.1. Data

3.1.1. Training data

Our training data was created by PASMED,[3] a pattern-based Open IE system [20]. PASMED extracts diverse types of binary relations from biomedical literature by using deep syntax patterns. Six predicate-argument structure patterns are applied to the output of Mogura [34], a high-speed version of the Enju parser [35], to extract relevant noun phrase (NP) pairs in a sentence, i.e., two noun phrases that are considered to have a certain relationship. The system then employs MetaMap [36] to locate named entities in the NP pairs. Finally, a relation between two entities in the NP pair is extracted if and only if the pair of semantic types is included in the UMLS Semantic Network.[4] A manual evaluation conducted on 500 randomly selected sentences on MEDLINE has shown that the system gained a mean precision of 47.39%. PASMED has been applied to the 2012 MEDLINE baseline[5] and extracted more than 137 million semantic relations in the format of <relational phrase, entity 1, semantic type 1, entity 2, semantic type 2>.

We have selected a subset of 47 million relations from the PASMED's output to create our training data. More specifically, the training data is the output for the MEDLINE abstracts in a period from 2004 to 2012. Since the output contains many false positive relations, we have cleaned it by removing relations that do NOT satisfy any of the following conditions:

- The relational phrase is a verb or a verb phrase. For example, the relation between 'insulin sensitivity' and 'rats' extracted from the phrase "...the insulin sensitivity in T2DM rats" is removed from our training data because its relational phrase is the preposition 'in', not a verb or a verb phrase.
- Entity 1 or entity 2 is composed by continuous words. For instance, a relation of <see in, Oral tuberculosis, secondary disease> detected in the sentence "Oral lesions of tuberculosis are seen in the secondary stages of the disease." is discarded since both of its entities are identified in discontinuous words.
- Their occurrence in the data is higher than 5.

As a result, our training data consists of more than 4 million relations with 763,065 unique relations and 7132 unique relational phrases. All entities and relational phrases were stemmed and lower-cased before training.

3.1.2. Evaluation data

To evaluate our clustering results, we created a gold standard of synonymous groups based on Nebot and Berlanga's data [12]. The data was manually crafted by selecting relational phrases that present relationship for each pair of semantic types expressed in the UMLS semantic network, e.g., protein–disease interactions, and cell–cell interactions. Synonymous phrases were then clustered into 249 groups. It should be noted that the data was created in a soft clustering fashion, i.e., one relational phrase can be assigned to more than one group. We have normalized the data by stemming every phrase and discarding duplicate terms in each group. As a result, our gold standard consists of 286 relational phrases clustered into 107 groups (including 7 singleton groups) with an average cluster size of 3.7.

3.2. Perplexity

There are several metrics that can be used for evaluating topic models [37]. In this work, we use the perplexity on the training and testing set. Formally, for a set S of M documents, perplexity is calculated as Eq. (4) [16], in which $p(w_m)$ is computed according to the value θ of the model.

$$Per(S) = \exp\left(-\frac{\sum_m \log p(w_m)}{\sum_m |w_m|}\right) \quad (4)$$

The lower the perplexity, the better the model.

3.3. Evaluation metrics

[3] SemRep [32,33], a similar system to PASMED, restricts its relations in a predefined predicate ontology based on the Semantic Network. Due to its extraction fashion, SemRep occasionally missed some types of relations while PASMED did not [20]. This is the reason why we selected PASMED over SemRep.

[4] http://semanticnetwork.nlm.nih.gov/.

[5] http://www.nlm.nih.gov/bsd/licensee/2012_stats/baseline_med_filecount.html.

Since the evaluation data was created in a soft clustering fashion, to evaluate our methods, we use two fuzzy measures, including fuzzy B-Cubed and fuzzy Normalized Mutual Information (NMI) proposed in the SemEval-2013 Task 13 [38].

