Recognition of Multi-sentence *n*-ary Subcellular Localization Mentions in Biomedical Abstracts

Gabor Melli[1], Martin Ester [1], Anoop Sarkar[1]

[1]School of Computing Science, Simon Fraser University, Burnaby, British Columbia, CANADA

Email addresses:

GM: melli@sfu.ca

ME: ester@cs.sfu.ca

AS: anoop@cs.sfu.ca

Abstract

Background

Research into semantic relation recognition from text has focused on the identification of binary relations that are contained within one sentence. In the domain of biomedical documents however relations of interest can have more than two arguments and can also have their entity mentions located on different sentences. An example of this scenario is the ternary relation of “subcellular localization” which relates whether an organism’s (*O)* protein (*P)* has subcellular location (*L)* as one of its target destinations. Empirical evidence suggests that approximately one half of the mentions for this ternary relation reside on multi-sentence passages.

Results

We introduce a relation recognition algorithm that can detect *n*-ary relations across multiple sentences in a document, and use the subcellular localization relation as a motivating example. The approach uses a text-graph representation of the entire document that is based on intrasentential edges derived from each sentence’s predicted syntactic parse trees, and on intersentential edges based on either the linking of adjacent sentences or the linking of coreferents, if reliable coreference predictions are available. From the text graph state-of-the-art features such as named-entity features and syntactic features are produced for each argument pairing. We test the approach on the task of recognizing , in PubMed abstracts, experimentally validated subcellular localization relations that have been curated by biomedical researchers. When tested against several baseline algorithms, our approach is shown to attain the highest F-measure.

Conclusions

We present a method that naturally supports the recognition of semantic relations with more than two arguments and whose mentions can reside across multiple sentences. The algorithm accelerated the extraction of experimentally validated subcellular localizations. Given that the corpus is based on abstracts, not copyrighted papers, the data is publicly available from koch.pathogenomics.ca/pplre/. Significant work remains to approximate human expert levels of performance. We hypothesize that additional features are required that provide contextual information from elsewhere in the document about whether the relation refers to an experimentally validated finding.

Background

Much of the world's biomedical knowledge is contained in the natural language text within research papers that are increasingly becoming available online. Applications that have begun to tap this knowledge include information extraction and question answering algorithms, but these algorithms require effective approaches to recognize semantic relations between entity mentions. As has occurred in other natural language processing tasks, such as named entity recognition, approaches to relation recognition have evolved over time from knowledge engineering heuristic-based ones [2] to those that apply supervised machine learning algorithms to the task [1,5,8,10,11,13,14]. Early supervised learning approaches used bag-of-word representations of the document [14], then quickly proceeded to analyze shallow sequence representations [1], and most recently the emphasis has been on full syntactic parsing of each sentence [8,10,13].

While the performance of supervised relation detection has improved significantly since initial proposals [11], many more advances in the field are required before human levels of competency are attained. State-of-the-art performance on the protein/gene interaction task is currently 75% F-measure [5], but this performance was attained on binary relations and the evaluation does not include the missed relations where entity mentions reside on separate sentences. Research in NLP that has looked at information in multiple sentences has focused on the topics of co-reference, and more specifically, in entity detection and tracking across sentences. However, such research has not yet been used in combination with state-of-the-art approaches to relation detection, especially those that use state-of-the-art features. As a motivating example, consider the following passage composed of three sentences: "*The pilus*$_{\text{LOCATION}}$ *of V. cholerae*$_{\text{ORGANISM}}$ *is essential for intestinal colonization. The pilus*$_{\text{LOCATION}}$ *biogenesis apparatus is composed of at least nine proteins. TcpC*$_{\text{PROTEIN}}$ *is an outer membrane*$_{\text{LOCATION}}$ *lipoprotein required for pilus*$_{\text{LOCATION}}$ *biogenesis.*" To our knowledge no supervised relation recognition algorithm can currently identify the ternary relation between the organism in the first sentence, and the subcellular location and protein in the third sentence. This relation would go undetected by current information extraction algorithms.

Our work aims to address this scenario in order to improve the Recall and F-measure of relation recognition methods. We propose a framework that subsumes the representation used by state-of-the-art approaches when applied to the detection of binary relations within a single sentence. The framework is centered on a text-graph representation that includes intersentential edges. We illustrate

that the generation of relation cases when dealing with multi-sentence passages can significantly increase the proportion of false relation cases from which to construct a classification model, but that our approach copes with this increase in negative cases.

