Improving Text Mining with Controlled Natural Language: A Case Study for Protein Interactions

Tobias Kuhn[1,2], Loïc Royer[1], Norbert E. Fuchs[2], and Michael Schröder[1]

[1] Biotechnological Center, TU Dresden, Germany
{loic.royer, michael.schroeder}@biotec.tu-dresden.de,
http://www.biotec.tu-dresden.de/schroeder
[2] Department of Informatics, University of Zurich, Switzerland
{tkuhn, fuchs}@ifi.unizh.ch,
http://www.ifi.unizh.ch/attempto

Abstract. Linking the biomedical literature to other data resources is notoriously difficult and requires text mining. Text mining aims to automatically extract facts from literature. Since authors write in natural language, text mining is a great natural language processing challenge, which is far from being solved. We propose an alternative: If authors and editors summarize the main facts in a controlled natural language, text mining will become easier and more powerful. To demonstrate this approach, we use the language Attempto Controlled English (ACE). We define a simple model to capture the main aspects of protein interactions. To evaluate our approach, we collected a dataset of 459 paragraph headings about protein interaction from literature. 56% of these headings can be represented exactly in ACE and another 23% partially. These results indicate that our approach is feasible.

1 Introduction

In this paper we introduce a new paradigm of how to make knowledge of scientific papers accessible by computers. We focus on the fields of life sciences – particular biology – but our approach could be used in other fields as well.

Our approach consists of letting authors express their scientific results in a formal summary that could be an integral part of the papers they publish. We argue that it is more reasonable to let the authors formalize their own results, instead of trying to extract these results from the articles.

This section explains our motivation, introduces the language Attempto Controlled English (ACE) and compares it with other knowledge representation languages. Section 2 shows how ACE is used to build an ontology for protein interactions. In Sect. 3 we use this ontology as foundation for the expression of scientific results and we show how 89 selected articles could have been summarized in ACE. Section 4 shows the benefits of our approach and Sect. 5, finally, gives a short outlook.

U. Leser, F. Naumann, and B. Eckman (Eds.): DILS 2006, LNBI 4075, pp. 66–81, 2006.

1.1 Motivation

Biomedical scientists are challenged by an ever-increasing amount of scientific papers. The indexing service *PubMed*[1] shows the huge quantity of literature that the scientists have to face. It contains at the moment 16 million articles and grows every year by over 600'000 articles. All these biomedical articles are written in natural language. That means that we cannot easily process them with computers. But, facing the quantity of literature, it is clear that we need computational support in order to manage the contained knowledge.

In the last years, *text mining* and *information extraction* – which build both upon natural language processing (NLP) – gained an increasing interest in biomedical sciences. They aim to extract some kind of formal knowledge from natural language texts, which is generally considered a very demanding task. Even the basic problem of *named entity recognition*, that aims to identify named entities (e.g. protein names) in natural texts, is far from being solved. Other major aspects of text mining are the extraction of relationships (e.g. protein interactions), the automatic classification of texts, and the generation of new hypotheses on the basis of the available literature [3]. The *BioCreAtIvE* contest [21] nicely shows, that even sophisticated tools for text mining have a considerable lack of precision and recall: For a simple "named entity recognition"-task the precision ranged up to 86% and the recall was at most 84%. Another attempt is described in [4]: Information about protein-interactions was extracted from a data set of 1.2 million sentences that were taken from biomedical abstracts. They achieved a precision of 91%, but with a poor recall of only 21%. We recommend [3] and [12] for a more comprehensive overview of the "accomplishments and challenges" of text mining.

As a first step towards a better management of biomedical literature, controlled vocabularies like *MeSH*[2] and the *Gene Ontology*[3] have been created. They serve to classify biomedical publications and to link them to other resources. *GoPubMed*[4], for example, is a search engine that connects the abstracts from PubMed with the formal structure of the Gene Ontology. Thus a researcher can exploit the Gene Ontology for the search of relevant literature. Such tools are very valuable for scientists and there has been a notable progress in the last years, but it will never be possible to extract all the information correctly. There is inherent ambiguity and vagueness in natural language that prevents its perfect processing by computers.

