A memory–based learning approach to event extraction in biomedical texts

Roser Morante, Vincent Van Asch, Walter Daelemans
CNTS - Language Technology Group
University of Antwerp
Prinsstraat 13
B-2000 Antwerpen, Belgium
{Roser.Morante,Walter.Daelemans,Vincent.VanAsch}@ua.ac.be

Abstract

In this paper we describe the memory-based machine learning system that we submitted to the BioNLP Shared Task on Event Extraction. We modeled the event extraction task using an approach that has been previously applied to other natural language processing tasks like semantic role labeling or negation scope finding. The results obtained by our system (30.58 F-score in Task 1 and 29.27 in Task 2) suggest that the approach and the system need further adaptation to the complexity involved in extracting biomedical events.

1 Introduction

In this paper we describe the memory-based machine learning system that we submitted to the BioNLP shared task on event extraction[1]. The system operates in three phases. In the first phase, event triggers and entities other than proteins are detected. In the second phase, event participants and arguments are identified. In the third phase, postprocessing heuristics select the best frame for each event.

Memory-based language processing (Daelemans and van den Bosch, 2005) is based on the idea that NLP problems can be solved by reuse of solved examples of the problem stored in memory. Given a new problem, the most similar examples are retrieved, and a solution is extrapolated from them. As language processing tasks typically involve many subregularities and (pockets of) exceptions, it has been argued that memory-based learning is at an advantage in solving these highly disjunctive learning problems compared to more eager learning that abstract from the examples, as the latter eliminates not only noise but also potentially useful exceptions (Daelemans et al., 1999).

The BioNLP Shared Task 2009 takes a linguistically-motivated approach, which is reflected in the properties of the shared task definition: rich semantics, a text-bound approach, and decomposition of linguistic phenomena. Memory-based algorithms have been successfully applied in language processing to a wide range of linguistic tasks, from phonology to semantic analysis. Our goal was to investigate the performance of a memory–based approach to the event extraction task, using only the information available in the training corpus and modelling the task applying an approach similar to the one that has been applied to tasks like semantic role labeling (Morante et al., 2008) or negation scope detection (Morante and Daelemans, 2009).

In Section 2 we briefly describe the task. Section 3 reviews some related work. Section 4 presents the system, and Section 5 the results. Finally, some conclusions are put forward in Section 6.

2 Task description

The BioNLP Shared Task 2009 on event extraction consists of recognising bio-molecular events in biomedical texts, focusing on molecular events involving proteins and genes. An event is defined as a relation that holds between multiple entities that fulfil different roles. Events can participate in one type

[1]Web page: http://www-tsujii.is.s.u-tokyo.ac.jp/GENIA/SharedTask/index.html

Proceedings of the Workshop on BioNLP: Shared Task, pages 59–67,
Boulder, Colorado, June 2009.

of events: regulation events.

The task is divided into the three subtasks listed below. We participated in subtasks 1 and 2.

- Task 1: event detection and characterization. This task involves event trigger detection, event typing, and event participant recognition.
- Task 2: event argument recognition. Recognition of entities other than proteins and the assignment of these entities as event arguments.
- Task 3: recognition of negations and speculations.

The task did not include a named entity recognition subtask. A gold standard set of named entity annotations for proteins was provided by the organisation. A dataset based on the publicly available portion of the GENIA (Collier et al., 1999) corpus annotated with events (Kim et al., 2008) and of the BioInfer (Pyysalo et al., 2007) corpus was provided for training, and held-out parts of the same corpora were provided for development and testing.

The inter-annotator agreement reported for the Genia Event corpus is 56% strict match[2], which means that the event type is the same, the clue expressions are overlapping and the themes are the same. This low inter-annotator agreement is an indicator of the complexity of the task. Similar low inter-annotator agreement rates (49.00 %) in identification of events have been reported by Sasaki et al. (2008).