The remainder of the paper is structured as follows: The next section defines the more general task of semantic relation detection considered in this paper and summarizes the current challenges that motivate further research into the topic. The text-graph framework and relation case generation are then described in detail along with the feature space that generalizes existing methods are introduced. Finally, we present the empirical results of experiments performed on the task of recognizing subcellular localizations within PubMed abstracts.

Implementation

As suggested, the implemented system involves the representation of a document as a text graph and the conversion of the graph into a vector-based feature space. We name the system described below TeGRR, for: Text Graph-based Relation Recognizer. To ground the implementation however, we first present a detailed task definition.

Task Definition

We present a definition of the task for relation recognition that generalizes the standard definition for the recognition of binary relation mentions within a single sentence in order to also encompass the recognition *n*-ary relations that can span across multiple sentences. Assume that we are given a natural language text *document* (*D*) composed of a sequence of one or more sentences, where each sentence is composed of a sequence of one or more tokens. A *token* can be either a *word*, *punctuation*, or an *entity mention*. Assume that all of the entity mentions in document *D* have been labeled by an entity mention recognition algorithm, as $E_{i,j}$, where *j* refers to the j^{th} entity mention of an entity of type E_i. Examples of entity mentions include the proper name "*TcpC*", the nominal "*the protein*", and the pronoun "*it*". We are also given a typed semantic relation $R(A_1,\ldots,A_n)$, where argument A_i accepts only entity mentions of type E_i. Examples of typed semantic relations are *ProteinInteraction*(*Protein*, *Protein*) and *Subcellular-Localization*(*Organism*, *Protein*, *Location*). A *binary relation* is a relation with two arguments; while one with more than two arguments is an *n*-ary *relation*. A *relation case* C_i for document *D* and for semantic relation *R* is a permutation of the entity mentions in *D* that satisfy the argument entity type requirements of relation *R*: $C_i=(D, R(E_{1,j'}, \ldots, E_{n,j''}))$. A *relation case* can be *labeled* as *true* or *false*

depending on whether the *n* entity mentions of the case indeed stand in relation *R* or do not. A document can contain zero or more relation cases. From the sample passage presented in the introduction, with one organism mention, one protein mention and four location mentions, three false relation cases and one true relation case can be generated. A *training relation case* is a relation case where this label is known, whereas a *test relation case* is one where the label is unknown (or is hidden). The set of all training relation cases forms the *training set*, and the set of test relation cases is the *test set*. Given a training set, the task of supervised relation detection is to learn a classification model that can accurately predict the (unknown) label of a test relation case based on its observed features.

Text-Graph Representation

The core of the proposed framework is a graph-based representation of each document. The text graph representation is composed of the following types of nodes and edges: 1) Intrasentential nodes and edges; 2) Sentence to Sentence edges; and 3) Coreference nodes and edges. A sample text graph which makes use of the three edge types is presented in Figure 1.

Intrasentential Nodes and Edges

Intrasentential nodes and edges are intended to represent the information contained within a single sentence. Many candidates for these edges exist in the literature. They include: word-to-word edges [4], shallow parsing edges [15], dependency parse tree edges [9], and phrase-structure parse tree edges [17]. We propose to use the phrase-structure parse tree as the source of intrasentential edges for two reasons. First the recent analysis by [8] suggests that the phrase-structure parse tree is the single best source of information for relation detection. Secondly, all other proposed intrasentential edges can be derived from phrase-structure parse trees by means of simple transformations. Two types of nodes are associated to a phrase-structure parse tree: leaf nodes and internal nodes. Leaf nodes contain 1) a word, punctuation mark or entity mention, 2) the part of speech tag, and 3) a named entity tag if one exists. Internal nodes contain the syntactic phrase-structure label.

Sentence-to-Sentence Edges

The first type of intersentential edges considered is the “sentence-to-sentence” edge. This edge type simply joins an end-of-sentence punctuation node with the first word of the subsequent sentence. The intuition for this edge is that an entity that is mentioned in one sentence can be in a semantic relation with an entity in the adjacent sentence and that the likelihood of such a relation diminishes with increasing number of sentences that exists between the two entity mentions. The text graph in Figure 1

contains two sentence-to-sentence edges: one between the period punctuation token in the first sentence and the first word ("The") in the second sentence; the other between the period punctuation token in the second sentence and the first word ("TcpC") in the third sentence.