For this reason we present an alternative approach: The authors of scientific articles formally summarize their own results. Such formal summaries are added to the articles which makes them processable by computers. This requires a formal language that on the one hand is easy to learn and understand, and on the other hand is expressive enough to represent even complicated scientific results. It is clear that this approach is not applicable for papers that have been

[1] `http://www.pubmed.gov`

[2] `http://www.nlm.nih.gov/mesh/meshhome.html`

[3] `http://www.geneontology.org`

[4] See [5] and `http://www.gopubmed.org`

written without the formal summaries, and that means that we still need NLP or manual extraction for such papers. Thus we propose rather a concept for the future than a solution for today's problems. To explore our approach we use Attempto Controlled English as knowledge representation language.

1.2 Formalization of Scientific Results

Since we want to access scientific results by computers, we have to formalize this knowledge at some point. Today researchers write their results in natural language. To extract these results and to formalize them, manual or computer-supported text mining is necessary. Thus the formalization is accomplished by computer-programs or by humans, and in either case it is done without the help of the corresponding researchers. The article is the only source of information. Since such articles are highly domain-specific, they require a lot of background knowledge. Therefore the formalization is a very demanding task, even for humans. Altogether this causes a lot of knowledge to be lost in the vast amount of biomedical literature.

We claim that most of these problems can be solved, if we simply let the authors of scientific articles formalize their own results. The researchers themselves are the most qualified to understand their results, and thus they can give the most precise formal representation. This is not even a big extra-effort for a scientist, since he already has a – more or less – formal model of the domain in his mind, and must write an abstract anyway. He just needs to learn how to express his knowledge in a formal way. This means that we need to provide an intuitive, yet formal language in which a scientist can write his results.

1.3 Attempto Controlled English

Attempto Controlled English (ACE)[5] is a controlled natural language that has been developed by Norbert E. Fuchs and his group at the University of Zurich. ACE is a subset of natural English with a restricted grammar. There are no limitations on the vocabulary, apart from some function words with predefined meanings (e.g. 'every', 'of'). ACE looks like English, but it is in fact a formal language, which means that texts can be translated unambiguously into first-order logic. Some ACE sentences would be ambiguous in natural English, but ACE provides interpretation rules that allow in each case only one reading. The report [13] contains a comprehensive description of the syntax of ACE.

In order to be able to write ACE texts, one has to learn the restrictions on the grammar. Thus, like every formal language, ACE has to be learned. However, since it looks like natural English, everyone is able to understand ACE texts with almost no training. This is a big advantage over other formal languages.

The Attempto parser APE[6] translates ACE texts into Discourse Representation Structures [6]. Such structures are equivalent to expressions in first-order logic, and thus every ACE sentence has a logical representation. Furthermore,

[5] See [7], [8], and `http://www.ifi.unizh.ch/attempto/`

[6] `http://www.ifi.unizh.ch/attempto/tools/`

first-order logic	$\forall X(protein(X) \rightarrow \exists Y(terminus(Y) \wedge has(X,Y)))$
DL	$Protein \sqsubseteq \exists has.Terminus$

OWL (RDF/XML)	`<owl:Class rdf:ID="Protein">`
	`  <rdfs:subClassOf>`
	`    <owl:Restriction>`
	`      <owl:onProperty rdf:resource="#has"/>`
	`      <owl:someValuesFrom rdf:resource="#Terminus"/>`
	`    </owl:Restriction>`
	`  </rdfs:subClassOf>`
	`</owl:Class>`

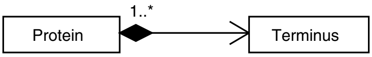
UML	
ACE	Every protein has a terminus.

