View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by DigitalCommons@Kennesaw State University

Kennesaw State University
DigitalCommons@Kennesaw State University

Faculty Publications

2010

Biomedical Relationship Extraction from Literature Based on Bio-Semantic Token Subsequences

Ying Xie
Kennesaw State University, yxie2@kennesaw.edu

Jayasimha R. Katukuri
EBay Inc.

Vijay V. Raghavan
University of Louisiana at Lafayette

Follow this and additional works at: http://digitalcommons.kennesaw.edu/facpubs

 Part of the Computer Sciences Commons

Recommended Citation
Xie, Ying; Katukuri, Jayasimha R.; and Raghavan, Vijay V., "Biomedical Relationship Extraction from Literature Based on Bio-Semantic Token Subsequences" (2010). *Faculty Publications*. 1600.
http://digitalcommons.kennesaw.edu/facpubs/1600

This Article is brought to you for free and open access by DigitalCommons@Kennesaw State University. It has been accepted for inclusion in Faculty Publications by an authorized administrator of DigitalCommons@Kennesaw State University. For more information, please contact digitalcommons@kennesaw.edu.

Biomedical Relationship Extraction from Literature Based on Bio-Semantic Token Subsequences

Jayasimha R. Katukuri
University of Louisiana, Lafayette
jay@louisiana.edu

Ying Xie
Kennesaw State University
yxie2@kennesaw.edu

Vijay V. Raghavan
University of Louisiana, Lafayette
vajay@cacs.louisiana.edu

Abstract

Relationship Extraction (RE) from biomedical literature is an important and challenging problem in both text mining and bioinformatics. Although various approaches have been proposed to extract protein-protein interaction types, their accuracy rates leave a large room for further exploration of more effective methods. In this paper, two supervised learning algorithms based on newly-defined "bio-semantic token subsequence" are proposed for multi-class biomedical relationship extraction. The first approach calculates a "bio-semantic token subsequence kernel", while the second one explicitly extracts weighted features from bio-semantic token subsequences. The proposed structure called "bio-semantic token subsequence" is able to capture semantic features from natural language sentences for biomedical RE. Two supervised learning algorithms based on the proposed structure outperform the state-of-the-art biomedical RE methods on multi-class protein-protein interaction extraction.

1 Introduction

Relationship Extraction (RE), which aims at extracting relationship(s) between given entities from unstructured data, has attracted intensive research efforts in the last few years especially in the bioinformatics area. Various computational approaches were reported to extract protein-protein interactions from biomedical literature. However, most of those approaches are limited to binary relationship extraction that determines whether two proteins interact. Since two proteins may interact with each other in multiple ways, it is far more useful to extract the exact type of interaction between them. More specifically, given a sentence containing two target biomedical entities, we would like to have a machine learning algorithm that is able to automatically identify the type of relationship expressed by the sentence between these two entities. This problem is referred to as multi-class relationship extraction.

Our work in this paper focuses on using supervised learning methods to solve the multi-class relationship extraction problem. Each sentence in the training set contains the two target entities and is assigned a relation type between these two entities. The challenge of this type of supervised learning methods for RE lies on explicit or implicit extraction of relationship-related features from natural language sentences. The work of Blaschke and Valencia [2] used manually generated rules composed of sequences of words, part of speech (POS) tags, and categories with positional information to capture the features for RE. Although the generated rules might be expressive, this approach itself is not scalable. The work on text categorization using the string kernels [8] motivated the design of kernel methods for RE [12]. The string kernel based approach uses character sequences to capture features for RE.

In order to address the challenge of automatic feature extraction from natural language sentences for biomedical RE, we first introduce a structure called *bio-semantic token subsequence*. A bio-semantic token subsequence is composed of both biomedical entities and their semantic types, as well as stemmed non-biomedical words that are automatically extracted from a given sentence. This proposed structure is anticipated to capture semantic features from natural language sentences for biomedical RE. Based upon the extracted bio-semantic token subsequences, two learning methods are proposed to conduct relationship classification. The first approach calculates a "*bio-semantic token subsequence kernel*" that implicitly utilizes features captured by the bio-semantic token subsequences. The second learning approach is called "*discriminative bio-semantic token subsequence classifier"*, which explicitly generates a discriminative subset of bio-semantic token subsequences from a training set to form the feature vectors for further induction.