Coreference Nodes and Edges

Another case of intersentential edges that will be considered is that of coreference edges. These edges assume that in-document coreference resolution has been accurately performed. The intuition for this edge is that because the entities refer to the same thing, anything that is said in one sentence can apply to the entity in the next mentioning of the entity. We create a node for each entity and associate an edge between the node and each entity mention. The text graph in Figure 1 contains three coreference edges. The edges all relate to the same entity "pilus" which we assume to be detected by a named-entity recognition system as be referring to the same concept.[1]

Relation Case Generation

This section describes a procedure to generate the relation cases that will be used to train and test a classification model. The standard approach to case selection used by single-sentence relation detection algorithms is to generate all possible permutations within the sentence. To handle multiple sentences we simply extend the discovery of permutations from within the entire document. Figure 3 presents a sample relation case drawn from one of the permutations for the text graph in Figure 1.

Input: 1) A text-graph, G; 2) A semantic relation with n arguments, $R(A_1,\ldots,A_n)$

Output: A set of unlabeled relation cases, C.

Algorithm:

- Select all entity mentions $E_{i,j}$ in G where E_i is of the same type as a_i.
- Create every permutation among these pairings.
- Associate the nodes along the path between each pair of arguments.

Addition of Syntactic Nodes within the Shortest Path-enclosed Tree

The relation case as described so far is simply composed of the nodes within the shortest path between each pair of entity mentions. We now attach other nearby nodes that are known to be relevant to predicting semantic relations. Specifically we add the syntactic nodes and edges contained within the smallest common subtree in a sentence. These nodes have been shown to be an important source for predictive features for relation detection by [17] and [8]. Figure 2 illustrates the shortest path-enclosed

[1] The word "pilus" is associated with the Gene Ontology's GO0009289 entry.

tree expansion. In the example one additional node, the one containing the preposition “of”, is appended to the path. Two extensions to the definition of path-enclosed tree are required as a result of the support of intersentential edges. As shown in the Figure 2, the subtree can now be bounded not just by entity mentions, such as the *Vibrio Cholerae* node in the example, but can also be bounded at the other end by a node attached to an intrasentential edge, such as the *pilus* node in the example, or by a node with a sentence-to-sentence edge. A second possible scenario is the one where a sentence is traversed but does not contain an entity mentions whatsoever. In this case the nodes between the two nodes in the traversal path are not included in the definition of path enclosed. The intuition here is that it is the words in the sentence containing an entity that are predictive of its semantic relation.

Clearly more nodes and edges from the text graph could be included into the relation case definition to provide additional information about the relation case. The proposal above however covers the portion of the graph suggested in state-of-the-art systems.

Feature Space Definition

Given the above definition of a relation case graph, we are now ready to describe the feature space that each relation case will be mapped to for the classification task. As a design principle our proposed feature space is intended to subsume the feature space of the current state-of-the-art methods. The benchmark that we aimed for is to generalize the proposal by [8]. We do however introduce two additional features to inform the classification algorithm about the multi-sentence structure of the relation case: 1) the number of sentences that separate each entity mention pair, and 2) the number of intervening entity mentions between each entity mention pairing. The feature space is illustrated in Figure 4. We briefly summarize the features below.

Entity Mention Argument-based Features

A basic source of information about a relation case comes from the entity mention arguments themselves. The majority of algorithms in the literature make use of some of these features.

Entity Mention Tokens: This pair of features indicates the actual sequence of tokens used to signify each of the two entity mention arguments. For example, whether an entity mention uses the phrase “extracellular melieux” versus the synonymous phrase “outside the cell” can have an impact on the class label prediction. Each one of these pair of feature is a binary vector of all the words that act as entity mention arguments.

Entity Mention Semantic Class: This pair of features indicates the semantic class and subclass that each of the entity mentions is associated with. An entity can belong to zero or one semantic classes and to zero or one semantic subclass. For example, "TcpC" would be associated with the semantic class PROTEIN. This feature requires preprocessing by an entity mention recognition algorithm.

