Reconstruction of semantic relationships from their projections in biomolecular domain

Juho Heimonen, Jari Björne, and **Tapio Salakoski**
University of Turku
Turku Centre for Computer Science and
Department of Information Technology
Joukahaisenkatu 3–5
20520 Turku, Finland
first.last@utu.fi

Abstract

The extraction of nested, semantically rich relationships of biological entities has recently gained popularity in the biomedical text mining community. To move toward this objective, a method is proposed for reconstructing original semantic relationship graphs from projections, where each node and edge is mapped to the representative of its equivalence class, by determining the relationship argument combinations that represent real relationships. It generalises the limited postprocessing step of the method of Björne et al. (2010) and hence extends this extraction method to arbitrarily deep relationships with unrestricted primary argument combinations. The viability of the method is shown by successfully extracting nested relationships in BioInfer and the corpus of the BioNLP'09 Shared Task on Event Extraction. The reported results, to the best of our knowledge, are the first for the nested relationships in BioInfer on a task in which only named entities are given.

1 Introduction

A recent shared task in biomedical text mining, the BioNLP'09 Shared Task on Event Extraction (Kim et al., 2009), showed that the biomedical natural language processing (BioNLP) community is greatly interested in heading towards the extraction of deep, semantically rich relationships. The shared task focused on biomolecular events involving proteins and called for methods that are capable of identifying nested structures. Biomolecular events are a major category of relationships in the biomedical domain in which, among others, relationships involving non-molecular entities such as diseases and static relations such as protein family memberships are also of interest.

Earlier, well-studied extraction tasks typically cast the problem in such a manner that relationships can be considered as mutually independent atomic units. However, as a nested semantic structure grows in its depth and in the total number of relationship arguments, its simultaneous extraction becomes difficult, if not impossible. Systems that bypass this problem by identifying atomic units of nested structures in a mutually independent manner must still decide which of the units collectively comprise a complete structure.

Another problem arises from the fact that a single syntactic token can refer to several, distinct relationships each having a unique combination of arguments. This is typically induced by coordinations which are common in the biomedical domain (Pyysalo et al., 2007). As a result, aside from the identification and classification of relationships and their potential arguments, extraction systems have to make decisions about how many relationships should be generated and how the arguments should be distributed among them. For example, the sentence "the binding of A and B to DNA regulates C and D, respectively" states that there are two binding events (*A–DNA* and *B–DNA*) the former of which regulates *C* and the latter *D* instead of, for example, that both binding events regulate both *C* and *D* or that there is a single binding event between *A*, *B*, and *DNA*.

This paper focuses on addressing the aforementioned problems in the case of the extraction method developed by Björne et al. (2010) for the BioNLP'09 Shared Task and generalises this method. Björne et al. showed that deep depen-

Proceedings of the 2010 Workshop on Biomedical Natural Language Processing, ACL 2010, pages 108–116,
Uppsala, Sweden, 15 July 2010.

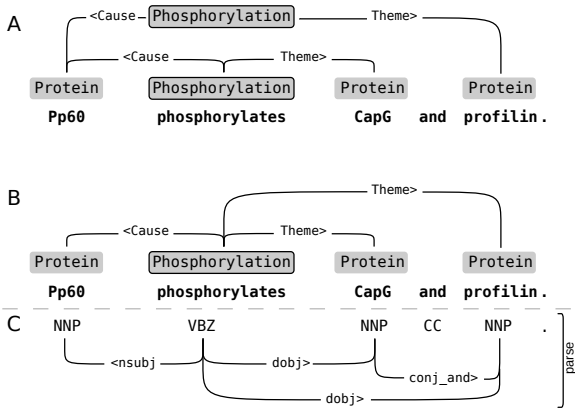

Figure 1: A one-node-per-token constrained graph (projected, B) cannot express the two distinct phosphorylation events while an unrestricted semantic graph (deprojected, A) can. A parse in the SD scheme is illustrated in C.