2 Related Work

We divide the related work on relationship extraction into three broad categories: rule-based, graphical models and discriminative models.

Rule-based systems: The paper by Blaschke and Valencia [2] uses manually built language constructs (patterns) to extract protein-protein interactions. This is one of the early RE works in biomedical domain and showed its potential. But the authors concluded that a system in future should be more flexible and easy to build without the need to construct rules manually. Other rule/template-based relationship extraction systems include [11]. The paper [7] uses an Inductive Logic Programming (ILP) method on protein-location relationship extraction. A major advantage of ILP is that it provides a straight-forward way to incorporate domain knowledge and produces logical

978-0-7695-3885-3/09 $26.00 © 2009 IEEE
DOI 10.1109/BIBM.2009.74

IEEE computer society

clauses suitable for analysis and revision by humans to improve performance. All of these works use predefined templates or semantic grammars which are not portable from one domain to another. Furthermore, none of these approaches studied multi-class relationship extraction.

Graphical Models: The paper by Ray and Craven [9] used Hidden Markov Models (HMM) to extract relationships among objects from biomedical texts. Their approach incorporates grammatical structure of sentences into HMM architecture to extract subcellular-localization relations. This work showed that using the grammatical structure of the sentences improves the precision and recall performance compared to using only words of sentences. However, one of the difficult aspects of using HMM is in designing the architecture, which also requires domain knowledge. Furthermore, HMM does not allow modeling of the long-range dependencies of the observations that are often found in biomedical RE. Conditional Random Fields (CRF) provide a solution to this problem.

Discriminative Models: Chang and Altman [5] developed models based on maximum entropy to extract pharmaco-genetic relationships between genes and drugs from the biomedical literature. String kernel-based methods for relationship extraction were employed in [12] and [3]. However, both of these methods conducted binary relationship extraction; that is, to determine if a relationship exists or not. The work of Erkan et al. [6] used the shortest path between two genes evaluated by edit distance in a dependency tree to define a kernel function for extracting gene interactions. Although, the dependency tree, which is obtained by shallow parsing of the sentences, is able to capture the relationships between non-contiguous words, the edit distance itself doesn't model the local matches well. The work of Airola et al. [1] presented a kernel method called "all-paths graph kernel" for protein-protein interaction extraction. Their method uses the dependency graph in defining a graph kernel. The work only studied binary relationship extraction. The two methods proposed in this paper also belong to the category of discriminative models. However, in contrast to [5], our methods use semantically enriched subsequences to implicitly and explicitly form features for classification, instead of just using bag of words.

3 Bio-semantic Token Subsequences

Given a sentence "This report describes a patient with generalized argyria caused by ingestion of homemade colloidal silver solution", the relationship between entities *argyria* and *colloidal silver solution* can be described by a non-contiguous subsequence, "argyria caused by colloidal silver solution". Meaningful subsequences like the previous example are good features for relationship extraction. According to the definition in [8], a sparse subsequence is a subsequence that may not be contiguous in the original sequence. Cancedda et al. [4] directly utilized sparse subsequences of words shared by two sentences to form a kernel that is an extension of string kernel [8] for text categorization. We propose a structure called *bio-semantic token subsequences* as the basis for feature extraction from biomedical sentences. A bio-semantic token subsequence is a semantically enriched sparse subsequence. In order to illustrate the concept of bio-semantic token subsequences, we take the sentence S as an example to go through the following steps that transform S to another sequence S^*. Step 1, remove stop words from S; Step 2, identify all biomedical entities, such as the ones underlined in sentence S; Step 3, each biomedical entity (BME) in S is tagged with its semantic type (ST), such that each BME becomes a ***(ST-BME)*** pair. The semantic types are a set of broad subject categories provided by the Unified Medical Language System (UMLS); Step 4, identify the verbs. We distinguish words which are verbs from other words because relationship keywords are more likely to be verbs. We denote this new sequence obtained as the result of applying the above three steps as S^*.