Subtree-based Features

To inform a classification algorithm about the structure of the relation case graph the subtree approach of [8] is followed. A feature is created for each possible neighborhood of the relation, where a neighborhood is defined by a subtree with e edges, where e ranges from zero through to some upper limit on edges: $e \in [0, e_{max}]$. The proposal in [8] is for $e_{max}=2$. Subtree-based features associated to the subtrees of size zero ($e=0$) simply summarize the number of nodes of a certain content type in either the entire relation case graph, or one of its pairings. For example, one feature would count the number of NP nodes in the relation case graph, while another feature would count the number of times that the word "required" is present. For the relation case graph represented in Figure 3 the "NP" feature would contain the value five (5) and the "required" feature the value one (1) for the pairing of the ORGANISM and LOCATION arguments. Subtree-based features associated to the subtrees of size $e>0$ represent the number of times that a subgraph with e edges appears within one of the paired entity instance subgraphs. For example, one feature would count the number of times that the triple IN – PP – NP appears in the graph. In the case of the graph in Figure 3 this feature would contain the value two (2) for the pairing of the ORGANISM and LOCATION arguments.

Intrasentential Features

This section introduces two features that are proposed to inform the classification algorithms about the multi-sentence structure of the relation. These features are novel to our proposal.

Sentence Count: This feature informs the classifier about the number of sentences that intervene between entity mentions. For example the number of intervening sentences between the ORGANISM and LOCATION arguments in the relation case in Figure 3 is two (2) sentences. This information will help the classifier adjust its predictions based on the separation: the further apart the less likely that a relation case is true.

Entity Mention's Sentence Location: Another related pair of features is simply the sentence identifier for each of the two entity mention pairs. For the example in Figure 3, the Organism entity mention is

located on the first sentence. This information will help the classifier adjust its predictions based on the sentence in which the entity is mentioned: the closer the mention is to the first sentence the less likely it is that an unlikely permutation is being considered.

Intervening Entity Mentions: This pair of features inform the classifier about the number of entities that intervene between two entity mention pairs. For example, in Figure 3 the number of intervening Location entity mentions between this case's Organism and Location entity mentions is two (2): from the first and second sentence. This information will help the classifier adjust its predictions based on how many other entity mention candidates exist. The greater the number of intervening entity mentions the less likely that a semantic relation between the two entity mentions is being stated.

Results and discussion

This section describes experiments performed to assess TeGRR's ability to detect biomedical semantic relation cases in natural language passages, specifically on the ternary relation of where an organism's protein localizes.

PPLRE: Prokaryote Protein Localization Relation Extraction

In partnership with the Brinkman Laboratory for Pathogen Bioinformatics, Genomics, and Interdisciplinary Studies[2] we have compiled a set of relation cases of experimentally confirmed subcellular localizations of Prokaryote proteins contained in a corpus of research paper abstracts found in PubMed[3]. The goal of our collaboration is to increase the number of experimentally validated localizations in the publicly available ePSORTdb[4] database [11] in order to improve the performance of classification models [6, 7] that are trained on known localizations in order to make predictions for proteins whose localization is currently unknown. A benefit of having more accurate localization data is that it allows biomedical researchers to make important insights into the function of proteins. In the case of bacterial pathogen proteins for example, the predictions can be used to expedite the identification of potential vaccine targets. Figure 5 presents the main localization targets that we labeled for Prokaryotes. The closest data resource that we know of which contains expert annotation of localization relation cases is the one published in [16][5]. Their data however is restricted to binary

[2] http://www.pathogenomics.sfu.ca/brinkman/
[3] http://www.ncbi.nlm.nih.gov/pubmed
[4] http://db.psort.org/docs/documentation.html#2
[5] http://www.biostat.wisc.edu/~craven/ie

relations because the task is limited to proteins from the *S. cerevisiae* yeast. Their data is also restricted to relations that are mentioned within a single sentence.

To identify relation cases for the domain experts to review, we first collected a subset of approximately 20,000 abstracts from PubMed based on queries that involved a known Prokaryote organism and a subcellular location. From these documents we identified passages that mention all three entity types. To accomplish this task the entities mentions were first labeled by a named entity recognition program. We used a hybrid approach composed of both a dictionary-based and classification-based[6] algorithm to recognize the organism and location entity mentions with high accuracy. The hybrid approach also achieved sufficiently high accuracy on the protein entities (F-measure 85%). This annotation program was used to annotate both the train and test data of the experiments; which is why the cooccurrence approach did not achieve 100% Recall. Currently the dataset is composed of 540 true and 4,769 false curated relation cases drawn from 843 research paper abstracts. Within this dataset 267 of the 540 true relation cases (~49%) span multiple sentences[7]. An advantage of a using a corpus based solely on abstracts rather than the entire paper is that it removes any concern for copyright infringement.

