High-precision biological event extraction with a concept recognizer

K. Bretonnel Cohen*, Karin Verspoor*, Helen L. Johnson, Chris Roeder,
Philip V. Ogren, William A. Baumgartner Jr., Elizabeth White, Hannah Tipney, and Lawrence Hunter
Center for Computational Pharmacology
University of Colorado Denver School of Medicine
PO Box 6511, MS 8303, Aurora, CO 80045 USA
kevin.cohen@gmail.com, karin.verspoor@ucdenver.edu, helen.linguist@gmail.com,
chris.roeder@ucdenver.edu, philip@ogren.info, william.baumgartner@ucdenver.edu,
elizabeth.white@colorado.edu, hannah.tipney@ucdenver.edu, larry.hunter@ucdenver.edu

Abstract

We approached the problems of event detection, argument identification, and negation and speculation detection as one of concept recognition and analysis. Our methodology involved using the OpenDMAP semantic parser with manually-written rules. We achieved state-of-the-art precision for two of the three tasks, scoring the highest of 24 teams at precision of 71.81 on Task 1 and the highest of 6 teams at precision of 70.97 on Task 2.

The OpenDMAP system and the rule set are available at bionlp.sourceforge.net.

*These two authors contributed equally to the paper.

1 Introduction

We approached the problem of biomedical event recognition as one of concept recognition and analysis. Concept analysis is the process of taking a textual input and building from it an abstract representation of the concepts that are reflected in it. Concept recognition can be equivalent to the named entity recognition task when it is limited to locating mentions of particular semantic types in text, or it can be more abstract when it is focused on recognizing predicative relationships, e.g. events and their participants.

2 BioNLP'09 Shared Task

Our system was entered into all three of the BioNLP'09 (Kim et al., 2009) shared tasks:

- **Event detection and characterization** This task requires recognition of 9 basic biological events: gene expression, transcription, protein catabolism, protein localization, binding, phosphorylation, regulation, positive regulation and negative regulation. It requires identification of the core THEME and/or CAUSE participants in the event, i.e. the protein(s) being produced, broken down, bound, regulated, etc.

- **Event argument recognition** This task builds on the previous task, adding in additional arguments of the events, such as the site (protein or DNA region) of a binding event, or the location of a protein in a localization event.

- **Recognition of negations and speculations** This task requires identification of negations of events (e.g. event X did *not* occur), and speculation about events (e.g. We *claim* that event X *should* occur).

3 Our approach

We used the OpenDMAP system developed at the University of Colorado School of Medicine (Hunter et al., 2008) for our submission to the BioNLP '09 Shared Task on Event Extraction. OpenDMAP is an ontology-driven, integrated concept analysis system that supports information extraction from text through the use of patterns represented in a classic form of "semantic grammar," freely mixing text literals, semantically typed basal syntactic constituents, and semantically defined classes of entities. Our approach is to take advantage of the high

Proceedings of the Workshop on BioNLP: Shared Task, pages 50–58,
Boulder, Colorado, June 2009.

quality ontologies available in the biomedical domain to formally define entities, events, and constraints on slots within events and to develop patterns for how concepts can be expressed in text that take advantage of both semantic and linguistic characteristics of the text. We manually built patterns for each event type by examining the training data and by using native speaker intuitions about likely ways of expressing relationships, similar to the technique described in (Cohen et al., 2004). The patterns characterize the linguistic expression of that event and identify the arguments (participants) of the events according to (a) occurrence in a relevant linguistic context and (b) satisfaction of appropriate semantic constraints, as defined by our ontology. Our solution results in very high precision information extraction, although the current rule set has limited recall.

3.1 The reference ontology

The central organizing structure of an OpenDMAP project is an ontology. We built the ontology for this project by combining elements of several community-consensus ontologies—the Gene Ontology (GO), Cell Type Ontology (CTO), BRENDA Tissue Ontology (BTO), Foundational Model of Anatomy (FMA), Cell Cycle Ontology (CCO), and Sequence Ontology (SO)—and a small number of additional concepts to represent task-specific aspects of the system, such as event trigger words. Combining the ontologies was done with the Prompt plug-in for Protégé.