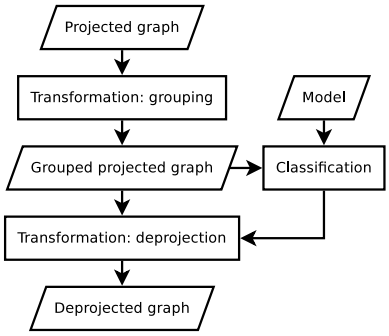

Figure 2: The deprojection process.

dency analyses in the well-established Stanford Dependency (SD) scheme (de Marneffe and Manning, 2008) can successfully be utilised in extracting graphs that express semantic entities as nodes and relationship arguments as edges but are limited to one node per syntactic token. Nodes and edges can be extracted in a mutually independent manner but the resulting graph cannot necessarily express all the real relationships. Rather, the graph can be seen as a projection of the original graph: each node and edge has been mapped to the representative of its equivalence class which is determined by the node and edge types and the referred tokens.

The research question of this paper is *can the original semantic graphs be reconstructed from projected graphs with an independent step in an information extraction (IE) process?* The objective of deprojection is illustrated as a transformation of the graph *B* to the graph *A* in Figure 1.

To answer the question, the problem of reconstructing complex, nested semantic structures from their projections is formulated and a generic deprojection method is proposed. The method specifically addresses primary arguments, as defined by the BioNLP'09 Shared Task, while leaving the extension to secondary arguments as a future work. The viability of the method is analysed with BioInfer (Pyysalo et al., 2007) and the BioNLP'09 Shared Task corpus, both of which containing nested structures, through an IE task essentially identical to the BioNLP'09 Shared Task. It is concluded that the proposed method can successfully augment the method of Björne et al. (2010) and generalise it to arbitrary graphs of nested biomolecular relationships without the strict restrictions of the BioNLP'09 Shared Task while retaining its performance level. Thus, the method can improve IE systems that produce relationships on the one-per-token basis.

2 Method

The proposed approach to deproject semantic graphs is outlined in Figure 2. In summary, the first transformation (*grouping*) alters a projected graph such that a minimal set of classes is sufficient to describe the behaviour of the nodes and the edges. Guided by predicted class labels, the second transformation (*deprojection*) then produces a deprojected graph. In the presented method, the classification problem is solved with machine-learning (ML) methods. Finally, corpus-specific constraints are enforced.

2.1 Definitions

The graph representation of semantic annotation introduced by Heimonen et al. (2008) is adopted with some additional definitions. Semantic knowledge is represented as a directed acyclic graph (DAG) as follows.

Nodes and edges correspond to semantic entities (such as protein and processes) and relationship arguments, respectively. The equality of nodes is determined by the equality of their types and of their references to text. Similarly, the equality of edges arises from the equality of their types and of their end nodes.

Shallow and deep relationships consist of a node, its outgoing edges, and its direct successors. The latter also recursively include the successor relationships. Nodes are equal as shallow relationships if they as well as their outgoing edges are

equal. Node equality as a deep relationship imposes the further requirement that the successors are equal as deep relationships.

A valid relationship is one which is valid in the given corpus-specific annotation scheme. Especially, it has a valid combination of arguments.

A deprojected graph (see Figure 1A) is one in which each node represents a valid, real relationship. Several equal nodes can exist provided they have unique combinations of outgoing edges. Note that there is one-to-one correspondence between nodes and real relationships but many-to-one between nodes and syntactic tokens.

A projected graph (see Figure 1B) is one generated by mapping each node and edge of a deprojected graph to the representative of its equivalence class. That is, each node represents a set of equal nodes of the deprojected graph, and similarly for edges. As a result, each token is referred to by at most one node[1] and there is a one-to-many correspondence between nodes and valid, real relationships. Also, the edges that are mapped to from the outgoing edges of equal nodes of the deprojected graph are the outgoing edges of a single node of the projected graph.

The deprojection of a semantic graph is the task of reproducing the original graph given a projected graph. This can also be seen as a task of finding the sets of outgoing edges that represent all the valid, real relationships.