Fig. 1. Comparison of first-order logic, DL, OWL, UML, and ACE

APE creates a paraphrase that shows the interpretation of an ACE text. If a writer is not familiar with the ACE interpretation rules, then he can check the paraphrase for the validation of his ACE text.

1.4 Comparison of Knowledge Representation Languages

In order to show the benefits of ACE, we compare it with four other knowledge representation languages: first-order logic [9], Description Logics (DL) [15], Web Ontology Language (OWL) with its RDF/XML-syntax [14], and Unified Modeling Language (UML) [2].

We have to state that these four languages are not independent. DL and ACE build upon first-order logic, and DL are the basis for OWL. While first-order logic and DL focus on the theoretical concepts of knowledge, OWL, UML, and ACE concentrate on the implementation and application of knowledge representation. Nevertheless we dare to give a direct comparison between these five languages.

Figure 1 shows how the fact 'everything that is a protein has a terminus' is expressed in the five different languages. The OWL representation (using the RDF/XML syntax) is the most verbose and – from the human perspective – the least readable one. The representations in first-order logic and DL are more concise, but they are still not understandable for people who are not familiar with formal notations. The graphical notation of UML looks nice, but for a non-specialist it is hard to guess the meaning of all the shapes and arrows. The ACE representation, in contrast, should be immediately understandable for any English speaking person. It looks perfectly like natural English and thus the reader might not even recognize that it is a formal language.

We can state that controlled natural languages like ACE minimize the gap between machines and humans. A reader is able to understand such languages with almost no training. Furthermore, writing sentences in a controlled natural language is possible with only little effort, especially if the writer is supported by an authoring tool (see Sect. 3.3).

2 Ontology for Protein Interactions in ACE

In order to have a clear basis for the formal representation of scientific knowledge, we defined an ontology for proteins and their interactions. This section shows how ACE can be used as an ontology language, and introduces our ontology for protein interactions.

2.1 Ontologies

The main goal of an ontology is to provide a *shared understanding* of a certain domain. This shared understanding can serve as basis for the communication between people, for the interoperability between systems, for the improvement of reusability and reliability of software systems, and for the specification of software [20]. Furthermore ontologies are an excellent basis for the formal representation of knowledge [11].

Ontologies are not yet broadly established in science, but they are expected to gain a very important role in the future, especially in life sciences. The *Gene Ontology* is the most famous example, although it is actually more a controlled vocabulary than a real ontology.

2.2 Ontology Elements

In order to provide basic structures for ontologies in ACE, we adopt the elements of DL – individuals, concepts, and roles – and we call them *ontology elements.* Furthermore we introduce an additional structure: context information.

Individuals. Individuals stand for single objects of the domain. They are represented in ACE as *proper names* like ‘Bub1’ (that stands for a protein) or ‘Alzheimer’ (that stands for a disease).

Concepts. Concepts stand for sets of objects, and there are two possibilities to express them in ACE. *Common nouns* are the most straight-forward way. The noun ‘protein’, for example, can stand for the concept of all proteins. As a second possibility we can use *adjectives* (in their positive form). The adjective ‘organic’, for example, can be used for the concept of all organic substances.

Roles. Roles stand for binary relations between objects, and they can be expressed in four different ways. First of all, we can use *transitive verbs* for expressing roles. For example, we can use ‘interacts-with’ to express a relationship between proteins. Next, we can combine transitive verbs with *adverbs.* For example, we can use the adverb ‘directly’ together with the transitive verb ‘interacts-with’ to express the role ‘directly interacts-with’. As a third possibility we can use *of-constructs* like ‘is a part of’. Due to the syntax of ACE, ‘of’ is the only allowed preposition for nouns. Finally, we