The ontology included concepts representing each event type. These were represented as frames, with slots for the various things that needed to be returned by the system—the trigger word and the various slot fillers. All slot fillers were constrained to be concepts in some community-consensus ontology. The core event arguments were constrained in the ontology to be of type *protein* from the Sequence Ontology (except in the case of regulation events, where biological events themselves could satisfy the THEME role), while the type of the other event arguments varied. For instance, the ATLOC argument of a gene expression event was constrained to be one of tissue (from BTO), cell type (from CTO), or cellular component (from GO-Cellular Component), while the BINDING argument of a binding event was constrained to be one of binding_site, DNA, domain, or chromosome (all from the SO and all tagged by LingPipe). Table 1 lists the various types.

3.2 Named entity recognition

For proteins, we used the gold standard annotations provided by the organizers. For other semantic classes, we constructed a compound named entity recognition system which consists of a LingPipe GENIA tagging module (LingPipe, (Alias-i, 2008)), and several dictionary look-up modules. The dictionary lookup was done using a component from the UIMA (IBM, 2009; Ferrucci and Lally, 2004) sandbox called the ConceptMapper.

We loaded the ConceptMapper with dictionaries derived from several ontologies, including the Gene Ontology Cellular Component branch, Cell Type Ontology, BRENDA Tissue Ontology, and the Sequence Ontology. The dictionaries contained the names and name variants for each concept in each ontology, and matches in the input documents were annotated with the relevant concept ID for the match. The only modifications that we made to these community-consensus ontologies were to remove the single concept *cell* from the Cell Type Ontology and to add the synonym *nuclear* to the Gene Ontology Cell Component concept *nucleus*.

The protein annotations were used to constrain the text entities that could satisfy the THEME role in the events of interest. The other named entities were added for the identification of non-core event participants for Task 2.

3.3 Pattern development strategies

3.3.1 Corpus analysis

Using a tool that we developed for visualizing the training data (described below), a subset of the gold-standard annotations were grouped by event type and by trigger word type (nominalization, passive verb, active verb, or multiword phrase). This organization helped to suggest the argument structures of the event predicates and also highlighted the variation within argument structures. It also showed the nature of more extensive intervening text that would need to be handled for the patterns to achieve higher recall.

Based on this corpus analysis, patterns were developed manually using an iterative process in which individual patterns or groups of patterns were tested

Table 1: Semantic restrictions on Task 2 event arguments. CCO = Cell Cycle Ontology, FMA = Foundational Model of Anatomy, other ontologies identified in the text.

Event Type	Site	AtLoc	ToLoc
binding	protein domain (SO), binding site (SO), DNA (SO), chromosome (SO)		
gene expression	gene (SO), biological entity (CCO)	tissue (BTO), cell type (CTO), cellular component (GO)	
localization		cellular component (GO)	cellular component (GO)
phosphorylation	amino acid (FMA), polypeptide region (SO)		
protein catabolism	cellular component (GO)		
transcription	gene (SO), biological entity (CCO)		

on the training data to determine their impact on performance. Pattern writers started with the most frequent trigger words and argument structures.

3.3.2 Trigger words

In the training data, we were provided annotations of all relevant event types occurring in the training documents. These annotations included a *trigger word* specifying the specific word in the input text which indicated the occurrence of each event. We utilized the trigger words in the training set as anchors for our linguistic patterns. We built patterns around the generic concept of, e.g. an *expression trigger word* and then varied the actual strings that were allowed to satisfy that concept. We then ran experiments with our patterns and these varying sets of trigger words for each event type, discarding those that degraded system performance when evaluated with respect to the gold standard annotations.

Most often a trigger word was removed from an event type trigger list because it was also a trigger word for another event type and therefore reduced performance by increasing the false positive rate. For example, the trigger words “level” and “levels” appear in the training data trigger word lists of gene expression, transcription, and all three regulation event types.