2.2 Grouping

The objective of the first transformation is accomplished with a *grouping algorithm*: the direct successors of each node are grouped by their syntactic and semantic roles relative to the predecessor. The groups are represented as additional nodes in the graph. The rationale for this grouping is that similar arguments tend to either be mutually exclusive (and be associated with some other arguments) or together form a single relationship. This behaviour can easily be described with two classes: *distributive* and *collective*. For example, in the sentence "A and B regulate C", the entities *A* and *B* share both the argument type (*agent*) and the syntactic role (*subject*) relative to the relationship *regulate*. They form a group and are mutually exclusive (distributive) while this group forms a single relationship (collective) together with *C*. As a result, *A*–*C* and *B*–*C* pairs of regulation are generated. This approach relates to the collectivity and distributivity of plurals which have been studied, among others, by Scha and Stallard (1988) and Brisson (2003).

Technically, the grouping is a series of transformations in each of which a set of successors is replaced with a single, newly-created successor and the original successors become the successors of this node. The successors are first trivially grouped by the corresponding edge type. Finally, they are recursively grouped by syntactic similarity until they form a single group or multiple singleton groups. As a result, nested groups are generated.

The groups by syntax are determined by first mapping both the predecessor and the successors into the referred tokens in the syntactic graph. Then, the tokens referred to by the predecessor are removed if they are not also referred to by any of the successors. This removal step is recursively applied to the predecessors of the removed tokens. As a result, the syntactic graph is decomposed into several connected components, each of which representing a group. Thus, two successors are grouped if their referred tokens belong to the same connected component.

2.3 Deprojection

The second transformation is guided by node class labels (Figure 3). A collective node remains unchanged: its successors are kept together. In contrast, a distributive node is duplicated for each outgoing edge and the edges are distributed, one edge per duplicate. These node classes are enough to solve most of the cases in the analysed data sets. However, especially in BioInfer, this is not sufficient since the duplicates of a distributive node may themselves be either collective or distributive under their predecessor.

To adequately describe the behaviour of the duplicates generated by a single distributive node, the incoming edges of each distributive node are classified as *collective* or *distributive* (Figure 4). The duplication of a node also duplicates its incoming edges which are then processed by the assigned class labels as follows. In the case of a collective edge, the generated duplicates of the edge share the predecessor and are thus arguments in a single relationship. In contrast, a distributive edge induces the duplication of the predecessor re-

[1] given that, in the deprojected graph, a token can be referred to by multiple nodes only if they are of the same type

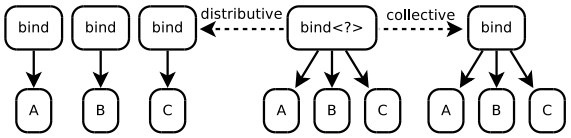

Figure 3: The effect of assigning collective or distributive class labels (marked as <?>) to a node in the deprojection process.

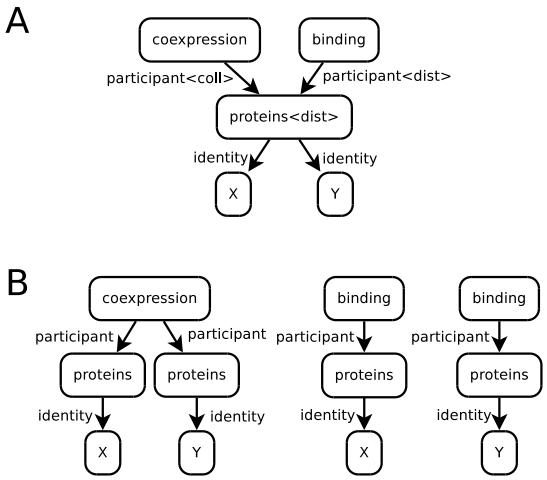

Figure 4: Correct node and edge class labels for the projected graph of the phrase "Coexpression and subsequent DNA-binding of X and Y proteins" (A) and the resulting deprojected graph (B).

lationships such that, as a result, the generated duplicate edges do not share any predecessors.

In Figure 4A, the node *proteins* is distributive because it represents two distinct nodes: one pertaining to *X*, another to *Y*. These two nodes are involved in the same *coexpression* relationship but in different *binding* relationships. Hence the incoming edges of the node *proteins* are collective and distributive, respectively.

Since the two transformation steps do not enforce corpus-specific constraints, a trivial algorithm is utilised to decompose relationships with invalid argument combinations into multiple valid relationships. In an ideal situation, this step makes no transformations. This is also used as a part of the baseline method (see Section 3.3).