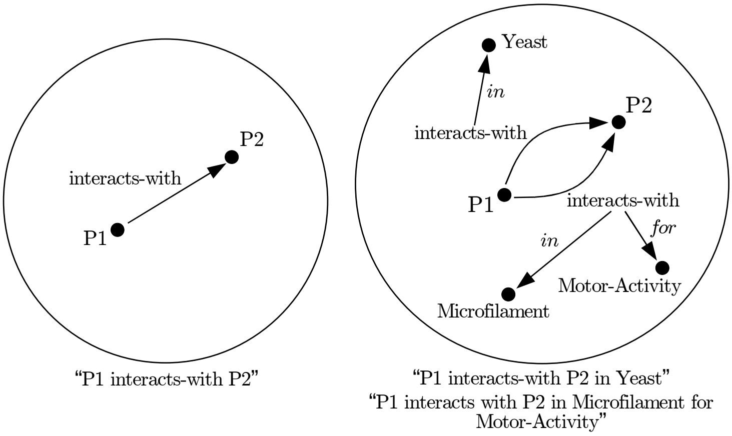


Fig. 2. Context information

can use *constructs with comparative forms of adjectives* like 'is larger than'. Such constructs typically represent transitive relationships.

Context Information. The examination of the results of scientific papers on protein interactions showed that normal roles are often not sufficient to express the needed information. We can express simple statements like 'P1 interacts-with P2', but we cannot express statements with contextual information like 'P1 interacts-with P2 in Yeast' or 'P1 interacts-with P2 in Microfilament for Motor-Activity'. In order to be able to express such results, we want to allow roles to have such additional information. In natural English we usually express such information with prepositional phrases, and this is exactly the way we will do it in ACE. Figure 2 illustrates the examples without and with context information.

Using these ontology elements, we can state for example

P1 is a protein and directly interacts-with P2 in Yeast.

where 'P1', 'P2', and 'Yeast' are individuals, 'protein' stands for a concept, and 'directly interacts-with' stands for a role. The phrase 'is a' is used to assign the individual 'P1' to the concept 'protein'. The conjunction 'and' connects the statements flanking left and right. The preposition 'in', finally, connects to the context 'Yeast'.

2.3 Ontology for Protein Interactions

Since we found no existing ontology that fits our needs, we had to create our own ontology for protein interactions. First, we defined concepts that allow us to make statements about the structure of proteins and protein-complexes. For the sake of a clear structure, we introduced the concept *protein-unit*, which is