The selection of trigger words was guided by a frequency analysis of the trigger words provided in the task training data. In a post-hoc analysis, we find that a different proportion of the set of trigger words was finally chosen for each different event type. Between 10-20% of the top frequency-ranked trigger words were used for simple event types, with the exception that phosphorylation trigger words were chosen from the top 30%. For instance, for gene expression all of the top 15 most frequent trigger words were used (corresponding to the top 16%). For complex event types (the regulations) better performance was achieved by limiting the list to between 5-10% of the most frequent trigger words.

In addition, variants of frequent trigger words were included. For instance, the nominalization “expression” is the most frequent gene expression trigger word and the verbal inflections “expressed” and “express” are also in the top 20%. The verbal inflection “expresses” is ranked lower than the top 30%, but was nonetheless included as a trigger word in the gene expression patterns.

3.3.3 Patterns

As in our previous publications on OpenDMAP, we refer to our semantic rules as *patterns*. For this task, each pattern has at a minimum an event argument THEME and an event-specific trigger word. For example, $\{phosphorylation\}$:=

$[phosphorylation\_nominalization][Theme]$, where $[phosphorylization\_nominalization]$ represents a trigger word. Both elements are defined semantically. Event THEMEs are constrained by restrictions placed on them in the ontology, as described above.

The methodology for creating complex event patterns such as regulation was the same as for simple events, with the exception that the THEMEs were defined in the ontology to also include biological processes. Iterative pattern writing and testing was a little more arduous because these patterns relied on the success of the simple event patterns, and hence more in-depth analysis was required to perform performance-increasing pattern adjustments. For further details on the pattern language, the reader is referred to (Hunter et al., 2008).

3.3.4 Nominalizations

Nominalizations were very frequent in the training data; for seven out of nine event types, the most common trigger word was a nominalization. In writing our grammars, we focused on these nominalizations. To write grammars for nominalizations, we capitalized on some of the insights from (Cohen et al., 2008). Non-ellided (or otherwise absent) arguments of nominalizations can occur in three basic positions:

- Within the noun phrase, after the nominalization, typically in a prepositional phrase
- Within the noun phrase, immediately preceding the nominalization
- External to the noun phrase

The first of these is the most straightforward to handle in a rule-based approach. This is particularly true in the case of a task definition like that of BioNLP '09, which focused on themes, since an examination of the training data showed that when themes were post-nominal in a prepositional phrase, then that phrase was most commonly headed by *of*.

The second of these is somewhat more challenging. This is because both agents and themes can occur immediately before the nominalization, e.g. *phenobarbital induction* (induction *by* phenobarbital) and *trkA expression* (expression *of* trkA). To decide how to handle pre-nominal arguments, we made use of the data on semantic roles and syntactic position found in (Cohen et al., 2008). That study found that themes outnumbered agents in the prenominal position by a ratio of 2.5 to 1. Based on this observation, we assigned pre-nominal arguments to the theme role.

Noun-phrase-external arguments are the most challenging, both for automatic processing and for human interpreters; one of the major problems is to differentiate between situations where they are present but outside of the noun phrase, and situations where they are entirely absent. Since the current implementation of OpenDMAP does not have robust access to syntactic structure, our only recourse for handling these arguments was through wildcards, and since they mostly decreased precision without a corresponding increase in recall, we did not attempt to capture them.

3.3.5 Negation and speculation

Corpus analysis of the training set revealed two broad categories each for negation and speculation modifications, all of which can be described in terms of the scope of modification.

Negation

Broadly speaking, an event itself can be negated or some aspect of an event can be negated. In other words, the scope of a negation modification can be over the existence of an event (first example below), or over an argument of an existing event (second example).