2.4 Machine-learning and features

For node and edge classifications, the C4.5 decision tree (Quinlan, 1993), and its J48 implementation in the Weka package, was utilised because its models can easily be examined. This facilitates the analysis of the problem and the further development of the solution. The default parameters were used since no improvement was gained with alternative parameters in preliminary experiments. The applied feature set emphasises higher-level features obtained from the semantic and syntactic graphs. It consists of three main groups: semantic, syntactic, and morphological.

Semantic features contain information gathered from the semantic graph as well as from the type hierarchies. For nodes, these features consists of the node type as well as the presence, count and combination of outgoing edge types. The count of successor groups and the distance to the first non-group predecessor are also included. For edges, the node features are generated for both the successor and the predecessor in addition to the type of the edge.

Syntactic features include the minimum syntactic distance[2] from the predecessor to the successors as well as between the successors. Also, in the case of the unit distance, the corresponding dependency type is included.

Morphological features consist of the Porter stems (Porter, 1980) and the part-of-speech tags of the referred tokens as well as the presence and the Porter stems of the tokens that are shared between the successors.

All features are also generated from the first non-group predecessor (which may be the node itself) to capture the original relationship node when processing a group node. The majority of the features are Boolean-valued in order to allow several values of a single property. This is utilised in features representing hierarchical knowledge (such as node and dependency types) as well as stem features. For example, a node receives *true* for the node type feature of its actual type as well as of its supertypes in the hierarchy.

3 Resources and experiments

An array of experiments was performed to analyse the deprojection problem and the proposed solution. Firstly, the same experiments were performed on two corpora, BioInfer and the BioNLP'09 Shared Task corpus in order to evaluate the effect of the annotation scheme to the properties of the problem. Secondly, the deprojection algorithm was applied to both projected gold-standard graphs and to predicted graphs in order to study the effect of the accuracy of the input graph. Thirdly, the effect of the quality of the parse was

[2] semantic nodes mapped into the referred tokens

examined by employing various parses including the BioInfer gold-standard annotation.

3.1 Data

BioInfer is a corpus of 1100 sentences selected from 836 publication abstracts available through PubMed. For this paper, the abstracts were randomly sampled in the ratio 2:1:1 into the training, development, and test sets. In contrast, the BioNLP'09 Shared Task corpus consists of the training, development, and test sets of 800, 150, and 260 abstracts, respectively. Since the annotation of the test set is not publicly available and the evaluation server does not provide the required details for the analysis, the development set was used as the test set and a random sample of 150 abstracts was cut from the training set to form the development set.

In this study, the task 1 annotation with the protein equivalence relations removed was used as the BioNLP'09 Shared Task data set. In this annotation, relationships are positively asserted, have only *Theme* and *Cause* arguments and are annotated only for one of the equivalent proteins. Furthermore, each node refers to at least one token in the syntactic graph. The BioInfer semantic annotation was transformed into a similar form by removing negation (*NOT*), equivalence (*EQUAL*), and reference nodes (*COREFER*, *REL-ENT*). Furthermore, to create a fully text-bound subset, family memberships relations (*MEMBER*) were resolved into single edges and suitable references to text were added for the remaining unbound nodes when possible. In an extreme case, an unbound relationship was discarded. As a result, the differences to the BioNLP'09 Shared Task data set were minimised to additional node and edge types reflecting the wider selection of primary arguments.

All employed parses follow the SD scheme. BioInfer contains uncollapsed gold-standard parses while the BioNLP'09 Shared Task corpus includes parses, in the collapsed representation, generated by the parser of Charniak and Johnson (2005) using the model of McClosky and Charniak (2008). For both corpora, additional parses were produced with the improved version of the aforementioned system created by McClosky (2009). These parses were transformed into both the collapsed and the conjunct dependency propagated representations with the tools provided by de Marneffe et al. (2006). All parses were further augmented by splitting tokens at non-alphanumeric characters that border named entities and connecting the newly-created tokens with dependencies denoting the character.

3.1.1 Predicted graphs

The predicted semantic graphs were obtained as a result of an extraction task adopted from the BioNLP'09 Shared Task. In this task, named entities are given as gold-standard annotation and their relationships are to be extracted by identifying text spans, determining types, and assigning arguments.