3 Related work

In recent years, research on text mining in the biomedical domain has experienced substantial progress, as shown in reviews of work done in this field (Krallinger and Valencia, 2005; Ananiadou and McNaught, 2006; Krallinger et al., 2008b). Some corpora have been annotated with event level information of different types: PropBank-style frames (Wattarujeekrit et al., 2004; Chou et al., 2006), frame independent roles (Kim et al., 2008), and specific roles for certain event types (Sasaki et al., 2008). The focus on extraction of event frames using machine learning techniques is relatively new because there were no corpora available.

[2]We did not find inter-annotator agreement measures in the paper that describes the corpus (Kim et al., 2008), but in `www-tsujii.is.s.u-tokyo.ac.jp/T-FaNT/T-FaNT.files/Slides/Kim.pdf`.

Most work focuses on extracting biological relations from corpora, which consists of finding associations between entities within a text phrase. For example, Bundschus et al. (2008) develop a Conditional Random Fields (CRF) system to identify relations between genes and diseases from a set of GeneRIF (Gene Reference Into Function) phrases. A shared task was organised in the framework of the Language Learning in Logic Workshop 2005 devoted to the extraction of relations from biomedical texts (Nédellec, 2005). Extracting protein-protein interactions has also produced a lot of research, and has been the focus of the BioCreative II competition (Krallinger et al., 2008a).

As for event extraction, Yakushiji et al. (2001) present work on event extraction based on full-parsing and a large-scale, general-purpose grammar. They implement an Argument Structure Extractor. The parser is used to convert sentences that describe the same event into an argument structure for this event. The argument structure contains arguments such as semantic subject and object. Information extraction itself is performed using pattern matching on the argument structure. The system extracts 23 % of the argument structures uniquely, and 24% with ambiguity. Sasaki et al. (2008) present a supervised machine learning system that extracts event frames from a corpus in which the biological process *E. coli gene regulation* was linguistically annotated by domain experts. The frames being extracted specify all potential arguments of gene regulation events. Arguments are assigned domain-independent roles (Agent, Theme, Location) and domain-dependent roles (Condition, Manner). Their system works in three steps: (i) CRF-based named entity recognition to assign named entities to word sequences; (ii) CRF-based semantic role labeling to assign semantic roles to word sequences with named entity labels; (iii) Comparison of word sequences with event patterns derived from the corpus. The system achieves 50% recall and 20% precision.

We are not aware of work that has been carried out on the data set of the BioNLP Shared Task 2009 before the task took place.

4 System description

We developed a supervised machine learning system. The system operates in three phases. In the first phase, event triggers and entities other than proteins are detected. In the second phase, event participants and arguments are identified. In the third phase, postprocessing heuristics select the best frame for each event. Parameterisation of the classifiers used in Phases 1 and 2 was performed by experimenting with sets of parameters on the development set. We experimented with manually selected parameters and with parameters selected by a genetic algorithm, but the parameters found by the genetic algorithm did not yield better results than the manually selected parameters

As a first step, we preprocess the corpora with the GDep dependency parser (Sagae and Tsujii, 2007) so that we can use part-of-speech tags and syntactic information as features for the machine learner. GDep is a a dependency parser for biomedical text trained on the Tsujii Lab's GENIA treebank. The dependency parser predicts for every word the part-of-speech tag, the lemma, the syntactic head, and the dependency relation. In addition to these regular dependency tags it also provides information about the IOB-style chunks and named entities. The classifiers use the output of GDep in addition to some frequency measures as features.

We represent the data into a columns format, following the standard format of the CoNLL Shared Task 2006 (Buchholz and Marsi, 2006), in which sentences are separated by a blank line and fields are separated by a single tab character. A sentence consists of tokens, each one starting on a new line.

4.1 Phase 1: Entity Detection

In the first phase, a memory based classifier predicts for every word in the corpus whether it is an entity or not and the type of entity. In this setting, *entity* refers to what in the shared task definition are events and entities other than proteins. Classes are defined in the IOB-style[3] in order to find entities that span over multiple words. Figure 1 shows a simplified version of a sentence in which *high level* is a Positive Regulation event that spans over multiple tokens and *proenkephalin* is a Protein. The Protein class does not need to be predicted by the classifier because this information is provided by the Task organisers. The classes predicted are: O, {B,I}-Entity, {B,I}-Binding, {B,I}-Gene Expression, {B,I}-Localization, {B,I}-Negative Regulation, {B,I}-Positive Regulation, {B,I}-Phosphorylation, {B,I}-Protein Catabolism, {B,I}-Transcription.