- *This failure to degrade IkappaBalpha ...* (PMID 10087185)
- *AP-1 but not NF-IL-6 DNA binding activity ...* (PMID 10233875)

Patterns were written to handle both types of negation. The negation phrases "but not" and "but neither" were appended to event patterns to catch those events that were negated as a result of a negated argument. For event negation, a more extensive list of trigger words was used that included verbal phrases such as "failure to" and "absence of."

The search for negated events was conducted in two passes. Events for which negation cues fall outside the span of text that stretches from argument to event trigger word were handled concurrently with the search for events. A second search was conducted on extracted events for negation cues that fell within the argument to event trigger word span, such as

. . . *IL-2 does not induce I kappa B alpha degradation* (PMID 10092783)

This second pass allowed us to capture one additional negation (6 rather than 5) on the test data.

Speculation

The two types of speculation in the training data can be described by the distinction between "de re" and "de dicto" assertions. The "de dicto" assertions of speculation in the training data are modifications that call into question the degree of known truth of an event, as in

. . . *CTLA-4 ligation did not appear to affect the CD28 - mediated stabilization* (PMID 10029815)

The "de re" speculation address the potential existence of an event rather that its degree of truth. In these cases, the event is often being introduced in text by a statement of intention to study the event, as in

. . . *we investigated CTCF expression . . . [10037138]*

To address these distinct types of speculation, two sets of trigger words were developed. One set consisted largely of verbs denoting research activities, e.g. *research, study, examine investigate,* etc. The other set consisted of verbs and adverbs that denote uncertainty, and included trigger words such as *suggests, unknown,* and *seems*.

3.4 Handling of coordination

Coordination was handled using the OpenNLP constituent parser along with the UIMA wrappers that they provide via their code repository. We chose OpenNLP because it is easy to train a model, it integrates easily into a UIMA pipeline, and because of competitive parsing results as reported by Buyko (Buyko et al., 2006). The parser was trained using 500 abstracts from the beta version of the GENIA treebank and 10 full-text articles from the CRAFT corpus (Verspoor et al., In press). From the constituent parse we extracted coordination structures into a simplified data structure that captures each conjunction along with its conjuncts. These were provided to downstream components. The coordination component achieves an F-score of 74.6% at the token level and an F-score of 57.5% at the conjunct level when evaluated against GENIA. For both measures the recall was higher than the precision by 4% and 8%, respectively.

We utilized the coordination analysis to identify events in which the THEME argument was expressed as a conjoined noun phrase. These were assumed to have a distributed reading and were post-processed to create an individual event involving each conjunct, and further filtered to only include given (A1) protein references. So, for instance, analysis of the sentence in the example below should result in the detection of three separate gene expression events, involving the proteins HLA-DR, CD86, and CD40, respectively.

> NAC was shown to down-regulate the production of cytokines by DC as well as **their surface expression of HLA-DR, CD86 (B7-2), and CD40 molecules** . . . (PMID 10072497)

3.5 Software infrastructure

We took advantage of our existing infrastructure based on UIMA (The Unstructured Information Management Architecture) (IBM, 2009; Ferrucci and Lally, 2004) to support text processing and data analysis.

3.5.1 Development tools

We developed a visualization tool to enable the linguistic pattern writers to better analyze the training data. This tool shows the source text one sentence at a time with the annotated words highlighted. A list following each sentence shows details of the annotations.

3.6 Errors in the training data

In some cases, there were discrepancies between the training data and the official problem definitions. This was a source of problems in the pattern development phase. For example, phosphorylation events are defined in the task definition as having only a THEME and a SITE. However, there were instances in the training data that included both a THEME and a CAUSE argument. When those events were identified by our system and the CAUSE was labelled, they were rejected during a syntactic error check by the test server.

4 Results

4.1 Official Results

We are listed as Team 13. Table 2 shows our results on the official metrics. Our precision was the highest achieved by any group for Task 1 and Task 2, at 71.81 for Task 1 and 70.97 for task 2. Our recalls were much lower and adversely impacted our F-measure; ranked by F-measure, we ranked 19th out of 24 groups.