Performance

We tested the performance of the proposed feature space by means of stratified five-fold cross validation. Each document was randomly assigned to one test set and four train sets, unless the test set already contained one fifth of the positive cases, in which case it was randomly assigned to another test set. This stratified dispersal of records was intended to reduce the variance in performance between each of the five train/test runs. The basic measurements used for the tests were: Precision, Recall and F-measure. Performance of TeGRR is contrasted against three baseline algorithms: A cooccurrence-based algorithm that always predicts true for every permutation (All True), and then two binary relation single-sentence approaches proposed in [10] (YSRL) and [13] (Zparser) that separately train and test an ORGANISM/PROTEIN classifier and a PROTEIN/LOCATION classifier whose predictions are then combined. Table 1 summarizes the results.

---

[6] http://www.alias-i.com/lingpipe/

[7] Interestingly, in approximately half of these multi-sentence cases it is the protein mention which resides in a separate sentence.

	P	R	F
TeGRR	18.0%	47.5%	**26.1%**
YSRL	29.1%	13.3%	18.3%
Zparser	**63.5%**	9.3%	16.1%
All True	8.3%	**75.6%**	14.9%

Table 1 – Precision/Recall/F-measure by the proposed algorithm (TeGRR); by the modelling of two separate single-sentence models as proposed in [13] (Zparser) and in [10] (YSRL); and a naïve cooccurrence-based approach that predicts True for every permutation (All True).

As can be seen from the table, the proposed approach achieved the highest F-measure and the second highest Recall. The proposals by [10] (YSRL) and [13] (Zparser) achieved higher Precision but with significantly lower Recall[8]. Finally, as expected the "All True" cooccurrence approach achieved the highest Recall and the lowest Precision. A source of improvement in TeGRR's F-measure performance over YSRL and Zparser is due to TeGRR's ability to predict multi-sentence relation cases. Approximately one quarter of TeGRR's true-positive predictions (approximately 13 of 48 predictions) were from multi-sentence cases[9].

The performance reported in the table above suggests that recognizing all of the experimentally validated subcellular localizations in a PubMed abstract is a difficult task. One of the expected challenges of the task is the downstream effect of inaccuracies in the automated detection of protein mentions. If named entity recognition performance were perfect then the Cooccurrence-based approach would attain 100% Recall and the Recall of all the other approaches would also be raised. A more novel challenge to this task is the requirement that the relation mention must be for an experimentally validated claim. Many of the subcellular localization relations in the literature however are of background knowledge or are hypothesized. Distinguishing whether a paper reports an experiment or a hypothesis is likely to involve a significant amount of contextual information that is not currently addressed in relation recognition systems.

Conclusions

This paper addresses the challenge of recognizing mentions of relations with more than two arguments, where the argument's entity mentions can be located in different sentences. A motivating example is

---

[8] Note that these performance numbers for Zparser and YSRL differ markedly from the ones reported in their papers due to: 1) the inclusion of multi-sentence relation cases in our experiments; 2) the use of automated, not curated, NEs, 3) the use of both negative and positive relation cases, 4) the use of cross-validation; and 5) the use of a newer dataset.

[9] Approx. one third of the false-positive predictions (approx. 61 of 157) were on multi-sentence cases.

the ability to identify subcellular localization relations in biomedical research abstracts. For this ternary relation a large proportion of relation cases appear outside of the single sentence boundary. In general the more arguments in the semantic relation the more likely it will be that the relation is spread beyond a single sentence. To support these more complex relation detection scenarios we proposed a text-graph representation of the entire document. As in state-of-the-art supervised algorithms the intrasentential graph edges are derived from each sentence's syntactic parse trees. For intersentential edges we propose linking adjacent sentence edges and also, if available, entities that are identified as coreferents by a coreference resolution process. Compared to three baseline algorithms, the proposed approach achieves competitive F-measure and Recall performance. The paper suggests several avenues of future research into the area of detecting *n*-ary relations across multiple sentences. We plan to explore the question of adding more contextual features. It will also be instructive to explore the challenges that will arise when the framework is applied to full papers (rather than abstracts only) which will generate much larger and sparser text graphs.

Authors' contributions

To be added after blind review is completed.