Token	Class	Token	Class
Upon	O	which	O
activation	O	correlate	O
,	O	with	O
T	O	high	B-Positive_regulation
lymphocyte	O	level	I-Positive_regulation
accumulate	O	of	O
high	O	proenkephalin	B-Protein
level	O	mRNA	O
of	O	in	O
the	O	the	O
neuropeptide	O	cell	O
enkephalin	O	.	O

Figure 1: Instance representation for the entity detection classifier.

We use the IB1 memory–based classifier as implemented in TiMBL (version 6.1.2) (Daelemans et al., 2007), a supervised inductive algorithm for learning classification tasks based on the k-nearest neighbor classification rule (Cover and Hart, 1967). The memory-based learning algorithm was parameterised in this case by using modified value difference as the similarity metric, gain ratio for feature weighting, using 7 k-nearest neighbors, and weighting the class vote of neighbors as a function of their inverse linear distance. For training we did not use the entire set of instances from the training data. We downsampled the instances keeping 5 negative instances (class label O) for every positive instance. Instances to be kept were randomly selected. The features used by this classifier are the following:

- About the token in focus: word, chunk tag, named entity tag as provided by the dependency parser, and, for every entity type, a number indicating how many times the focus word triggered this type of entity in the training corpus.
- About the context of the token in focus: lemmas ranging from the lemma at position -4 until the lemma at position +3 (relative to the focus word); part-of-speech ranging from position -1 until position +1; chunk ranging from position -1 until position +1 relative to the focus word; the chunk be-

[3] *I* stands for 'inside', *B* for 'beginning', and *O* for 'outside'.

fore the chunk to which the focus word belongs; a boolean indicating if a word is a protein or not for the words ranging from position -2 until position +3.

Class label	Precision	Recall	F-score
B-Gene_expression	59.32	60.23	59.77
B-Regulation	30.41	33.58	31.91
B-Entity	40.21	41.49	40.84
B-Positive_regulation	41.16	46.25	43.56
B-Binding	57.76	53.14	55.36
B-Negative_regulation	42.94	48.67	45.63
I-Negative_regulation	7.69	3.33	4.65
I-Positive_regulation	14.29	13.24	13.74
B-Phosphorylation	75.68	71.80	73.68
I-Regulation	14.29	10.00	11.77
B-Transcription	48.78	59.70	53.69
I-Entity	20.00	16.13	17.86
B-Localization	75.00	60.00	66.67
B-Protein_catabolism	73.08	100.00	84.44
O	97.66	97.62	97.64

Table 1: Results of the entity detection classifier. Entities that are not in the table have a precision and recall of 0.

Table 1 shows the results[4] of this first step. All class labels with a precision and recall of 0 are left out. The overall accuracy is 95.4%. This high accuracy is caused by the skewness of the data in the training corpus, which contains a higher proportion of instances with class label O. Instances with this class are correctly classified in the development test. B-Protein_catabolism and B-Phosphorylation get the highest scores. The reason why these classes get higher scores can be that the words that trigger these events are less diverse.

4.2 Phase 2: predicting the arguments and participants of events

In the second phase, another memory-based classifier predicts the participants and arguments of an event. Participants have the main role in the event and arguments are entities that further specify the event. In (1), for the event *phosphorylation* the system has to find that *STAT1, STAT3, STAT4, STAT5a,* and *STAT5b* are participants with the role *Theme* and that *tyrosine* is an argument with the role *Site*.

[4]In this section we provide results on development data because the gold test data have not been made available.

(1) IFN-alpha enhanced tyrosine phosphorylation of STAT1, STAT3, STAT4, STAT5a, and STAT5b.