We noted that our results for the exact match metric and for the approximate match metric were very close, suggesting that our techniques for named entity recognition and for recognizing trigger words are doing a good job of capturing the appropriate spans.

4.2 Other analysis: Bug fixes and coordination handling

In addition to our official results, we also report in Table 3 (see last page) the results of a run in which we fixed a number of bugs. This represents our current best estimate of our performance. The precision drops from 71.81 for Task 1 to 67.19, and from 70.97 for Task 2 to 65.74, but these precisions are still well above the second-highest precisions of 62.21 for Task 1 and 56.87 for Task 2. As the table shows, we had corresponding small increases in our recall to 17.38 and in our F-measure to 27.62 for Task 1, and in our recall to 17.07 and F-measure to 27.10 for Task 2.

We evaluated the effects of coordination handling by doing separate runs with and without this element of the processing pipeline. Compared to our unofficial results, which had an overall F-measure for Task 1 of 27.62 and for Task 2 of 27.10, a version of the system without handling of coordination had an overall F-measure for Task 1 of 24.72 and for Task 2 of 24.21.

4.3 Error Analysis

4.3.1 False negatives

To better understand the causes of our low recall, we performed a detailed error analysis of false negatives using the devtest data. (Note that this section includes a very small number of examples from the devtest data.) We found five major causes of false negatives:

- Intervening material between trigger words and arguments
- Coordination that was not handled by our coordination component
- Low coverage of trigger words
- Anaphora and coreference
- Appositive gene names and symbols

Intervening material For reasons that we detail in the *Discussion* section, we avoided the use of wildcards. This, and the lack of syntactic analysis in the version of the system that we used (note that syntactic analyses *can* be incorporated into an OpenDMAP workflow), meant that if there was text intervening between a trigger word and an argument, e.g. in *to efficiently [express] in developing thymocytes a mutant form of the [NF-kappa B inhibitor]* (PMID 10092801), where the bracketed text is the trigger word and the argument, our pattern would not match.

Unhandled coordination Our coordination system only handled coordinated protein names. Thus, in cases where other important elements of the utterance, such as the trigger word *transcription* in *transcription and subsequent synthesis and secretion of galectin-3* (PMID 8623933) were in coordinated structures, we missed the relevant event arguments.

Low coverage of trigger words As we discuss in the *Methods* section, we did not attempt to cover all trigger words, in part because some less-frequent trigger words were involved in multiple event types, in part because some of them were extremely low-frequency and we did not want to overfit to the training data, and in part due to the time constraints of the shared task.

Anaphora and coreference Recognition of some events in the data would require the ability to do anaphora and coreference resolution. For example, in *Although 2 early lytic transcripts, [BZLF1] and [BHRF1], were also detected in 13 and 10 cases, respectively, the lack of ZEBRA staining in any case indicates that these lytic transcripts are most likely*

	Tasks 1 and 3					Task 2		
Event class	GS	answer	R	P	F	R	P	F
Localization	174 (18)	18 (18)	10.34	100.00	18.75	9.77	94.44	17.71
Binding	347 (44)	110 (44)	12.68	40.00	19.26	12.32	39.09	18.74
Gene expression	722 (263)	306 (263)	36.43	85.95	51.17	36.43	85.95	51.17
Transcription	137 (18)	20 (18)	13.14	90.00	22.93	13.14	90.00	22.93
Protein catabolism	14 (4)	6 (4)	28.57	66.67	40.00	28.57	66.67	40.00
Phosphorylation	135 (30)	30 (30)	22.22	100.00	36.36	20.14	93.33	33.14
EVENT TOTAL	1529 (377)	490 (377)	24.66	76.94	37.35	24.30	76.12	36.84
Regulation	291 (9)	19 (9)	3.09	47.37	5.81	3.08	47.37	5.79
Positive regulation	983 (32)	65 (32)	3.26	49.23	6.11	3.24	49.23	6.08
Negative regulation	379 (10)	22 (10)	2.64	45.45	4.99	2.37	40.91	4.49
REGULATION TOTAL	1653 (51)	106 (51)	3.09	48.11	5.80	3.02	47.17	5.67
Negation	227 (4)	76 (4)	1.76	5.26	2.64			
Speculation	208 (14)	105 (14)	6.73	13.33	8.95			
MODIFICATION TOTAL	435 (18)	181 (18)	4.14	9.94	5.84			
ALL TOTAL	3182 (428)	596 (428)	13.45	71.81	22.66	13.25	70.97	22.33