The predicted graphs were produced with the system developed for the BioNLP'09 Shared Task by Björne et al. (2010). The system has two machine-learning steps. First, relationship nodes are predicted, one per token, based only on the syntax and the given named entity nodes. Next, outgoing edges are predicted for the relationship nodes. As a result, a projected graph is obtained. With the graph representation, the system can transparently be trained for both the BioNLP'09 Shared Task corpus and BioInfer regardless of the differences in their annotation schemes.

The two prediction steps utilise the SVM^{multiclass} implementation of a multi-class support vector machine (Crammer and Singer, 2002; Tsochantaridis et al., 2004). In this study, the steps were independently optimised for model parameters and, in contrast to the original training procedure, the recall boosting optimisation was omitted due to limited resources available. When training the edge prediction, the gold-standard relationship nodes were used.

In the graph prediction, the conjunct dependency propagated parses produced with the parser of McClosky (2009) were systematically applied.

3.2 Experiments

Original gold-standard graphs were used in generating decision tree models as well as subjected to projection. Predicted graphs and the projected gold-standard graphs were deprojected with the models. The evaluation of the deprojected graphs was performed against the original graphs.

During the system development, the training and development sets were available and the data were thoroughly analysed. The progress was estimated by training the system with the former and testing against the latter. The final results were obtained on the test sets by applying the system

trained on the combined training–development set. For analysis, also the baseline method and the method of Björne et al. (2010) were evaluated on the test sets.

3.3 Baselines

The baselines were designed to reflect an IE system following the one-node-per-token principle without an advanced postprocessing but still enforcing the annotation scheme constraints.

With the strict specifications of the BioNLP'09 Shared Task, a sound baseline is obtained simply by enforcing the constraints through a minimal set of changes. Nodes with outgoing *Cause* and *Theme* edges are duplicated into all *Cause–Theme* pairs. *Binding* nodes remain unchanged since they can have several *Theme* arguments while the others are treated as distributive nodes with distributive incoming edges.

Although BioInfer is less restricted with respect to valid argument combinations, a feasible baseline can be obtained by adapting the BioNLP'09 Shared Task baseline algorithm. *Cause–Theme* is replaced with *agent–patient* while *Binding* is extended to symmetric relationships (i.e. *participant* arguments). In addition, relationships with *sub* arguments are treated as collective which reflects multiple components in a single complex. These changes were also applied to the method of Björne et al. (2010) when analysing BioInfer.

3.4 Evaluation

The standard precision–recall–F_1 metrics was used in the evaluation. True/false positive/negative instances were determined by the equality of the nodes as relationships: pairs of equal nodes were true positives while unique nodes in the deprojected and the original graph were false positives and false negatives, respectively.

The equality of references to text was determined after removing the tokens found in a non-exhaustive list of common stop-words including prepositions, articles, and non-alphanumeric characters. This relaxes an unnecessary requirement of the node prediction step to find also those tokens in the BioInfer annotation that do not contribute to the semantics of the nodes. For example, prepositions should be associated with edges rather than nodes.

The F_1-scores were further analysed with the Wilcoxon signed-rank test (Wilcoxon, 1945), as implemented in Scipy v. 0.7.0, by considering each document as an experiment and using the 95% confidence level.

BioInfer

	gold		predicted	
method	total	symm.	total	symm.
baseline	88.26	63.62	29.38	18.64
Björne et al.	89.15	72.35	29.14	20.37
proposed	**92.42**	**78.79**	**30.79**	**24.47**

BioNLP'09

	gold		predicted	
method	total	symm.	total	symm.
baseline	92.52	64.15	43.70	21.05
Björne et al.	94.51	83.37	45.13	35.21
proposed	**95.08**	**84.32**	**45.32**	**36.63**

Table 1: The F_1-scores on the test sets. *Total* is cumulative over all nodes with outgoing edges while *symm.* refers to the symmetric types. *Gold* and *predicted* refer to the experiments with gold-standard and predicted graphs, respectively.

4 Results and discussion

The following discussion focuses on the deep relationship equality as the evaluation criterion because it reflects the relationships of interest by requiring the identification of the pertaining named entities. Also, the discussion only considers the experiments performed with the conjunct dependency propagated parses obtained with the parser of McClosky (2009) because switching parses did not produce statistically significant differences in performance. Note that the results are not comparable to those of Björne et al. (2010) because the graph prediction was not fully optimised.