Sentence S: "*Additional treatment with losartan potentiated the stimulatory effects of a low-salt diet, of furosemide and of isoproterenol infusion on renin gene expression.*"

Sequence S^*: "*Additional treatment* ***(DRUG-losartan)*** *potentiated stimulatory effects* ***(TREATMENT-low-salt diet) (DRUG-furosemide) (DRUG-isoproterenol)*** *infusion* ***(PROTEIN-rennin) (GENE FUNCTION-gene expression)***"

We refer to S^* as *bio-semantic token sequence*. The bio-semantic token subsequences shared by two or more bio-semantic token sequences form bio-semantic token subsequence patterns.

4 Bio-semantic Kernel

In this section, we present the bio-semantic kernel that utilizes the common bio-semantic token subsequences shared by two sentences to evaluate the similarity between the two sentences. The bio-semantic kernel can be viewed as an implicit way to perform feature extraction based on the concept of bio-semantic token subsequence.

Given two sentences S and T, we first convert them to the corresponding bio-semantic token sequences S^* and T^*. The proposed kernel is similar to that of word sequence kernel (WSK) proposed in [4], with the added property that each token can carry additional features such as entity types, part-of-speech tag information. The common subsequences (bio-semantic token subsequences) are made-up of tokens of different types: 'biomedical entities", 'semantic types", "verb words" and "non-verb words".
For example, consider the following sentences:

S = "*Results show that meropenem interacts synergistically in combination with aminoglycoside*"
T = "*Results show that meropenum acts more synergistically with zidovudine than with aminoglycoside"*

The corresponding bio-semantic sequences are:

S^* = "*Results (DRUG-meropenem) interacts synergistically combination (DRUG- aminoglycoside)*"

T^* = "*Results (DRUG-meropenem) acts more synergistically (DRUG-zidovudine) (DRUG-aminoglycoside)*"

Some of the common bio-semantic token subsequences that will be generated are: "Results synergistically", "meropenum synergistically aminoglycoside", "DRUG synergistically DRG".

In string kernels [8], two common subsequences of same length and same number of characters in the gap contribute exactly the same value to the similarity between two sequences. In WSK, authors proposed the idea of using symbol-dependent decay factors for words in the common subsequences (matches) and for words in the gap. We further build on this idea by explaining two key observations in the context of relationship extraction.

First, two common bio-semantic token subsequences with the same length and the same number of tokens in gap but composed of different types of tokens may contribute differently to the degree of similarity between the corresponding pair of sequences.
The following two common bio-semantic token subsequences shared by S^* and T^* are examples for token subsequences that are the same in length but differ in the type of tokens that they contain: "*DRUG synergistically DRUG*", "*meropenem synergistically aminoglycoside*". The subsequence "meropenem synergistically aminoglycoside" contributes more to the degree of similarity between S^* and T^* than that of "DRUG synergistically DRUG", given that a biomedical entity itself provides more details than its semantic type.

Secondly, two common bio-semantic token subsequences with the same length and composed of the same types of tokens, but having gaps that are composed of different types of tokens may contribute differently to the degree of similarity between the corresponding pair of sequences.

In order to model the complexity brought by different types of tokens in a bio-semantic token sequence, we use two different sets of decay factors. One is $\lambda_{m,x}$ to reflect the effect that different types of tokens in a common bio-semantic token subsequence have on the evaluation of similarity between two corresponding bio-semantic token sequences; the other is $\lambda_{g,x}$ to reflect the effect brought by different types of tokens in a gap within a common subsequence on similarity evaluation. Each element in the following set of $\lambda_{m,x}$ represents the matching decay factors for tokens of type semantic type, biomedical entity, verb, and non-verb word respectively: $\{\lambda_{m,st}, \lambda_{m,bme}, \lambda_{m,v}, \lambda_{m,nv}\}$. While the following set of $\lambda_{g,x}$ are the gap decay factors for the following tokens (***ST-BME***), verb, and non-verb when they occur in a gap: $\{\lambda_{g,bme}, \lambda_{g,v}, \lambda_{g,nv}\}$ (Please note that when a (***ST-BME***) occurs in a gap, we only consider the effect of the biomedical entity to avoid dual counting). The type-dependent decay factors were also used in the word sequence kernels [4]. In word sequence kernels, however, the decay factors of words are just based on their part-of-speech tags. The bio-semantic kernel is the combination of the basic construct of string kernel and the complex setting of $\lambda_{m,x}$, $\lambda_{g,x}$. The bio-semantic kernel function is presented as formula 4.1:

$$\bar{K}_n(S^*, T^*) = \sum_{l=1}^{n} l * K_l(S^*, T^*) \quad \ldots\ldots (4.1)$$

In the bio-semantic kernel, kernel value between two sequences is calculated by computing the sub-kernel values for subsequences of length from 1 to n separately and giving the higher weight to the kernel values corresponding to longer common subsequences. As shown in the equation 4.1, $K_l(S^*, T^*)$ is the sub-kernel value computed using the common bio-semantic subsequences of length l; and the weighted sum of all sub-kernels with different lengths forms the final kernel value $\bar{K}_n(S^*, T^*)$. The sub-kernel $K_l(S^*, T^*)$ between two sequences S^* and T^* is defined as:

$$K_l(S^*, T^*) = \sum_{u \in CU} \phi_u(S^*)\phi_u(T^*) \quad \ldots\ldots\ldots (4.2)$$

In equation 4.2, CU is the set of all common bio-semantic subsequences of length l and $\phi_u(S^*)$, $\phi_u(T^*)$ are the weights of a common subsequence $u \in CU$ in sequences S^*, T^* respectively. The weights are computed using the below equation:

$$\phi_u(S^*) = \sum_{\mathbf{k}: s[\mathbf{k}]=u} \prod_{1 \le j \le l} \lambda_{m,u_j} \prod_{k_1 < i \le k_{|u|}, i \notin \mathbf{k}} \lambda_{g,s_i} \quad \ldots\ldots (4.3)$$

In the above equation, λ_{m,u_j} and λ_{g,s_i} are the token dependent match and gap decay factors that we have discussed earlier in this section. Please refer to the section 4.1.2 of the paper by Cancedda et al. [4] for further details about the weight equation and kernel formulation. The weight $\phi_u(T^*)$ can be computed using the same equation. The direct computation of the kernel defined in equation 4.2 is very costly. Hence, in the implementation of bio-semantic kernel we use the recursive dynamic programming formulation proposed in [4].

Although this proposed bio-semantic kernel is able to take advantage of semantic information provided by the bio-semantic token subsequences, a potential issue of this kernel is that all common token subsequences are utilized in computing the kernel value for a pair of sequences, which may lead to overfitting. In the next section, therefore, we propose a new method for biomedical RE that is not only able to utilize all the information that the bio-semantic kernel based

method does, but also takes into account the discriminative measure of the common token subsequences.

5 Discriminative Bio-semantic Token Subsequence Classifier (DTS)

The proposed DTS classifier utilizes only the top ranking discriminative bio-semantic token subsequence patterns as the feature set for classification. This method uses feature vectors formed by explicit feature extraction (The kernel method proposed in section 4 does not enumerate the common subsequences and hence is referred as implicit feature extraction method). In subsection 5.1, we first discuss how to explicitly extract bio-semantic token subsequence patterns and calculate their weights. Then in subsection 5.2, we present the strategy for selecting a given number of top ranking discriminative patterns to form the feature vectors for further classification.

5.1 Pattern-Sentence Matrix Computation

We first convert all the sentences in the training set to bio-semantic token sequences. Then, for each sequence in a relationship class, we compute the common bio-semantic token subsequences with other sequences from the same class. In order to avoid generating too many bio-semantic token subsequences, we require that each subsequence must contain at least one of the two target entities.