Table 2: Official scores for Tasks 1 and 2, and modification scores only for Task 3, from the approximate span matching/approximate recursive matching table. GS = gold standard (true positives) (given for Tasks 1/3 only), answer = all responses (true positives) (given for tasks 1/3 only), R = recall, P = precision, F = F-measure. All results are as calculated by the official scoring application.

[expressed] by rare cells in the biopsies entering lytic cycle (PMID 8903467), where the bracketed text is the arguments and the trigger word, the syntactic object of the verb is the anaphoric noun phrase *these lytic transcripts*, so even with the addition of a syntactic component to our system, we still would not have recognized the appropriate arguments without the ability to do anaphora resolution.

Appositives The annotation guidelines for proteins apparently specified that when a gene name was present in an appositive with its symbol, the symbol was selected as the gold-standard argument. For this reason, in examples like *[expression] of Fas ligand [FasL]* (PMID 10092076), where the bracketed text is the trigger word and the argument, the gene name constituted intervening material from the perspective of our patterns, which therefore did not match.

We return to a discussion of recall and its implications for systems like ours in the *Discussion* section.

4.3.2 False positives

Although our overall rate of false positives was low, we sampled 45 false positive events distributed across the nine event types and reviewed them with a biologist.

We noted two main causes of error. The most common was that we misidentified a slot filler or were missing a slot filler completely for an actual event. The other main reason for false positives was when we erroneously identified a (non)event. For example, in *coexpression of NF-kappa B/Rel and Sp1 transcription factors* (PMID 7479915), we mistakenly identified *Sp1 transcription* as an event.

5 Discussion

Our results demonstrate that it is possible to achieve state-of-the art precision over a broad range of tasks and event types using our approach of manually constructed, ontologically typed rules—our precision of 71.81 on Task 1 was ten points higher than the second-highest precision (62.21), and our precision of 70.97 on Task 2 was 14 points higher than the second-highest precision (56.87). It remains the case that our recall was low enough to drop our F-measure considerably. Will it be the case that a system like ours can scale to practical performance levels nonetheless? Four factors suggest that it can.

The first is that there is considerable redundancy in the data; although we have not quantified it for this data set, we note that the same event is often

	Tasks 1 and 3					Task 2		
Event class	GS	answer	R	P	F	R	P	F
Localization	174 (33)	41 (33)	18.97	80.49	30.70	16.67	69.05	26.85
Binding	347 (62)	152 (62)	17.87	40.79	24.85	17.48	40.13	24.35
Gene expression	722 (290)	344 (290)	40.17	84.30	54.41	40.17	84.30	54.41
Transcription	137 (28)	31 (28)	20.44	90.32	33.33	20.44	90.32	33.33
Protein catabolism	14 (4)	6 (4)	28.57	66.67	40.00	28.57	66.67	40.00
Phosphorylation	135 (47)	48 (47)	34.81	97.92	51.37	32.37	84.91	46.88
EVENT TOTAL	1529 (464)	622 (464)	30.35	74.60	43.14	29.77	72.77	42.26
Regulation	291 (11)	31 (11)	3.78	35.48	6.83	3.77	35.48	6.81
Positive regulation	983 (60)	129 (60)	6.10	46.51	10.79	6.08	46.51	10.75
Negative regulation	379 (18)	41 (18)	4.75	43.90	8.57	4.49	41.46	8.10
REGULATION TOTAL	1653 (89)	201 (89)	5.38	44.28	9.60	5.31	43.78	9.47
Negation	227 (6)	129 (6)	2.64	4.65	3.37			
Speculation	208 (25)	165 (25)	12.02	15.15	13.40			
MODIFICATION TOTAL	435 (31)	294 (31)	7.13	10.54	8.50			
ALL TOTAL	3182 (553)	823 (553)	17.38	67.19	27.62	17.07	65.74	27.10