Acknowledgements

Our thanks go to the team members of SFU's Brinkman Laboratory, particularly Nancy Yu, Matthew Laird, Sébastien Rey, Geoff Windsor and Fiona Brinkman for introducing us to a problem that exposed the challenge of multi-sentence n-ary relations in the first place, for their expert assistance with the manual labeling of subcellular localization relation cases, and for computational resources. Thanks also to Zhongmin Shi and Fred Popowich of SFU's Natural Language Laboratory for fruitful discussions particularly into relation detection from the NLP perspective, and for assistance with the automated annotation and manual curation of the PPLRE corpus. Finally, thanks to the anonymous reviewers and Jerre McQuinn for their helpful suggestions on ways to make the ideas in the paper more accessible.

References

1. Agichtein E, Gravano L: **Snowball: Extracting Relations from Large Plain-Text Collections**. *Procs. of the 5th ACM Int. Conf. on Digital Libraries*; 2000.
2. Appelt DE, Hobbs JR, Bear J, Israel DJ, Tyson M: **FASTUS: A Finite-state Processor for Information Extraction from Real-world Text**. *Procs. of IJCAI*; 1993.

3. Craven M, Kumlien J: **Constructing biological knowledge-bases by extracting information from text sources**. *Procs. of the Seventh International Conference on Intelligent Systems for Molecular Biology*; 1999.
4. Frietag D, McCallum A: **Information Extraction with HMMs and Shrinkage**. *AAAI'99 Workshop on Machine Learning for Information Extraction*; 1999.
5. Fundel K, Kuffner F, Zimmer R: **RelEx--relation extraction using dependency parse trees**. *Bioinformatics*, 23(3):365-71; 2007.
6. Gardy JL, Laird MR, Chen F, Rey S, Walsh CJ, Ester M, Brinkman FSL: **PSORTb v.2.0: expanded prediction of bacterial protein subcellular localization and insights gained from comparative proteome analysis**. *Bioinformatics*. 21(5):617-23; 2005.
7. Hoglund A, Blum T, Brady S, Donnes P, San Miguel J, Rocheford M, Kohlbacher O, Shatkay H: **Significantly improved prediction of subcellular localization by integrating text and protein sequence data**. *Pacific Symposium on Biocomputing*; 2006.
8. Jiang J, Zhai C: **A Systematic Exploration of the Feature Space for Relation Extraction**, *Procs. of NAACL/HLT*; 2007
9. Kambhatla N: **Combining lexical, syntactic, and semantic features with maximum entropy models for extracting relations**. *Proc. of ACL;* 2004.
10. Liu Y, Shi Z, Sarkar A: **Exploiting Rich Syntactic Information for Relation Extraction from Biomedical Articles**. *Procs. of NAACL/HLT*; 2007.
11. S. Miller, M. Crystal, H. Fox, L. Ramshaw, R. Schwartz, R. Stone, R. Weischedel, and the Annotation Group. (1998). **Algorithms that learn to extract information BBN: Description of the SIFT system as used for MUC-7**. *Procs of MUC-7*.
12. Rey S, Acab M, Gardy JL, Laird MR, deFays K, Lambert C, Brinkman FSL: **PSORTdb: a protein subcellular localization database for bacteria**. *Nucleic Acids Research 33:D164-168*; 2005.
13. Shi Z, Sarkar A, Popowich F: **Simultaneous Identification of Biomedical Named-Entity and Functional Relation Using Statistical Parsing Techniques.** *Procs. of NAACL/HLT*; 2007.

14. Stapley BJ, Benoit G: **Biobibliometrics: information retrieval and visualization from co-occurrences of gene names in Medline abstracts**. *Pacific Symposium on Biocomputing*; 2000.
15. Stapley BJ, Kelley LA, Sternberg MJ: **Predicting the sub-cellular location of proteins from text using support vector machines**. *Pacific Symposium on Biocomputing*; 2002.
16. Skounakis M, Craven M, Ray S: **Hierarchical Hidden Markov Models for Information Extraction**. *Procs. of IJCAI*; 2003.
17. Zhang M, Zhang J, Su J: **Exploring Syntactic Features for Relation Extraction using a Convolution Tree Kernel**. *Procs. of NAACL/HLT-2006*; 2006.

Figures

Figure 1 - Sample text graph

A sample text graph derived from three sentences drawn from the abstract of biomedical research paper PubMedID 15774863 "*Identification of a TcpC-TcpQ outer membrane complex involved in the biogenesis of the toxin-coregulated pilus of Vibrio cholerae*." The graph contains 52 intrasentential edges connecting 24 internal nodes and 32 leaf nodes. See the section "Text Graph Representation" for more details on the representation.