Fuzzy B-Cubed is generalized from the formalization of B-Cubed, which estimates the fit between two clusterings based on item level. When we compare the two clusterings X and Y, B-Cubed precision and recall are calculated as follows [39].

$$\text{B-Cubed Precision} = \text{avg}_i[\text{avg}_{j\neq i\in\mu_y(i)}P(i,j)]$$
$$\text{B-Cubed Recall} = \text{avg}_i[\text{avg}_{j\neq i\in\mu_x(i)}R(i,j)],$$

where *avg* is a function that returns the mean value of a series, $\mu_x(i)$ is the set of clusters in clustering X of which item i is a member, $P(i,j)$ and $R(i,j)$ are item-based functions. In case of fuzzy/soft clustering, Jurgens and Klapaftis [38] redefined precision and recall as $P(i,j,X) = \frac{Min(C(i,j,X),C(i,j,Y))}{C(i,j,X)}$ and $R(i,j,X) = \frac{Min(C(i,j,X),C(i,j,Y))}{C(i,j,Y)}$. C with respect to X of the two items i, j is calculated as:

$$C(i,j,X) = \sum_{k\in l_X(i)\cup\ell_X(j)} 1 - |w_k(i) - w_k(j)|, \quad (5)$$

where $\ell_X(i)$ is the set of clusters of which i is a member, and $w_k(i)$ is the membership weight of item i in cluster k in X. In our evaluation, the membership weight is the same for every cluster.

To calculate fuzzy NMI, each cluster X_i is represented as a continuous random variable $\{w_1,\ldots,w_n\}$, a set of weight ranges that denotes the strength of membership in the fuzzy set. Jurgens and Klapaftis [38] defined the entropy and joint entropy for X_i as

$$H(X_i) = \sum_{j=1}^{n} p(w_j)\log_2 p(w_j), \quad (6)$$

where $p(w_i)$ is the probability of an instance being labeled with rating w_i, and n denotes the number of bins that X_i is discretized into. Similarly, the joint entropy of two fuzzy clusters is computed as

$$H(X_k,Y_l) = \sum_{i=1}^{n}\sum_{j=1}^{m} p(w_i,w_j)\log_2 p(w_i,w_j), \quad (7)$$

where $p(w_i,w_j)$ is the probability of an instance being labeled with rating w_i in cluster X_k and w_j in cluster Y_l. The conditional entropy between two clusters is calculated as $H(X_k|Y_l) = H(X_k,Y_l) - H(Y_l)$.

All fuzzy B-Cubed and NMI scores in our evaluation are produced by using the SemEval-2013 task 13 scorer.[6]

3.4. Other configurations

Regarding the CBOW model, we select a hierarchical softmax classification for the output layer, a context window size of 5, and in turn increase the dimension.

In case of SNE, we directly use more than 763 thousand unique relations as the input to produce clusters of synonymous strings. SNE[7] allows us to tune three parameters: the total value of $\alpha+\beta$, λ, and μ. We started with the empirical values reported in [10], which are 10, 100, and 100 respectively. We, then, tuned those values in increments of 10, 100 and 100 to find out the best performance.

[6] http://www.cs.york.ac.uk/semeval-2013/task13/index.php?id=data.
[7] http://alchemy.cs.washington.edu/papers/kok08/.

4. Results and discussion

4.1. Perplexity of the LDA-SP model

In this evaluation, we divided our training data into 10 parts; 9 parts were used for training and the other part for testing. Elements in the training and testing sets share the same indices of relational phrases.

We ran 2,000 iterations for inference with a varied number of topics and obtained the corresponding perplexity results of training and testing sets in Table 3.

This table shows that the perplexities decreased when the number of topics increased but they did not substantially change from 200 topics. Hence, we used the output of 200 topics for our clustering step.

4.2. Clustering results

After representing the relational phrases as vectors, we applied k-means clustering in Bayon[8] on top of those vectors with varying numbers of clusters (k). For each value of k, we run k-means with 10 random seeds and calculate the mean scores. We also compare our methods with Semantic Network Extractor (SNE) [10], a probabilistic model based on two MLNs. Finally, we have collected the highest performance of each method and compared them with two baselines: (1) All-One: assigning all phrases into one big cluster, and (2) One–One: assigning one phrase into one cluster.

4.2.1. Results by BOW and LDA-SP models

In terms of fuzzy B-Cubed scores, Table 4 indicates that LDA-SP performs better than BOW. More specifically, the precision scores by the two models are comparable, but the recall scores produced by the LDA-SP model are higher than those by BOW. With LDA-SP, the precision scores are decreased when the value of k is increased in all cases. Among the two models, the highest performance is an F-score of 0.103, produced by LDA-SP when k is 200.