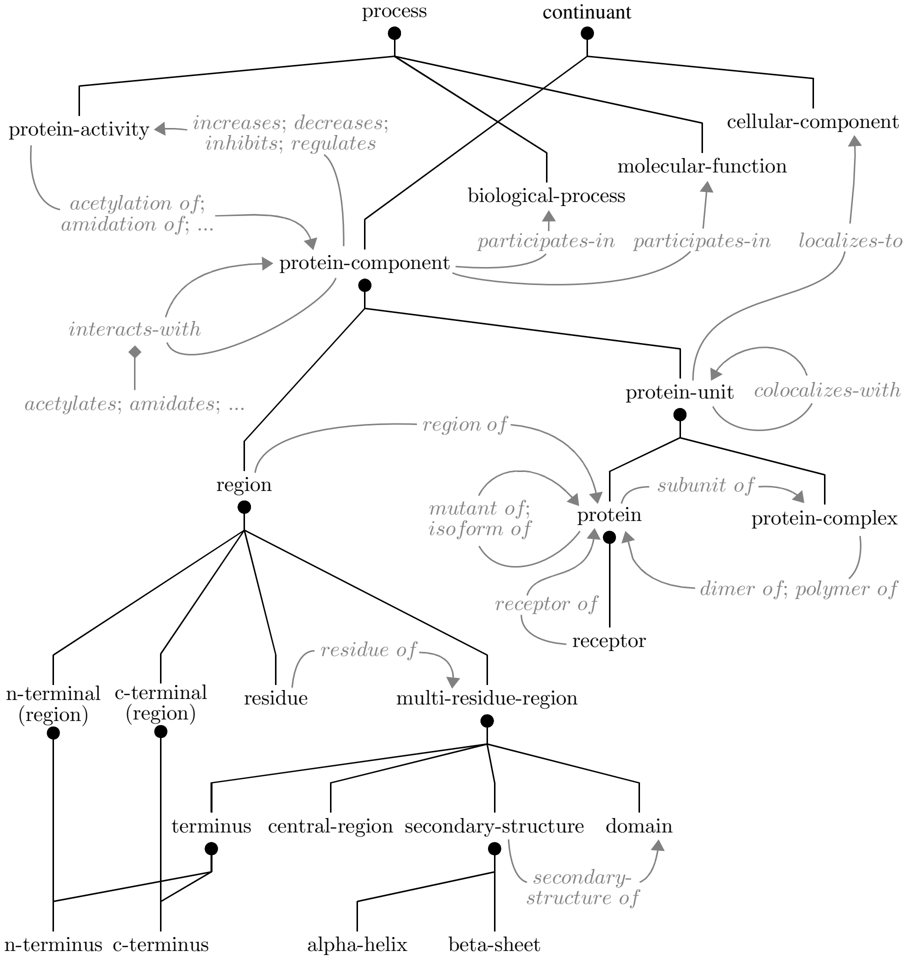


Fig. 3. The structure of the ontology for protein interactions

either a protein or a protein-complex, and *protein-component*, which is either a protein-unit or a region of a protein. In order to describe the structure of such regions, we defined concepts like ‘residue’, ‘secondary-structure’, and ‘domain’.

Next, we defined the roles for the description of interactions between proteins like ‘interacts-with’ or ‘binds’. We can also express more complicated interactions like ‘increases the phosphorylation of’.

Furthermore, we defined some concepts for expressing additional information about proteins, like the localization to a certain cellular component or the participation in a certain process. The big picture of this ontology is shown in Fig. 3.

3 ACE Summaries

Our goal is to show how scientists could write formal summaries of their results. There are some questions that naturally arise: What are these results about? How complex is it to formulate them in a formal language? In the following we present an empirical study of the feasibility of our approach.

3.1 ACE Summaries for 89 Selected Articles

Since we want to show how results of papers about protein interactions could have been written in ACE in the first place, we picked 89 *Elsevier*-articles that concern protein interactions. Such articles mostly have a section called "Results" which is subdivided into subsections. The headings of these subsections are short descriptions of the corresponding results. It turned out that these headings are highly suitable for a manual translation into ACE. Please note that the intended methodology is *not* to express the results first in natural language and then to translate them into ACE. We do this just to demonstrate the feasibility of our approach.

The 89 articles contain 457 such headings. 184 of them are ignored, because they are not formulated as facts (e.g. "Functional characterization of Pellino2"[7]) or because they contain information that is not about protein interactions.

total:		457	*(100%)*
ignored:	(not a fact)	87	*(19%)*
	(off-topic)	97	*(21%)*
used:		273	*(60%)*

We then tried to translate the 273 remaining headings into ACE. For 154 of them there is a perfect match, which means that the complete information can be expressed in ACE; e.g. the heading "Interaction of Act1 with TRAF6"[8] can be rephrased perfectly as "Act1 interacts-with TRAF6". For another 62 headings only a part of the information is expressed; e.g. the heading "The mtFabD protein is part of the core of the FAS-II complex"[9] can only partially be rephrased as "MtFabD is a subunit of FAS-II". For the remaining 57 headings there is no translation at all.

used:		273	*(100%)*
matched:	(perfect)	154	*(56%)*
	(partial)	62	*(23%)*
unmatched:		57	*(21%)*

Let us take a closer look at the reason, why 119 headings cannot be rephrased in ACE at all, or only partially. 56 of them could not be rephrased because their content is not covered by our model, but they could be expressed with an extended model. Another 21 headings describe relations of relations, like the heading "Kal-GEF1 activation of Pak does not require GEF activity"[10]. In this