We use the IB1 algorithm as implemented in TiMBL (version 6.1.2) (Daelemans et al., 2007). The classifier was parameterised by using gain ratio for feature weighting, overlap as distance metrics, 11 nearest neighbors for extrapolation, and normal majority voting for class voting weights.

For this classifier, instances represent combinations of an event with all the entities in a sentence, for as many events as there are in a sentence. Entities include entities and events. We use as input the output of the classifier in Phase 1, so only events and entities classified as such in Phase 1, and the gold proteins will be combined. Events can have participants and arguments in a sentence different that their sentence. We calculated that in the training corpus these cases account for 5.54% of the relations, and decided to restrict the combinations at the sentence level. For the sentence in (1) above, where *tyrosine, phosphorylation, STAT1, STAT3, STAT4, STAT5a*, and *STAT5b* are entities and of those only *phosphorylation* is an event, the instances would be produced by combining *phosphorylation* with the seven entities.

The features used by this classifier are the following:

- Of the event and of the combined entity: first word, last word, type, named entity provided by GDep, chain of lemmas, chain of part-of-speech (POS) tags, chain of chunk tags, dependency label of the first word, dependency label of the last word.
- Of the event context and of the combined entity context: word, lemma, POS, chunk, and GDep named entity of the five previous and next words.
- Of the context between event and combined entity: the chain of chunks in between, number of tokens in between, a binary feature indicating whether event is located before or after entity.
- Others: four features indicating the parental relation between the first and last words of the event and the first and last words of the entity. The values for this feature are: event_father, event_ancestor, entity_father, entity_ancestor, none. Five binary features indicating if the event accepts certain roles (Theme, Site, ToLoc, AtLoc, Cause).

Table 2 shows the results of this classifier per type of participant (Cause, Site, Theme) and type of argument (AtLoc, ToLoc). Arguments are very infrequent, and the participants are skewed towards the class Theme. Classes Site and Theme score high F1, and in both cases recall is higher than precision. The fact that the classifier overpredicts Sites and Themes will have a negative influence in the final scores of the full system. Further research will focus on improving precision.

Part/Arg	Total	Precision	Recall	F1
Cause	61	28.88	21.31	24.52
Site	20	54.83	85.00	66.66
Theme	683	55.50	72.32	62.80
AtLoc	1	25.00	100.00	40.00
ToLoc	4	75.00	75.00	75.00

Table 2: Results of finding the event participants and arguments.

Table 3 shows the results of finding the event participants and arguments per event type, expressed in terms of accuracy on the development corpus. Cause is easier to predict for Positive Regulation events, Site is the easiest class to predict, taking into account that AtLoc and ToLoc occur only 5 times in total, and Theme can be predicted successfully for Transcription and Gene Expression events, whereas it gets lower scores for Regulation, Binding, and Positive Regulation events.

Event Type	Arguments/Participants				
	Cause	Site	Theme	AtLoc	ToLoc
Binding	-	100.00	56.00	-	-
Gene Expr.	-	-	89.95	-	-
Localization	-	-	73.07	100.00	75.00
- Regulation	11.11	0.00	75.00	-	-
Phosphorylation	0.00	100.00	70.83	-	-
+ Regulation	27.77	90.90	56.77	-	-
Protein Catab.	-	-	60.00	-	-
Regulation	13.33	0.00	46.87	-	-
Transcription	-	-	94.44	-	-

Table 3: Results of finding the event participants and arguments per event type (accuracy).

Table 4 shows the results of finding the event participants that are Entity and Protein per type of event for events that are not regulations. Entity scores high in all cases, whereas Protein scores high for Transcription and Gene Expression events and low for Binding events.

Event Type	Arg./Part. Type	
	Entity	Protein
Binding	100.00	56.00
Gene Expr.	-	89.90
Localization	80.00	73.07
Phosphorylation	100.00	68.00
Protein Catab.	-	60.00
Transcription	-	94.44

Table 4: Results of finding the event participants and arguments that are Entity and Protein per event type (accuracy).