Regarding the fuzzy NMI scores (Table 5), LDA-SP also performs better than BOW.

4.2.2. Results by SNE

Table 6 shows that SNE produced lower scores than those by LDA-SP on our data set. The best fuzzy B-Cubed F-score and fuzzy NMI are obtained in two cases: (20, 200, 200) and (30, 300, 300). It should be noted that only SNE produced singleton clusters while BOW, LDA-SP and CBOW did not. This is the primary reason why we did not mention singleton clusters in our clustering results.

4.2.3. Results by CBOW

We have conducted experiments for CBOW model with different numbers of dimensions, including 200, 600 and 1000 dimensions. However, since the three results are almost the same, we have only reported the results of 200 dimensions in this article.

The clustering results in Table 7 show that the fuzzy B-Cubed scores have the same tendency as those produced by the two LDA-SP models, i.e., the precision scores are decreased when the value of k is increased, while the recall scores are not consistent. The highest B-Cubed F-scores were obtained by 100 clusters for *Role*, and 300 clusters for both *Sentence* and *Relation*.

Among the three types of representations, the *Role* representation performs slightly better than the others despite the fact that this representation make the training data sparser. Our observation shows that this type of representation generated more correct clusters. For example, with *Sentence* and *Relation*, three strings 'in-

[8] https://code.google.com/p/bayon/.

Table 3
Perplexity of LDA-SP on the training and testing sets.

Set	Number of topics			
	50	100	200	300
Training	476.2	414.7	370.0	347.5
Testing	452.2	394.0	352.9	331.2

fect', 'be infectious for' and 'infest' were assigned to two different clusters. However, in case of *Role*, those strings were grouped into one clusters, which is more accurate according to the gold standard. Referring to their definition, it is clear that the *Sentence* and *Role* can capture the continuous context around relations while the *Relation* cannot. Therefore, these two representations yield better results than the *Relation*.

Compared with BOW, SNE, and LDA-SP, CBOW boosts the performance of clustering on both precision and recall scores. CBOW tends to produce more correct synonymous terms in clusters. For instance, it can assign eleven verbs of laboratory procedures into one group, while the other methods can partially do it, i.e., they can assign at most six terms into one group, as illustrated in Table 8. It is clear that by using word embeddings, the performance of clustering was improved significantly.

We collect the highest performance figures of each method and show them in Table 9 in comparison with the two baselines. In terms of fuzzy B-Cubed F-scores, all methods performed substantially better than the two baselines. However, regarding fuzzy NMI (see Table 5), only LDA-SP, CBOW-*Relation* and CBOW-*Role* produced better scores than the baselines.

χ^2 tests with one degree of freedom were conducted on the fuzzy B-Cubed precision and recall of three pairs of methods including SNE vs. CBOW-*Relation*, LDA-SP vs. CBOW-*Relation*, and CBOW-*Relation* vs. CBOW-*Role*. Regarding the first pair, we gained p-value < 0.05 for both precision and recall. With LDA-SP vs. CBOW-*Relation*, the p-value was less than 0.05 in case of precision, while this happened in case of recall for CBOW-*Relation* vs. CBOW-*Role*. These results can be interpreted as (1) when using the same information as SNE and LDA-SP, the CBOW model performs significantly better than the two methods; and (2) the recall is further improved by embedding the relations into sentences with keeping their roles.

4.3. Error analysis

Our error analysis has revealed that one of the main classes of errors is polysemous phrases, i.e., a phrase can belong to more than one cluster, which is identical to the definition of soft clustering. For example, the verb 'activate' is assigned to four different clusters in the evaluation data. Since our methods only focus on hard clustering, they could not solve such kind of phrases. There is about 26% of polysemous phrases, which occupy about 47% of occurrences of all phrases in the evaluation data.