With respect to the deprojection task, BioInfer was found to be similar to the BioNLP'09 Shared Task corpus: it contains symmetric relationships (c.f. *Binding*), asymmetric relationships (c.f. *Regulation*), and single-argument relationships. Only the symmetric relationships are a challenge in the deprojection task because they can have an arbitrary number of arguments. In contrast, the baseline F_1-scores for the others are above 94% on the gold-standard graphs.

Table 1 shows the F_1-scores on the test sets of BioInfer and the BioNLP'09 Shared Task corpus for the overall performance as well as for the symmetric relationships only. The proposed method outperforms the two other methods in all

experiments and the ΔF_1 against the proposed method are statistically significant with the exception of the method of Björne et al. (2010) on the BioNLP'09 Shared Task corpus. Although not conclusively better than the earlier, specialised method in its own task, the proposed method successfully achieves the intended generalisation without an adverse effect.

The observed improvement over the method of Björne et al. (2010) is likely due to two factors. First, using machine-learning rather than a simple rule-based system allows for more accurate modelling of the problem. Second, the proposed method can handle a wider variety of cases due to the classification of edges. For example, the graph in Figure 4A can correctly be deprojected, which is not possible for the earlier method. However, the latter factor is only effective on BioInfer, the more complex of the two corpora, which is consistent with the observed statistical significances.

The proposed deprojection method is currently limited to the phenomena encountered in the two analysed corpora since the decision to use binary classification was based on the experimental observation that neither class is appropriate only in rare cases. More classes will be needed to further generalise and improve the system. One such class could be *respective* which denotes a selective pairing of sibling nodes. For example, the sentence "A and B binds C and D, respectively" currently results in false positive pairs *A–D* and *B–C*. Similarly, adding secondary arguments (e.g. *location*) and relationship modifiers (e.g. *negation*) into consideration is likely to necessitate new, more complex transformations and their respective classes. Also, to filter out incorrectly predicted edges will require the introduction of additional classes. The critical question is whether a reasonably small set of classes with extensive enough a coverage can be found.

Another limitation is that the approach expects an annotation scheme in which relationship arguments have the tendency of following syntactic dependencies as observed for BioInfer by Björne et al. (2008). This expectation may deteriorate the performance on highly refined schemes which do not consider syntax. On the other hand, since it relies more on the syntactic than on the biological properties of the relationships, the proposed approach should be applicable beyond the domain of biomolecular events (e.g. to gene–disease relationships or static relations).

The F_1-scores in Table 1 indicate that the BioNLP'09 Shared Task corpus is easier to extract than BioInfer. This is likely due to the narrower scope and the stricter constraints of the former. In absolute terms, the proposed method yields the largest improvement over the baseline on the gold-standard graphs which suggests that it is negatively affected by the presence of false nodes/edges or that the predicted graphs contain relatively more relationships that are trivially deprojected. On the other hand, in relative terms, the largest improvements are observed for symmetric relationships in the BioNLP'09 Shared Task corpus but overall in BioInfer. This is likely due to the differences in the relationship type distributions.

The system recently developed by Miwa et al. (2010), based on the architecture of Björne et al. (2010), utilises a ML-based deprojection which enumerates all possible argument combinations and classifies them as positive or negative. While this approach may be prohibitively expensive in more complex schemes in which the number of arguments and their types is higher, it should outperform the proposed method on the BioNLP'09 Shared Task corpus. Since Miwa et al. do not analyse the contribution of the deprojection to the overall performance, a direct comparison of the two methods is impossible. In any case, the systems of Björne et al. (2010) and Miwa et al. (2010) demonstrate the success of the architecture using deprojection and further motivate the investigation of deprojection methods.

4.1 Future directions

In the future, the proposed method will be studied and further improved with two other corpora, GENIA Event Annotation (Kim et al., 2008) and Gene Regulation Event Corpus (Thompson et al., 2009), which are similar in their purpose compared to the already-analysed corpora. The former corpus is interesting because of the co-operativity of event participants which relaxes the restrictions on asymmetric relationships while the latter contains an extensive annotation for non-primary arguments. The method could also be examined with the static relation extraction task recently introduced by Pyysalo et al. (2009).