Table 5 shows the results of finding the participants and arguments of regulation events. In the case of regulation events, Entity is easier to classify with Positive Regulation events, and Protein with Negative Regulation events. In the cases in which events are participants of regulation events, Binding, Gene Expression and Phosphorylation are easier to classify with Positive Regulation events, Localization with Regulation events, Protein Catabolism with Negative Regulation events, and Transcription is easy to classify in all cases.

Arg./Part. Type	Event Type		
	Regulation	+ Regulation	-Regulation
Entity	0.00	90.90	0.00
Protein	17.85	38.88	45.45
Binding	-	75.00	66.66
Gene Expr.	66.66	90.47	75.00
Localization	100.00	80.00	75.00
Phosphorylation	0.00	44.44	0.00
Protein Catab.	0.00	40.00	100.00
Transcription	100.00	92.85	100.00

Table 5: Results of finding event arguments and participants for regulation events (accuracy).

From the results of the system in this phase we can extract some conclusions: data are skewed towards the Theme class; Themes are not equally predictable for the different types of events, they are better predictable for Gene Expression and Transcription; Proteins are more difficult to classify when they are Themes of regulation events; and Transcription and Localization events are easier to predict as Themes of regulation events, compared to the other types of events that are Themes of regulation events. This

suggests that it could be worth experimenting with a classifier per entity type and with a classifier per role, instead of using the same classifier for all types of entities.

4.3 Phase 3: heuristics to select the best frame per event

Phases 1 and 2 aimed at identifying events and candidates to event participants. However, the purpose of the task is to extract full frames of events. For a sentence like the one in (1) above, the system has to extract the event frames in (2).

(2)
1. Phosphorylation (phosphorylation): Theme (STAT1) Site (tyrosine)
2. Phosphorylation (phosphorylation): Theme (STAT3) Site (tyrosine)
3. Phosphorylation (phosphorylation): Theme (STAT5a) Site (tyrosine)
4. Phosphorylation (phosphorylation): Theme (STAT4) Site (tyrosine)
5. Phosphorylation (phosphorylation): Theme (STAT5b) Site (tyrosine)

It is necessary to apply heuristics in order to build the event frames from the output of the second classifier, which for the sentence in (1) above should contain the predictions in (3).

(3)
1. phosphorylation STAT1 : Theme
2. phosphorylation STAT3 : Theme
3. phosphorylation STAT5a : Theme
4. phosphorylation STAT4 : Theme
5. phosphorylation STAT5b : Theme
6. phosphorylation tyrosine : Site

Thus, in the third phase, postprocessing heuristics determine which is the frame of each event.

4.3.1 Specific heuristics for each type of event

The system contains different rules for each of the 5 types of participants (Cause, Site, Theme, AtLoc, ToLoc). The text entities are the entities defined during Phase 2. An event is created for every text entity for which the system predicted at least one participant or argument. To illustrate this we can take a look at the predictions for the Gene Expression event in (4) where the identifiers starting by T refer to entities in the text. The prediction would results in the events listed in (5).

(4) Gene_expression= Theme:T11=Theme:T12=Theme:T13

(5) E1 Gene_expression:T23 Theme:T11
E2 Gene_expression:T23 Theme:T12
E3 Gene_expression:T23 Theme:T13

Gene expression, Transcription, and Protein catabolism. These type of events have only a Theme. Therefore, an event frame is created for every Theme predicted for events that belong to these types.

Localization. A Localization event can have one Theme and 2 arguments: AtLoc and ToLoc. A Localization event with more than one predicted Theme will result in as many frames as predicted Themes. The arguments are passed on to every frame.

Binding. A Binding event can have multiple Themes and multiple Site arguments. If the system predicts more than one Theme for a Binding event, the heuristics first check if these Themes are in a coordination structure. Coordination checking consists of checking whether the word 'and' can be found between the Themes. Coordinated Themes will give rise to separate frames. Every participant and loose Theme is added to all created event lines. This case applies to the sentence in (6)

(6) When we analyzed the nature of STAT proteins capable of binding to IL-2Ralpha, pim-1, and IRF-1 GAS elements after cytokine stimulation, we observed IFN-alpha-induced binding of STAT1, STAT3, and STAT4, but not STAT5 to all of these elements.