Table 4
Fuzzy B-Cubed scores when BOW and LDA-SP are used to represent relational phrases.

k	BOW			LDA-SP		
	Pre.	Re.	F.	Pre.	Re.	F.
10	0.210	0.031	0.054	0.208	0.035	0.060
50	0.147	0.051	0.076	0.106	0.070	0.083
100	0.116	0.061	0.080	0.096	0.108	0.102
200	0.100	0.068	0.081	0.080	0.147	0.103
300	0.087	0.081	0.084	0.073	0.149	0.098
400	0.079	0.085	0.082	0.070	0.152	0.096
500	0.071	0.087	0.078	0.068	0.149	0.093

Considering the results reported by Jurgens and Klapaftis [38], there are systems that achieved high fuzzy B-Cubed scores in the single-sense setting (i.e., hard clustering) but low scores in multiple senses (i.e., soft clustering). More specifically, the highest B-Cubed F-score in case of hard clustering was 0.441, but the score was decreased to 0.134 in soft clustering. Therefore, we may infer that the low performance can happen even if the performance of hard clustering is reasonable.

We have manually analyzed the clustering results and found that the hard clustering is not as poor as the scores imply. The clustering results contain many clusters that are relatively correct despite the fact that they are not identical to clusters in the evaluation data. For instance, in the evaluation data, the phrases, 'interact_with', 'bind', 'activate', 'inactivate', 'cleave', 'recruit', and 'target', are in different clusters. However, CBOW-*Role* assigns all of them into one cluster, which might be correct in the sense that these phrases describe the interactions between genes, proteins and enzymes. We have reported some of such clusters generated by CBOW-*Role* with $k = 300$ in Table 10.

Another class of errors is the incorrect extracted relations. Although we tried to filter out false positive relations from the training data, we could not remove all of them. This type of relations may mislead the training process and produce noise in the clustering results. For example, a relation of <feed, pig, anaerobic bacteria> extracted from "Pigs fed the C-NP diet also showed significantly increased number of anaerobic bacteria ..." is incorrect. By accident, this incorrect relation has the same context with other relations, which may cause a wrong clustering of the verb 'feed' and other phrases.

4.4. Discussion

Unlike previous work that tried to cluster both entities and relational phrases [6,10], our work only aims at clustering the phrases. However, since we treated entities in relations as *words* in *sentences*, the trained model by CBOW can also be used to cluster entities. By calculating the cosine similarity between vector representations of entities, we can detect similar or synonymous entities. For instance, according to our model, the closet to the entity 'gastric_cancer' are 'gastric_carcinoma' and 'gastric_adeno carcinoma' with a value of 0.82, and indeed they are synonymous. This is an advantage that does not exist in BOW or LDA-SP.

Table 5
Fuzzy NMI scores produced by BOW, LDA-SP and CBOW.

k	BOW	LDA-SP	CBOW		
			Relation	*Sentence*	*Role*
10	0.0	0.002	0.009	0.008	0.005
50	0.011	0.018	0.026	0.028	0.029
100	0.037	0.060	0.073	0.052	0.061
200	0.037	0.112	0.084	0.080	0.068
300	0.048	0.118	0.101	0.084	0.101
400	0.052	0.119	0.111	0.094	0.103
500	0.056	0.125	0.120	0.112	0.121

Table 6
Clustering results of SNE with varying values of $(\alpha + \beta, \lambda, \mu)$.

Values of parameters	Fuzzy B-Cubed			Fuzzy NMI
	Pre.	Re.	F.	
(10, 100, 100)	0.120	0.052	0.072	0.060
(20, 200, 200)	0.103	0.059	0.075	0.070
(30, 300, 300)	0.102	0.060	0.075	0.070
(40, 400, 400)	0.091	0.049	0.064	0.065
(50, 500, 500)	0.099	0.050	0.067	0.060

Table 7
Fuzzy B-Cubed scores when the CBOW model is used to learn relational phrases' vectors.

k	*Relation*			*Sentence*			*Role*		
	Pre.	Re.	F.	Pre.	Re.	F.	Pre.	Re.	F.
10	0.355	0.037	0.061	0.383	0.038	0.069	0.386	0.041	0.074
50	0.205	0.169	0.125	0.270	0.087	0.132	0.256	0.084	0.126
100	0.173	0.133	0.150	0.197	0.116	0.146	0.209	0.118	0.151
200	0.141	0.146	0.143	0.174	0.156	0.164	0.179	0.144	0.159
300	0.137	0.160	0.147	0.160	0.172	0.165	0.167	0.181	0.174
400	0.128	0.168	0.146	0.150	0.177	0.162	0.147	0.185	0.164
500	0.121	0.181	0.146	0.143	0.192	0.160	0.150	0.206	0.173