In addition to improving the method and extending it to non-primary arguments, embedding

the presented approach to a joint inference system, such as Markov Logic Network (MLN), will be studied. Deprojection is likely to greatly benefit methods based on Markov Logic which is "not yet able to deal with cases where the number and identity of entities is unknown, while relations/links between known objects can be readily modelled" (Riedel et al., 2009). The objective is to combine the graph prediction and the deprojection steps as well as to simultaneously enforce task-specific constraints and adapt to the presence of false positive nodes and edges. This should be achievable by extending the methods developed for the BioNLP'09 Shared Task corpus by Riedel et al. (2009) or by Poon and Vanderwende (2010), both of which determine the correct argument combinations outside of the Markov Logic framework.

Semantic role labelling (SRL) is a task similar to the graph-based relationship extraction applied in this paper although the former typically only concerns shallow predicate–argument structures (Hacioglu, 2004; Surdeanu et al., 2008). The similarities between the tasks suggest that exploring them jointly may benefit the development of information extraction methods.

In the long term, semantic schemes should be developed such that, ideally, all syntactic tokens are considered for their semantics and semantic relationships readily follow from their dependencies. Such schemes, closely following the syntax, could improve both the graph prediction and the deprojection. In this research direction, graph-based knowledge representations such as conceptual graphs (Sowa, 1976; Chein and Mugnier, 2008) or graphical logical forms such as the one proposed by Allen et al. (2008) could be adopted.

Given the frequency of coordinations in the biomedical domain, deprojection may prove to be useful in the development of deep semantic parsing in the biomedical domain. For example, with improved semantic schemes, it could provide a means to generate complete, detailed semantic graphs directly from deep dependency analyses in a single-step by applying joint inference to achieve simultaneous node/edge relabelling and graph deprojection.

5 Conclusions

This study presents a method for reconstructing the original semantic graphs from their projections by determining the correct combinations of relationship arguments. It generalises the postprocessing step of the system described by Björne et al. (2010) and extends the extraction capability of this system to arbitrary graphs of nested biomolecular relationships. The evaluation of the method on BioInfer and the BioNLP'09 Shared Task corpus indicates that the approach is viable for primary relationship arguments. For BioInfer, the outcome is, to the best of our knowledge, the first reported result of the task of extracting the nested relationships in its original version.

The presented method facilitates an IE approach in which the identification of semantic entities is performed on the one-entity-per-token basis and relationship arguments are identified in a mutually independent manner disregarding the semantics of the argument combinations. The method handles the selection of the correct argument combinations, which is non-trivial particularly when coordinations are involved, and generates the final output in which a single token can refer to several entities. This approach improves the utilisation of deep dependency analyses by simplifying the correlation between them and semantic graphs. Due to its independent nature, the method can be coupled to any system identifying relationships on the one-per-token basis.

The implemented system will be available upon request.

Acknowledgements

Thanks to Filip Ginter for his help with the parses. This study was funded by Academy of Finland. Computational resources were provided by CSC – IT Center for Science.

References

J. Allen, M. Swift, and W. de Beaumont. 2008. Deep semantic analysis of text. In *Proceedings of the 2008 Conference on Semantics in Text Processing (STEP'08)*, pages 343–354.

J. Björne, S. Pyysalo, F. Ginter, and T. Salakoski. 2008. How complex are complex protein-protein interactions? In *Proceedings of the Third International Symposium on Semantic Mining in Biomedicine (SMBM'08)*, pages 125–128.

J. Björne, J. Heimonen, F. Ginter, A. Airola, T. Pahikkala, and T. Salakoski. 2010. Extracting contextualized complex biological events with rich graph-based feature sets. *Computational Intelligence*. To appear.

C. Brisson. 2003. Plurals, all, and the nonuniformity of collective predication. *Linguistics and Philosophy*, 26:129–184.

E. Charniak and M. Johnson. 2005. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 173–180.

M. Chein and M.-L. Mugnier. 2008. *Graph-based Knowledge Representation: Computational Foundations of Conceptual Graphs*. Springer Publishing Company Inc.