The frames that should be created for this sentence listed in (7).

(7)
1. Binding (binding): Theme(STAT4) Theme2(IRF-1) Site2(GAS elements)
2. Binding (binding): Theme(STAT3) Theme2:(IL-2Ralpha) Site2(GAS elements)
3. Binding (binding): Theme(STAT3) Theme2(IRF-1) Site2(GAS elements)
4. Binding (binding): Theme(STAT4) Theme2(pim-1) Site2(GAS elements)
5. Binding (binding): Theme(STAT1) Theme2(IL-2Ralpha) Site2(GAS elements)

6. Binding (binding): Theme(STAT4) Theme2(IL-2Ralpha) Site2(GAS elements)
7. Binding (binding): Theme(IL-2Ralpha) Site(GAS elements)
8. Binding (binding): Theme(pim-1) Site(GAS elements)
9. Binding (binding): Theme(STAT1) Theme2(IRF-1) Site2(GAS elements)
10. Binding (binding): Theme(STAT3) Theme2(pim-1) Site2(GAS elements)
11. Binding (binding): Theme(IRF-1) Site(GAS elements)
12. Binding (binding): Theme(STAT1) Theme2(pim-1) Site2(GAS elements)

Phosphorylation. A Phosphorylation event can have one Theme and one Site. Multiple Themes for the same event will result in multiple frames. The Site argument will be added to every frame.

Regulation, Positive regulation, and Negative regulation. A Regulation event can have a Theme, a Cause, a Site, and a CSite. For Regulation events the system uses a different approach when creating new frames. It first checks which of the participants and arguments occurs the most frequent in a prediction and it creates as many separate frames as are needed to give every participant/argument its own frame. The remaining participants/arguments are added to the nearest frame. For this type of event a new frame can be created not only for multiple Themes but also for e.g. multiple Sites. The purpose of this strategy is to increase the recall of Regulation events.

4.3.2 Postprocessing

After translating predictions into frames some corrections are made.

1. Every Theme and Cause that is not a Protein is thrown away.
2. Every frame that has no Theme is provided with a default Theme. If no Protein is found before the focus word, the closest Protein after the word is taken as the default Theme.
3. Duplicates are removed.

5 Results

The official results of our system for Task 1 are presented in Table 6. The best F1 score are for Gene Expression and Protein Catabolism events. The lowest results are for all the types of regulation events and for Binding events. Binding events are more difficult to predict correctly because they can have more than one Theme.

	Total	Precision	Recall	F1
Binding	347	12.97	31.03	18.29
Gene Expr.	722	51.39	68.96	58.89
Localization	174	20.69	78.26	32.73
Phosphorylation	135	28.15	67.86	39.79
Protein Catab.	14	64.29	42.86	51.43
Transcription	137	24.82	41.46	31.05
Regulation	291	8.93	23.64	12.97
+Regulation	983	11.70	31.68	17.09
-Regulation	379	11.08	29.85	16.15
TOTAL	3182	22.50	47.70	30.58

Table 6: Official results of Task 1. Approximate Span Matching/Approximate Recursive Matching.

The official results of our system for Task 2 are presented in Table 7. Results are similar to the results of Task 1 because there are not many more arguments than participants. Recognising arguments was the additional goal of Task 2 in relation to Task 1.

	Total	Precision	Recall	F1
Binding	349	11.75	28.28	16.60
Gene Expr.	722	51.39	68.96	58.89
Localization	174	17.82	67.39	28.18
Phosphorylation	139	15.83	39.29	22.56
Protein Catab.	14	64.29	42.86	51.43
Transcription	137	24.82	41.46	31.05
Regulation	292	8.56	22.73	12.44
+Regulation	987	11.35	30.85	16.59
-Regulation	379	11.08	29.20	15.76
TOTAL	3193	21.52	45.77	29.27

Table 7: Official results of Task 2. Approximate Span Matching/Approximate Recursive Matching.