Moreover, we have found that the property of algebraic operations on vector representations is maintained in this task. As stated above, we group continuous relational phrases as *words*, but not with discontinuous phrases. For example, the relational phrase between entities in the following sentence is discontinuous: "We **investigate** surviving messenger RNA MRNA expression **in** gastric_cancer." However, as expected, *vector*(investigate) + *vector*(in) is close to vectors of 'investigate_in', 'assess_in', and 'evaluate_in', which means that they are similar phrases. This property, again, confirms the robustness of the CBOW model in comparison with BOW, SNE and LDA-SP.

The highest empirical fuzzy B-Cubed F-score achieved in our experiments was 0.174. This is not an ideal level of performance but at the same time is an encouraging performance figure, considering that the clustering is done in a fully unsupervised fashion and the evaluation criteria are strict. An interesting line of future work would be to incorporate some level of supervision to further improve the clustering accuracy.

In Table 10, we show some clusters of relational phrases obtained by our model, in which most of the phrases are indeed synonymous. Referring to our motivating example in Section 1, these synonymous clusters will be useful for question–answering systems that support natural language queries such as Linked Open Data Question–Answering (LODQA).[9] Assuming that the system queries on a database of general relations output by [20]. When we input the query "What genes are essential for cell survival?", the system will return 58 unique relations in which the semantic type of the first entity is gene, the second entity is 'cell survival', and the relational phrase is 'be essential for'. However, if we use the synonymous cluster of necessity relations (the third row in Table 11), the search term can be expanded and the number of the answers would be increased to 261.

Another application-level example is applying synonymous groups to entailment detection. Rei and Briscoe [40] defined four entailment relations between two fragments A and B: $A \rightarrow B$, $B \rightarrow A$, $A = B$, and $A \neq B$. Our synonymous groups can be directly used for the third relation and for expanding results of the other relations. For instance, according to their pilot dataset,[10] there is an entailment relation as "investigates = examines", which is identical to our synonymous pair (investigate, examine). Also, an entailment relation between "stimulate → affect" can be expanded to "activate → affect" since we know that 'activate' is a synonym of 'stimulate'.

As pointed out in Section 4.3, one limitation of our work is that we did not solve the problem of polysemy. A natural approach to this issue would be using soft clustering methods. The output vector of the LDA-SP model can be interpreted as a result of soft clustering, in which LDA-SP assigns, for instance, a probability of 0.27 for topic 1, 0.15 for topic 2, 0.25 for topic 3 ..., to a phrase *p*. Let consider the topics as senses of a phrase. If we set a threshold of 0.2, the phrase *p* will belong to senses 1 and 3. However, if we set a threshold of 0.3, the phrase *p* has no sense. Ideally, for a polysemous phrase, instead of assigning a probability to each sense, the method should assign the probability of having more than two senses. This issue may be addressed by using statistical models for partial membership [41], but we leave it for future work.

Another limitation of our work is that the proposed methods cannot properly group some relational phrases that are identical in textual surface but convey different meanings, e.g., relational phrases that are involved in negation or possibility. For example, relational phrases in the following sentences are normalized to the phrase 'be_treat_with':

> patients with disease X could not be treated with drug Y because ...

Table 8
An example of clustering verbs that convey laboratory procedures by the four methods. The italic phrases are incorrect terms according to the gold standard.

Method	Clustering result
BOW	analyze, assess, examine, evaluate, estimate, test
LDA-SP	analyze, assess, examine, evaluate, investigate, test
SNE	assess, examine, evaluate, measure, *compare, confirm, detect*
CBOW	analyze, analyze, assay, assess, define, estimate, evaluate, examine, investigate, measure, test, *characterize, characterize, compare, determine, map*

Table 9
The highest performance of each method in comparison with the two baselines.