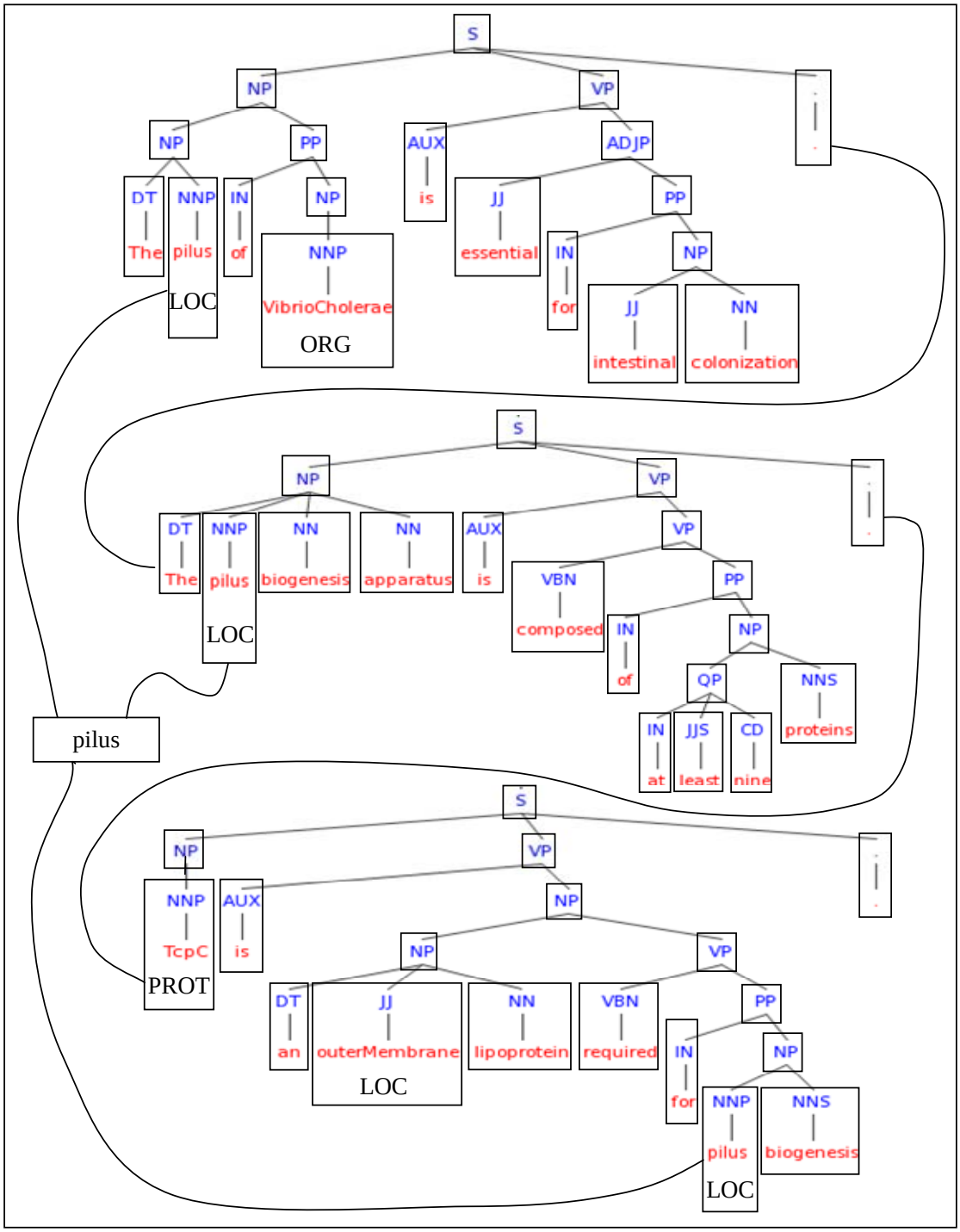

Figure 2 - Shortest-path Enclosed Tree

Example of a path enclosed tree where VibrioCholerae is one of the entities in the relation case and where the shortest path traverses through the pilus coreference edge. The encircled portion within the syntactic tree between the two entities is now attached to the relation case.