Results obtained on the development set are a little bit higher. For Task1 an overall F1 of 34.78 and for Task 2 33.54.

For most event types precision and recall are unbalanced, the system scores higher in recall. Further research should focus on increasing precision because the system is predicting false positives. It would be possible to add a step in order to filter out the false positives by comparing word sequences with event patterns derived from the corpus, which is an approach taken in the system by Sasaki et al. (2008) .

In the case of Binding events, both precision and recall are low. There are two explanations for this. In the first place, the first classifier misses almost half of the binding events. As an example, for the sentence in (8.1), the gold standard identifies as binding event the multiwords *binds as a homodimer* and *form heterodimers*, whereas the system identifies two binding events for the same sentence, *binds* and *homodimer*, none of which is correct because the correct one is the multiword unit. For the sentence in (8.2), the gold standard identifies as binding events *bind*, *form homo-*, and *heterodimers*, whereas the system identifies only *binds*.

(8) 1. The KBF1/p50 factor *binds as a homodimer* but can also *form heterodimers* with the products of other members of the same family, like the c-rel and v-rel (proto)oncogenes.

2. A mutant of KBF1/p50 (delta SP), unable to *bind* to DNA but able to *form homo-* or *heterodimers*, has been constructed.

From the sentence in (8.1) above the eight frames in (9) should be extracted, whereas the system extracts only the frames in (10), which are incorrect because the events have not been correctly identified.

(9)
1. Binding(binds as a homodimer) : Theme(KBF1)
2. Binding(binds as a homodimer) : Theme(p50)
3. Binding(form heterodimers) : Theme(KBF1) Theme2(c-rel)
4. Binding(form heterodimers) : Theme(p50) Theme2(v-rel)
5. Binding(form heterodimers) : Theme(p50) Theme2(c-rel)
6. Binding(form heterodimers) : Theme(KBF1) Theme2(v-rel)
7. Binding(bind) : Theme(p50)
8. Binding(bind) : Theme(KBF1)

(10)
1. Binding(binds) : Theme(v-rel)
2. Binding(homodimer) : Theme(c-rel)

The complexity of frame extraction of Binding events contrasts with the less complex extraction of frames for Gene Expression events, like the one in sentence (11), where *expression* has been identified correctly by the system as an event and the frame in (12) has been correctly extracted.

(11) Thus, c-Fos/c-Jun heterodimers might contribute to the repression of DRA gene *expression*.

(12) Gene Expression(expression) : Theme(DRA)

6 Conclusions

In this paper we presented a supervised machine learning system that extracts event frames from biomedical texts in three phases. The system participated in the BioNLP Shared Task 2009, achieving an F-score of 30.58 in Task 1, and 29.27 in Task 2. The frame extraction task was modeled applying the same approach that has been applied to tasks like semantic role labeling or negation scope detection, in order to check whether such an approach would be suitable for a frame extraction task. The results obtained for the present task do not compare to results obtained in the mentioned tasks, where state of the art F-scores are above 80.

Extracting biomedical event frames is more complex than labeling semantic roles because of several reasons. Semantic roles are mostly assigned to syntactic constituents, predicates have only one frame and all the arguments belong to the same frame. In contrast, in the biomedical domain one event can have several frames, each frame having different participants, the boundaries of which do not coincide with syntactic constituents.

The system presented here can be improved in several directions. Future research will concentrate on increasing precision in general, and precision and recall of binding events in particular. Analysing in depth the errors made by the system at each phase will allow us to find the weaker aspects of the system. From the results of the system in the second phase we could draw some conclusions: data are skewed towards the Theme class; Themes are not equally predictable for the different types of events; Proteins are more difficult to classify when they are Themes of regulation events; and Transcription and Localization events are easier to predict as Themes of regulation events, compared to the other types of events that are Themes of regulation events. We plan to experiment with a classifier per entity type and with a classifier per role, instead of using the same classifier for all types of entities. Additionally, the effects of the postprocessing rules in Phase 3 will be evaluated.