Methods	Feature	Fuzzy B-Cubed			fNMI
		Pre.	Re.	F.	
All–One	–	0.980	0.019	0.038	0.0
One–One	–	0.0	0.0	0.0	0.118
BOW	Relations	0.087	0.081	0.084	0.056
SNE	Unique relations	0.103	0.059	0.075	0.070
LDA-SP	Relations	0.080	0.147	0.103	0.125
CBOW-*Rel.*	Relations	0.173	0.133	0.150	0.120
CBOW-*Sent*	Embeded relations	0.160	0.172	0.165	0.112
CBOW-*Role*	Embeded relations with roles	0.167	0.181	0.174	0.121

Table 10
Some clusters produced by CBOW-*Role* (300 topics) are relatively correct although they are not identical to clusters in the evaluation data.

No.	Cluster
1	transfer, convert, sequester, transform, transduce
2	obtain, detect, isolate, classify, collect, cultivate, screen
3	interact_with, bind, activate, inactivate, cleave, recruit, target
4	control, drive, modulate, promote, mediate, regulate, trigger, initiate
5	affect, alter, interfere_with, modify, preserve

[9] Currently, LOQDA (http://lodqa.dbcls.jp) queries on the Online Mendelian Inheritance in Man (OMIM) database.
[10] http://www.marekrei.com/?cat=projects&page=fragmentail.

Table 11
Examples of good clusters of relational phrases. Each cluster is assigned a name that conveys its meaning.

Laboratory procedures	analyze at, analyze at, ascertain at, assess at, collect at, compare at, determine at, do at, evaluate at, examine at, exercise at, harvest at, identify at, investigate at, isolate at, measure at, monitor at, note at, obtain at, perform at, record at, remove at, sample at, screen at, study at, take at, test at
Localization relations	accumulate at, be localized in, be localized to, bud at, cluster at, colocalize with, colocalize in, colocalize with, colocalize in, co-localize in, co-localize to, colocalize with, co-localize with, colocalize within, concentrate at, concentrate in, enrich at, enrich on, localize in, localize to, localize at, localize in, localize on, localize to, localize with, localize within, localized to, locate to, recruit to, shuttle between, target to, translocate from, translocate into, translocate to
Necessity relations	be central in, be central to, be critical for, be critical in, be critical to, be crucial for, be crucial in, be crucial to, be dispensable for, be essential for, be essential in, be essential to, be fundamental to, be important for, be important in, be important to, be instrumental in, be integral to, be key to, be necessary for, be pivotal in, be sufficient for, contribute to, cooperate in, function in, involve in, participate in, require for

patients with disease X may be treated with drug Y ...

patients with disease X were treated with drug Y without success.

Consequently, these phrases are considered to be identical to 'be_treat_with' despite the fact that the corresponding sentence of each phrase presents a different possibility of the treatment of drug Y on patients with disease X. To differentiate these phrases, additional processing such as negation detection [42] and relation classification [43] is required. These tasks are important in biomedical text mining but are beyond the scope of this article.

5. Conclusions

In this paper, we have applied four unsupervised methods to cluster relational phrases. The first three methods, BOW, LDA-SP and CBOW, encode relational phrases into vector format, while SNE approaches the task by using a probabilistic model enhanced with two Markov logic networks. Our experimental results have shown that CBOW significantly outperforms BOW, LDA-SP and SNE, which demonstrates that using word embeddings is effective for detecting synonymous phrases.

Acknowledgments

We would like to thank Jin-Dong Kim for his valuable information about LODQA. We would also like to express our gratitude to the anonymous reviewers for their valuable comments and suggestions, which were helpful in improving the quality of the article.