K. Crammer and Y. Singer. 2002. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 2:265–292.

M.-C. de Marneffe and C. Manning. 2008. The stanford typed dependencies representation. In *Proceedings of the Coling'08 Workshop on Cross-Framework and Cross-Domain Parser Evaluation*, pages 1–8.

M.-C. de Marneffe, B. MacCartney, and C. Manning. 2006. Generating typed dependency parses from phrase structure parses. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)*, pages 449–454.

K. Hacioglu. 2004. Semantic role labeling using dependency trees. In *Proceedings of the 20th International Conference on Computational Linguistics (COLING'04)*, pages 1273–1276.

J. Heimonen, S. Pyysalo, F. Ginter, and T. Salakoski. 2008. Complex-to-pairwise mapping of biological relationships using a semantic network representation. In *Proceedings of the Third International Symposium on Semantic Mining in Biomedicine (SMBM'08)*, pages 45–52.

J.-D. Kim, T. Ohta, and J. Tsujii. 2008. Corpus annotation for mining biomedical events from literature. *BMC Bioinformatics*, 9:10.

J.-D. Kim, T. Ohta, S. Pyysalo, Y. Kano, and J. Tsujii. 2009. Overview of bionlp'09 shared task on event extraction. In *Proceedings of the NAACL–HLT'09 Workshop on BioNLP: Companion Volume for Shared Task (BioNLP'09)*, pages 1–9.

D. McClosky and E. Charniak. 2008. Self-training for biomedical parsing. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Short Papers (ACL'08: HLT)*, pages 101–104.

D. McClosky. 2009. *Any Domain Parsing: Automatic Domain Adaptation for Natural Language Parsing*. Ph.D. thesis, Department of Computer Science, Brown University, Providence, Rhode Island, USA.

M. Miwa, R. Saetre, J.-D. Kim, and J. Tsujii. 2010. Event extraction with complex event classification using rich features. *Journal of Bioinformatics and Computational Biology*, 8:131–146.

H. Poon and L. Vanderwende. 2010. Joint inference for knowledge extraction from biomedical literature. In *Proceedings of Human Language Technologies: The 11th Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL–HLT'10)*.

M. Porter. 1980. An algorithm for suffix stripping. *Program*, 14:130–137.

S. Pyysalo, F. Ginter, J. Heimonen, J. Björne, J. Boberg, J. Järvinen, and T. Salakoski. 2007. BioInfer: a corpus for information extraction in the biomedical domain. *BMC Bioinformatics*, 8:50.

S. Pyysalo, T. Ohta, J.-D. Kim, and J. Tsujii. 2009. Static relations: a piece in the biomedical information extraction puzzle. In *Proceedings of the NAACL–HLT'09 Workshop on BioNLP (BioNLP'09)*, pages 1–9.

J. Quinlan. 1993. *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc.

S. Riedel, H.-W. Chun, T. Takagi, and J. Tsujii. 2009. A markov logic approach to bio-molecular event extraction. In *Proceedings of the NAACL–HLT'09 Workshop on BioNLP: Companion Volume for Shared Task (BioNLP'09)*, pages 41–49.

R. Scha and D. Stallard. 1988. Multi-level plurals and distributivity. In *Proceedings of the 26th Annual Meeting of the Association for Computational Linguistics (ACL'88)*, pages 17–24.

J. Sowa. 1976. Conceptual graphs for a data base interface. *IBM Journal of Research and Development*, 20:336–357.

M. Surdeanu, R. Johansson, A. Meyers, L. Màrquez, and J. Nivre. 2008. The CoNLL 2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of the Twelfth Conference on Computational Natural Language Learning (CoNLL'08)*, pages 159–177.

P. Thompson, S. Iqbal, J. McNaught, and S. Ananiadou. 2009. Construction of an annotated corpus to support biomedical information extraction. *BMC Bioinformatics*, 10:349.

I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. 2004. Support vector machine learning for interdependent and structured output spaces. In *Proceedings of the 21st International Machine Learning Conference (ICML'04)*, page 104.

F. Wilcoxon. 1945. Individual comparisons by ranking methods. *Biometrics Bulletin*, 1:80–83.