Table 3: Updated results on test data for Tasks 1-3, with important bug fixes in the code base. See key above.

mentioned repeatedly, but for knowledge base building and other uses of the extracted information, it is only strictly necessary to recognize an event once (although multiple recognition of the same assertion may increase our confidence in its correctness).

The second is that there is often redundancy across the literature; the best-supported assertions will be reported as initial findings and then repeated as background information.

The third is that these recall results reflect an approach that made no use of syntactic analysis beyond handling coordination. There is often text present in the input that cannot be disregarded without either using wildcards, which generally decreased precision in our experiments and which we generally eschewed, or making use of syntactic information to isolate phrasal heads. Syntactic analysis, particularly when combined with analysis of predicate-argument structure, has recently been shown to be an effective tool in biomedical information extraction (Miyao et al., 2009). There is broad need for this—for example, of the thirty localization events in the training data whose trigger word was *translocation*, a full eighteen had intervening textual material that made it impossible for simple patterns like $translocationof[Theme]$ or $[ToLoc]translocation$ to match.

Finally, our recall numbers reflect a very short development cycle, with as few as four patterns written for many event types. A less time-constrained pattern-writing effort would almost certainly result in increased recall.

Acknowledgments

We gratefully acknowledge Mike Bada's help in loading the Sequence Ontology into Protégé.

This work was supported by NIH grants R01LM009254, R01GM083649, and R01LM008111 to Lawrence Hunter and T15LM009451 to Philip Ogren.

References

Alias-i. 2008. LingPipe 3.1.2.

Ekaterina Buyko, Joachim Wermter, Michael Poprat, and Udo Hahn. 2006. Automatically mapping an NLP core engine to the biology domain. In *Proceedings of the ISMB 2006 joint BioLINK/Bio-Ontologies meeting*.

K. B. Cohen, L. Tanabe, S. Kinoshita, and L. Hunter. 2004. A resource for constructing customized test suites for molecular biology entity identification systems. *BioLINK 2004*, pages 1–8.

K. Bretonnel Cohen, Martha Palmer, and Lawrence Hunter. 2008. Nominalization and alternations in biomedical language. *PLoS ONE*, 3(9).

D. Ferrucci and A. Lally. 2004. Building an example application with the unstructured information management architecture. *IBM Systems Journal*, 43(3):455–475, July.

Lawrence Hunter, Zhiyong Lu, James Firby, William A. Baumgartner Jr., Helen L. Johnson, Philip V. Ogren, and K. Bretonnel Cohen. 2008. OpenDMAP: An open-source, ontology-driven concept analysis engine, with applications to capturing knowledge regarding protein transport, protein interactions and cell-specific gene expression. *BMC Bioinformatics*, 9(78).

IBM. 2009. UIMA Java framework. http://uima-framework.sourceforge.net/.

Jin-Dong Kim, Tomoko Ohta, Sampo Pyysalo, Yoshinobu Kano, and Jun'ichi Tsujii. 2009. Overview of BioNLP'09 shared task on event extraction. In *Proceedings of Natural Language Processing in Biomedicine (BioNLP) NAACL 2009 Workshop*. To appear.

Yusuke Miyao, Kenji Sagae, Rune Saetre, Takuya Matsuzaki, and Jun'ichi Tsujii. 2009. Evaluating contributions of natural language parsers to protein-protein interaction extraction. *Bioinformatics*, 25(3):394–400.

Karin Verspoor, K. Bretonnel Cohen, and Lawrence Hunter. In press. The textual characteristics of traditional and Open Access scientific journals are similar. *BMC Bioinformatics*.