References

[1] Y. Miyao, T. Ohta, K. Masuda, Y. Tsuruoka, K. Yoshida, T. Ninomiya, J. Tsujii, Semantic retrieval for the accurate identification of relational concepts in massive textbases, in: Proceedings of the Annual Meeting of the Association for Computational Linguistics, 2006, pp. 1017–1024.
[2] D. Rebholz-Schuhmann, H. Kirsch, M. Arregui, S. Gaudan, M. Riethoven, P. Stoehr, EBIMed – text crunching to gather facts for proteins from Medline, Bioinformatics 23 (2) (2007) 237–244.
[3] Y. Tsuruoka, M. Miwa, K. Hamamoto, J. Tsujii, S. Ananiadou, Discovering and visualizing indirect associations between biomedical concepts, Bioinformatics 27 (13) (2011) 111–119.
[4] P. Thompson, J. McNaught, S. Montemagni, N. Calzolari, R. Del Gratta, V. Lee, S. Marchi, M. Monachini, P. Pezik, V. Quochi, The BioLexicon: a large-scale terminological resource for biomedical text mining, BMC Bioinform. 12 (1) (2011) 397.
[5] L. Rimell, T. Lippincott, K. Verspoor, H.L. Johnson, A. Korhonen, Acquisition and evaluation of verb subcategorization resources for biomedicine, J. Biomed. Inform. 46 (2) (2013) 228–237.
[6] A. Yates, O. Etzioni, Unsupervised methods for determining object and relation synonyms on the web, J. Artif. Intell. Res. 34 (1) (2009) 255–296.
[7] D. Lin, P. Pantel, Discovery of inference rules for question answering, Nat. Language Eng. 7 (4) (2001) 343–360.
[8] B. Min, S. Shi, R. Grishman, C.-Y. Lin, Towards large-scale unsupervised relation extraction from the web, Int. J. Semantic Web Inform. Syst. 8 (3) (2012) 1–23.
[9] A. Moro, R. Navigli, Integrating syntactic and semantic analysis into the open information extraction paradigm, in: Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, 2013, pp. 2148–2154.
[10] S. Kok, P. Domingos, Extracting semantic networks from text via relational clustering, in: Proceedings of European Conference on Machine Learning and Knowledge Discovery in Databases, 2008, pp. 624–639.
[11] M. Richardson, P. Domingos, Markov logic networks, Machine Learn. 62 (2006) 107–136.
[12] V. Nebot, R. Berlanga, Exploiting semantic annotations for open information extraction: an experience in the biomedical domain, Knowl. Inform. Syst. 38 (2) (2014) 369–385.
[13] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, in: Proceedings of Workshop at International Conference on Learning Representations, 2013.
[14] Y. Bengio, R. Ducharme, P. Vincent, C. Janvin, A neural probabilistic language model, J. Machine Learn. Res. 3 (2003) 1137–1155.
[15] L. Sun, A. Korhonen, Improving verb clustering with automatically acquired selectional preferences, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2009, pp. 638–647.
[16] D.M. Blei, A.Y. Ng, M.I. Jordan, J. Lafferty, Latent Dirichlet allocation, J. Machine Learn. Res. 3 (2003) 993–1022.
[17] A. Ritter, Mausam, O. Etzioni, A latent Dirichlet allocation method for selectional preferences, in: Proceedings of the Annual Meeting of the Association for Computational Linguistics, 2010, pp. 424–434.
[18] E. Erosheva, S. Fienberg, J. Lafferty, Mixed-membership models of scientific publications, Proc. Natl. Acad. Sci. 101 (Suppl. 1) (2004) 5220–5227.
[19] N.T.H. Nguyen, M. Miwa, Y. Tsuruoka, S. Tojo, Open information extraction from biomedical literature using predicate-argument structure patterns, in: Proceedings of The 5th International Symposium on Languages in Biology and Medicine (LBM 2013), 2013, pp. 51–55.
[20] N.T.H. Nguyen, M. Miwa, Y. Tsuruoka, T. Chikayama, S. Tojo, Wide-coverage relation extraction from MEDLINE using deep syntax, BMC Bioinform. 16 (1) (2015) 107.
[21] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, in: Proceedings of the 27th Annual Conference on Neural Information Processing Systems, 2013, pp. 3111–3119.
[22] R. Socher, E.H. Huang, J. Pennington, A.Y. Ng, C.D. Manning, Dynamic pooling and unfolding recursive autoencoders for paraphrase detection, in: Proceedings of the 25th Annual Conference on Neural Information Processing Systems, 2011, pp. 801–809.
[23] R. Socher, J. Pennington, E.H. Huang, A.Y. Ng, C.D. Manning, Semi-supervised recursive autoencoders for predicting sentiment distributions, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2011, pp. 151–161.
[24] K. Hashimoto, M. Miwa, Y. Tsuruoka, T. Chikayama, Simple customization of recursive neural networks for semantic relation classification, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2013, pp. 1372–1376.
[25] A. Tamura, T. Watanabe, E. Sumita, Recurrent neural networks for word alignment model, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Baltimore, Maryland, 2014, pp. 1470–1480. <http://www.aclweb.org/anthology/P14-1138>.
[26] S. Pyysalo, F. Ginter, H. Moen, T. Salakoski, S. Aniadou, Distributional semantics resources for biomedical text mining, in: Proceedings of The 5th International Symposium on Languages in Biology and Medicine (LBM 2013), 2013, pp. 39–43.
[27] T. Mikolov, Q.V. Le, I. Sutskever, Exploiting Similarities among Languages for Machine Translation, CoRR abs/1309.4168, 2013.
[28] W.Y. Zou, R. Socher, D.M. Cer, C.D. Manning, Bilingual word embeddings for phrase-based machine translation, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2013, pp. 1393–1398.
[29] J. Devlin, R. Zbib, Z. Huang, T. Lamar, R. Schwartz, J. Makhoul, Fast and robust neural network joint models for statistical machine translation, in: Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Association for Computational Linguistics, Baltimore, Maryland, 2014, pp. 1370–1380. <http://www.aclweb.org/anthology/P14-1129>.
[30] T.L. Griffiths, M. Steyvers, Finding scientific topics, Proc. Natl. Acad. Sci. 101 (Suppl. 1) (2004) 5228–5235.
[31] M. Banko, M. Cafarella, S. Soderland, M. Broadhead, O. Etzioni, Open information extraction from the web, in: Proceedings of IJCAI, 2007, pp. 2670–2676.