[7] See article *PMID 12860405.*

[8] See article *PMID 12459498.*

[9] See article *PMID 16213523.*

[10] See article *PMID 15950621.*

case, there is a relation between two objects ("Pak activates Kal-GEF1") and this relation itself stands in another relation ("... does-not-require GEF-activity"). At the moment, we cannot express such structures in ACE in a satisfying way. But there are attempts to extend the language ACE, and we hope that we will be able to express such statements in the future. Furthermore there are 11 headings with fuzzy statements (e.g. "ANKRD contains potential CASQ2 binding sequences ..."[11]) and 31 headings that we could not understand without reading the whole article.

not perfectly matched:	119	*(100%)*
not covered by our model:	56	*(47%)*
relations of relations:	21	*(18%)*
fuzzy:	11	*(9%)*
not understood:	31	*(26%)*

Thus, altogether we could rephrase 79% of the relevant headings, either partially or perfectly. This makes us confident that our approach is feasible for practical use. The reason, why 119 headings are not rephrased perfectly, is mostly our simple model and our lack of understanding. If we used a more detailed model, and if we let the scientists themselves express their own results in ACE, then we expect to be able to express much more than 79% of the results.

3.2 ACE Summary as an Integral Part of an Article

Since ACE looks like natural English, every reader of a scientific article is able to understand ACE texts. Thus the ACE summary of the results could be an integral part of the article. Figure 4 shows how an article with an ACE summary could look like[12]. Figure 5 shows the corresponding logical representation as a Discourse Representation Structure (consult [6] for details). As we see, the natural looking ACE summary can be translated automatically into a formal representation which is processable by computers.

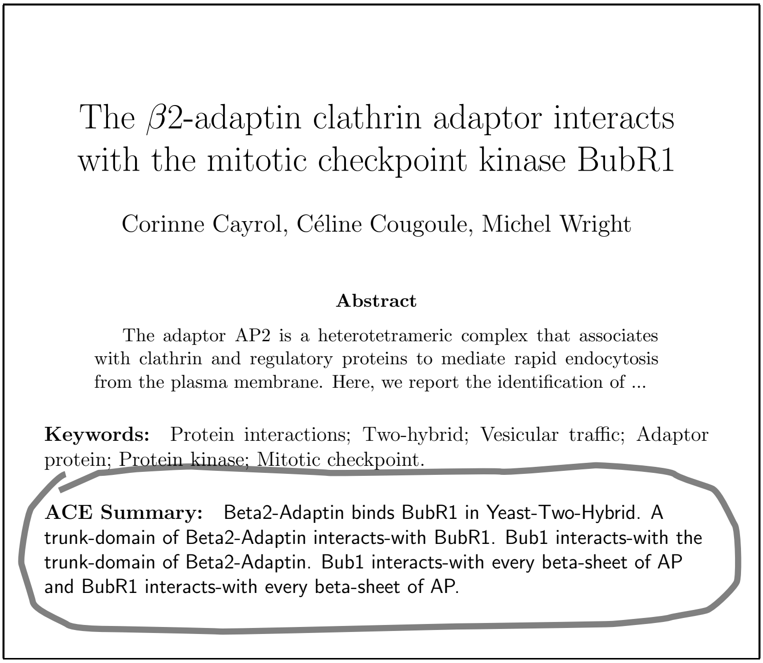
Together with the abstract and a keyword list, the ACE summary gives a concise insight into the content. In contrast to the abstract, the ACE summary is readable by both, humans and machines; and in contrast to the keyword list, the ACE summary does not only mention the objects of interest, but describes the relations among them. Thus, every published article could be a contribution to a constantly growing knowledge base.

3.3 Authoring Tool

Now we sketch a tool that would help writing ACE texts. It would guide the user step by step and would need almost no training. Similar systems are the look-ahead editor *ECOLE* [17,18], the natural language interface *LingoLogic* [19], and the *Ginseng*-system [1]. Our tool would solve several problems:

[11] See article *PMID 15698842*.