The common bio-semantic token subsequences generated are stored in a matrix called the Pattern-Sentence matrix. In this matrix, rows correspond to the computed common bio-semantic token subsequences, also referred to as bio-semantic token subsequence patterns or just patterns for simplicity, columns correspond to sentences. The cell values $w(i, j)$ represent the weight of a pattern in a sentence. The weight of a pattern within a sentence is computed by considering the following factors: 1) different types of tokens included in the pattern, 2) gaps within the pattern in the corresponding bio-semantic token sequence; 3) the number of tokens in the pattern; and 4) the frequency of a pattern within the sequence.

$$\phi_u(S^*) = \sum_{\mathbf{k}: s[\mathbf{k}]=u} l \prod_{1 \le j \le l} \lambda_{m,u_j}^{1/\sqrt{l}} \prod_{k_1 < i \le k_{|u|}, i \notin \mathbf{k}} \lambda_{g,s_i}^{1/\sqrt{l}} \quad \text{.....(5.1)}$$

5.2 Discriminative Patterns Computation

The weighting formula discussed in the previous subsection only takes into consideration statistics within each sequence. We also need to factor the statistics within each relationship class and across classes in order to achieve better performance. In this subsection, we use Chi-Square test to evaluate different degrees of discriminative power of different patterns. Chi-Square scores are computed for all the patterns in the Pattern-Sentence matrix. A high value of Chi-Square for a pattern u and class c means the pattern u is a discriminatory pattern that can help in distinguishing a sentence belonging to class c. All the patterns are ordered for each class according to their Chi-Square values. Feature selection is done by choosing only the top "k" patterns from the ordered list of patterns for every class.

The feature vector for each sentence is finally formed using the set of discriminative patterns selected by the Chi-Square measure. Recall that the weight of each pattern selected as one of the features has already been calculated by using formula 5.1 in the pattern-sentence matrix computation process. In our experiments, we use the Support Vector Machine (SVM) as the learning algorithm.

6 Experimental Results

In this section, we present the experimental results of the two proposed relationship extraction methods on a protein-protein interaction data. We used the protein-protein interaction data from the BioText group at the University of California, Berkeley [10]. This dataset is created using the *HIV-1 human protein interactions database* that contains the following information: 1) A pair of proteins (PP); 2) The interaction types between them; and 3) Pubmed identification numbers (PMID) of the journal articles describing the interactions. The combination of a pair of protein and a related PMID is referred to as a triple. A triple is assigned an interaction type in this database. All the sentences of a triple are assigned the same interaction type of the triple as their class label. We randomly selected 75% of the data (triples) as training set and the remaining is held out as test set (The same percentages are used in [16] for dividing training and testing data).

6.1 Learning Parameter Values on Training Set

An important step for learning both the bio-semantic kernel based classifier and the DTS classifier is to find out an optimal set of values for the following parameters: match decay factors ($\lambda_{m,x}$), gap decay factors ($\lambda_{g,x}$), and the maximum pattern length (L). We divide the parameter learning for the DTS classifier into two stages: The first stage of learning finds out a good set of values for decay factors; then the second stage of learning finds out a good value for L. On both stages of parameter learning, 10-fold cross-validation is used on the training set to evaluate the performance of each set of parameter values under examination. The same set of decay factors and L value learned for the DTS classifier will also be used to build the bio-semantic kernel based classifier on the training set. Further details on the parameter learning are omitted due to page limit.

6.2 Overall Performance of the Proposed Methods on the Test Set

Finally, we compare the results of our proposed methods with the following three methods *Neural Networks, Dynamic Model,* and *Naïve Bayes* that were

reported in [10]. We use the optimal parameter settings found in subsection 6.1 to test both the methods on the held-out test data. In [10], two settings are used to test the proposed methods. The first setting retains all target proteins in the sentences, whereas the second setting replaces the target proteins with "PROT1" and "PROT2". The second setting is "fairer" in the sense that it tries to avoid the same pair of target proteins to appear in both training data and test data, which may cause a learning algorithm to overfit on the target protein names. The results in Table-1 show that our discriminative method outperforms all three existing approaches reported in [10] in both settings. Our bio-semantic kernel approach outperforms the three existing learning methods in second setting, and outperforms two of the three methods in the first setting.