[32] T.C. Rindflesch, M. Fiszman, The interaction of domain knowledge and linguistic structure in natural language processing: interpreting hypernymic propositions in biomedical text, J. Biomed. Inform. 36 (6) (2003) 462–477.
[33] T.C. Rindflesch, H. Kilicoglu, M. Fiszman, G. Rosemblat, D. Shin, Semantic MEDLINE: an advanced information management application for biomedicine, Inform. Services Use (31) (2011) 15–21.
[34] T. Matsuzaki, Y. Miyao, J. Tsujii, Efficient HPSG parsing with supertagging and CFG-filtering, in: Proceedings of IJCAI, 2007, pp. 1671–1676.
[35] Y. Miyao, R. Sætre, K. Sagae, T. Matsuzaki, J. Tsujii, Task-oriented evaluation of syntactic parsers and their representations, in: Proceedings of ACL, 2008, pp. 46–54.
[36] A.R. Aronson, F.-M. Lang, An overview of MetaMap: historical perspective and recent advances, JAMIA 17 (3) (2010) 229–236, http://dx.doi.org/10.1136/jamia.2009.002733.
[37] H.M. Wallach, I. Murray, R. Salakhutdinov, D. Mimno, Evaluation methods for topic models, in: Proceedings of the 26th Annual International Conference on Machine Learning, 2009, pp. 1105–1112. http://dx.doi.org/10.1145/1553374.1553515.
[38] D. Jurgens, I. Klapaftis, SemEval-2013 task 13: word sense induction for graded and non-graded senses, in: Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013), Association for Computational Linguistics, Atlanta, Georgia, USA, 2013, pp. 290–299.
[39] E. Amigó, J. Gonzalo, J. Artiles, F. Verdejo, A comparison of extrinsic clustering evaluation metrics based on formal constraints, Inform. Retr. 12 (5) (2009) 613.
[40] M. Rei, T. Briscoe, Unsupervised entailment detection between dependency graph fragments, in: Proceedings of BioNLP 2011 Workshop, 2011, pp. 10–18.
[41] K.A. Heller, S. Williamson, Z. Ghahramani, Statistical models for partial membership, in: Proceedings of the 25th International Conference on Machine Learning, 2008, pp. 392–399. <http://dx.doi.org/10.1145/1390156.1390206>.
[42] J.-D. Kim, T. Ohta, S. Pyysalo, Y. Kano, J. Tsujii, Overview of bionlp'09 shared task on event extraction, in: Proceedings of the Workshop on Current Trends in Biomedical Natural Language Processing: Shared Task, BioNLP '09, Association for Computational Linguistics, Stroudsburg, PA, USA, 2009, pp. 1–9. <http://dl.acm.org/citation.cfm?id=1572340.1572342>.
[43] O. Uzuner, B.R. South, S. Shen, S.L. DuVall, 2010 i2b2/VA challenge on concepts, assertions, and relations in clinical text, JAMIA 18 (5) (2011) 552–556.