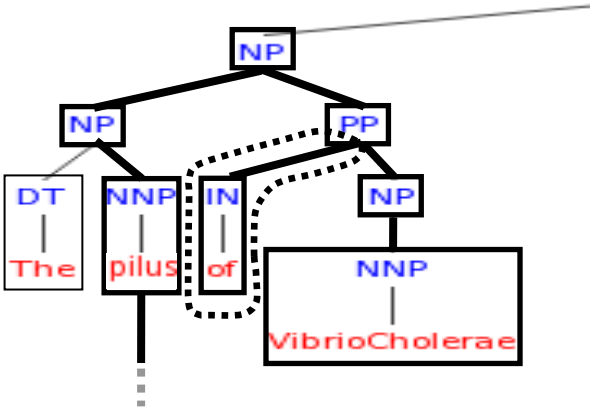

Figure 3 - Sample Relation Case Graph

A sample relation case graph used to describe the feature space. The case is drawn from the text graph represented in Figure 1 for the OPL() relation. The three entities in the case are represented as thick-lined square nodes, the rounded nodes represent nodes in a shortest path, and the thin-lined square nodes are syntactic nodes in the path-enclosed tree.

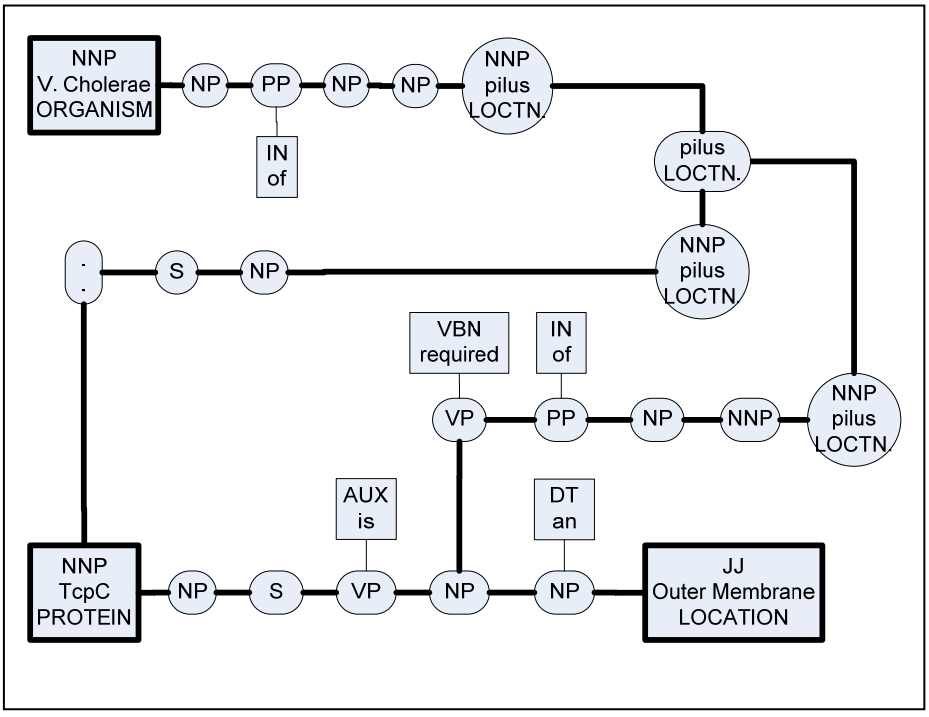

Figure 4 - Feature Space Illustration

A tabular representation of the feature space, relation case identifiers, and label assignment used for the ternary PPLRE task. Details of the features are presented in the “Feature Space” section.

Rel. Case				Feature Space																																label
				Organism - Protein											Protein - Location										Organism - Location											
				Intra.			Entity				Subtrees					Intra.			Entity				Subtrees					Intra.			Entity				Subtrees			
D	O_j	P_j	$L_{,j}$																																	
1	O_1	P_1	L_2	3	1	4	0	1	0	1	0	1	0	0	0	5	1	1	0	1	0	1	0	0	0	0	0	4	2	1	0	1	0	1	0	0	0	0	0	T
1	O_1	P_1	L_3	1	2	1	0	0	1	1	0	0	0	0	1	1	1	2	0	0	1	1	0	0	0	0	1	2	1	3	0	0	0	1	0	0	0	0	1	F
…	…	…	…	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	…
d	O_i	P_i	L_i	2	2	2	0	0	0	0	0	0	1	0	0	4	1	2	0	0	0	0	0	1	1	0	0	3	1	1	0	0	1	0	0	1	1	0	0	?

Figure 5 - Prokaryote Localization Compartments

The nine different subcellular locations that have been annotated in the PPLRE corpus