	Classification Accuracy	
	Setting- 1	Setting - 2
Dynamic Model	60.5%	60.5%
Naïve Bayes	58.1%	59.7%
Neural Networks	63.7%	51.6%
Discriminative Method	**66.7%**	**65.6**%
Bio-semantic Kernel	**60.9%**	**60.6**%

Table-1: Comparison of the two methods with other methods

7 Conclusions

In this paper, we proposed a structure called *bio-semantic token subsequence* to capture semantic features from natural language sentences for biomedical RE. Two supervised learning algorithms based on the proposed structure are designed. The first approach "*bio-semantic token subsequence kernel*" implicitly utilizes features captured by the bio-semantic token subsequences. The second learning approach is called "*discriminative bio-semantic token subsequence classifier"*, which explicitly generates a discriminative subset of bio-semantic token subsequences. Compared with the proposed kernel-based approach, the proposed discriminative bio-semantic token subsequence classifier further takes into consideration different discriminative degrees of different features, so as to select the most discriminative features to build a classification model. Both of these two proposed methods outperform the state-of-the-art methods reported in the literature. As expected, the performance of discriminative bio-semantic token subsequence classifier is the best.

References

[1] Airola, A., Pyysalo, S., Bjorne, J., Pahikkala, T., Ginter, F. and Salakoski, T. (2008) All-paths graph kernel for protein-protein interaction extraction with evaluation of cross-corpus learning. *BMC Bioinformatics, Vol. 9, Suppl 11.*

[2] Blaschke, C. and Valencia, A. (2002) The frame-based module of the Suiseki information extraction system, *IEEE Intelligent Systems*, **17**, 14-20.

[3] Bunescu, R.C. and Mooney, R.J. (2005) Subsequence Kernels for Relation Extraction. *Proceedings of the 19th Conference on Neural Information Processing Systems (NIPS)*. Vancouver, BC, 171-178.

[4] Cancedda, N., Gaussier, E., Goutte, C. and Renders, J. (2003) Word-Sequence Kernels, *Journal of Machine Learning Research*, **3**, 1059-1082.

[5] Chang, J.T. and Altman, R.B. (2004) Extracting and Characterizing Gene-drug Rela-tionships from the Literature, *Pharmacogentics*, **14**, 577-586.

[6] Erkan,G., Ozgur, A., and Radev, D. (2007). Semi-supervised classification for extracting protein interaction sentences using dependency parsing. *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, Prague, Czech Republic, pp. 228–237.

[7] Goadrich, M., Oliphant, L. and Shavlik, J. (2004) Learning Ensembles of First-Order Clauses for Recall-Precision Curves: A Case Study in Biomedical Information Extraction. *Proceedings of the 14th International Conference on Inductive Logic Programming (ILP)*. Porto, Portugal, 98-115.

[8] Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N. and Watkins, C. (2002) Text Classification using String Kernels, *Journal of Machine Learning Research*, **2**, 419-444.

[9] Ray, S. and Craven, M. (2001) Representing Sentence Structure in Hidden Markov Models for Information Extraction. *Proceedings of the 17th International Joint Conference on Artificial Intelligence*. Morgan Kaufmann, Seattle, WA, 1273-1279.

[10] Rosario, B. and Hearst, M. (2005) Multi-way Relation Classification: Application to Protein-Protein Interaction. *Proceedings of the HLT/EMNLP'05*. Association for Computational Linguistics, Vancouver, 732-739.

[11] Thomas, J., Milward, D., Ouzounis, C., Pulman, S. and Carroll, M. (2000) Automatic Extraction of Protein Interactions from Scientific Abstracts., *Pacific Symposium on Biocomputing*. World Scientific Press, Honolulu, Hawaii, 541-552.

[12] Zelenko, D., Aone, C. and Richardella, A. (2003) Kernel Methods for Relation Extraction, *Journal of Machine Learning Research*, **3**, 1083-1106.