Acknowledgments

Our work was made possible through financial support from the University of Antwerp (GOA project BIOGRAPH). We are grateful to two anonymous reviewers for their valuable comments.

References

S. Ananiadou and J. McNaught. 2006. *Text Mining for Biology and Biomedicine*. Artech House Books, London.

S. Buchholz and E. Marsi. 2006. CoNLL-X shared task on multilingual dependency parsing. In *Proc. of the X CoNLL Shared Task*, New York. SIGNLL.

M. Bundschus, M. Dejori, M. Stetter, V. Tresp, and H-P Kriegel. 2008. Extraction of semantic biomedical relations from text using conditional random fields. *BMC Bioinformatics*, 9.

W.C. Chou, R.T.H. Tsai, Y-S. Su, W. Ku, T-Y Sung, and W-L Hsu. 2006. A semi-automatic method for annotating a biomedical proposition bank. In *Proc. of ACL Workshop on Frontiers in Linguistically Annotated Corpora 2006*, pages 5–12.

N. Collier, H.S. Park, N. Ogata, Y. Tateisi, C. Nobata, T. Sekimizu, H. Imai, and J. Tsujii. 1999. The GENIA project: corpus-based knowledge acquisition and information extraction from genome research papers. In *Proc. of EACL 1999*.

T. M. Cover and P. E. Hart. 1967. Nearest neighbor pattern classification. *Institute of Electrical and Electronics Engineers Transactions on Information Theory*, 13:21–27.

W. Daelemans and A. van den Bosch. 2005. *Memory-based language processing*. Cambridge University Press, Cambridge, UK.

W. Daelemans, A. Van den Bosch, and J. Zavrel. 1999. Forgetting exceptions is harmful in language learning. *Machine Learning, Special issue on Natural Language Learning*, 34:11–41.

W. Daelemans, J. Zavrel, K. Van der Sloot, and A. Van den Bosch. 2007. TiMBL: Tilburg memory based learner, version 6.1, reference guide. Technical Report Series 07-07, ILK, Tilburg, The Netherlands.

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

M. Krallinger and A. Valencia. 2005. Text-mining and information-retrieval services for molecular biology. *Genome Biology*, 6:224.

M. Krallinger, F. Leitner, C. Rodriguez-Penagos, and A. Valencia. 2008a. Overview of the protein–protein interaction annotation extraction task of BioCreative II. *Genome Biology*, 9(Suppl 2):S4.

M. Krallinger, A. Valencia, and L. Hirschman. 2008b. Linking genes to literature: text mining, information extraction, and retrieval applications for biology. *Genome Biology*, 9(Suppl 2):S8.

R. Morante and W. Daelemans. 2009. A metalearning approach to processing the scope of negation. In *Proceedings of CoNLL 2009*, Boulder, Colorado.

R. Morante, W. Daelemans, and V. Van Asch. 2008. A combined memory-based semantic role labeler of English. In *Proc. of the CoNLL 2008*, pages 208–212, Manchester, UK.

C. Nédellec. 2005. Learning language in logic – genic interaction extraction challenge. In *Proc. of Learning Language in Logic Workshop 2005*, pages 31–37, Bonn.

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).

K. Sagae and J. Tsujii. 2007. Dependency parsing and domain adaptation with lr models and parser ensembles. In *Proc. of CoNLL 2007 Shared Task, EMNLP-CoNLL*, pages 82–94, Prague. ACL.

Y. Sasaki, P. Thompson, P. Cotter, J. McNaught, and S. Ananiadou. 2008. Event frame extraction based on a gene regulation corpus. In *Proc. of Coling 2008*, pages 761–768.

T. Wattarujeekrit, P.K. Shah, and N. Collier. 2004. PASBio: predicate-argument structures for event extraction in molecular biology. *BMC Bioinformatics*, 5:155.

A. Yakushiji, Y. Tateisi, Y. Miyao, and J. Tsujii. 2001. Event extraction from biomedical papers using a full parser. In *Pac Symp Biocomput*.