[12] Article *PMID 12419313* is used for this example.



The β2-adaptin clathrin adaptor interacts with the mitotic checkpoint kinase BubR1

Corinne Cayrol, Céline Cougoule, Michel Wright

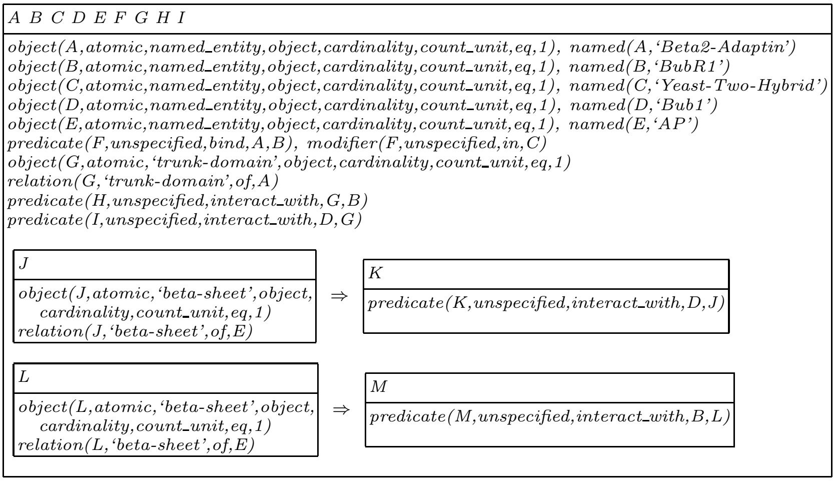
Abstract

The adaptor AP2 is a heterotetrameric complex that associates with clathrin and regulatory proteins to mediate rapid endocytosis from the plasma membrane. Here, we report the identification of ...

Keywords: Protein interactions; Two-hybrid; Vesicular traffic; Adaptor protein; Protein kinase; Mitotic checkpoint.

ACE Summary: Beta2-Adaptin binds BubR1 in Yeast-Two-Hybrid. A trunk-domain of Beta2-Adaptin interacts-with BubR1. Bub1 interacts-with the trunk-domain of Beta2-Adaptin. Bub1 interacts-with every beta-sheet of AP and BubR1 interacts-with every beta-sheet of AP.

Fig. 4. Article with ACE summary



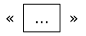
```
A B C D E F G H I
object(A,atomic,named_entity,object,cardinality,count_unit,eq,1), named(A,'Beta2-Adaptin')
object(B,atomic,named_entity,object,cardinality,count_unit,eq,1), named(B,'BubR1')
object(C,atomic,named_entity,object,cardinality,count_unit,eq,1), named(C,'Yeast-Two-Hybrid')
object(D,atomic,named_entity,object,cardinality,count_unit,eq,1), named(D,'Bub1')
object(E,atomic,named_entity,object,cardinality,count_unit,eq,1), named(E,'AP')
predicate(F,unspecified,bind,A,B), modifier(F,unspecified,in,C)
object(G,atomic,'trunk-domain',object,cardinality,count_unit,eq,1)
relation(G,'trunk-domain',of,A)
predicate(H,unspecified,interact_with,G,B)
predicate(I,unspecified,interact_with,D,G)

  J                                          K
  object(J,atomic,'beta-sheet',object,   ⇒   predicate(K,unspecified,interact_with,D,J)
      cardinality,count_unit,eq,1)
  relation(J,'beta-sheet',of,E)

  L                                          M
  object(L,atomic,'beta-sheet',object,   ⇒   predicate(M,unspecified,interact_with,B,L)
      cardinality,count_unit,eq,1)
  relation(L,'beta-sheet',of,E)
```

Fig. 5. DRS-representation of the ACE summary

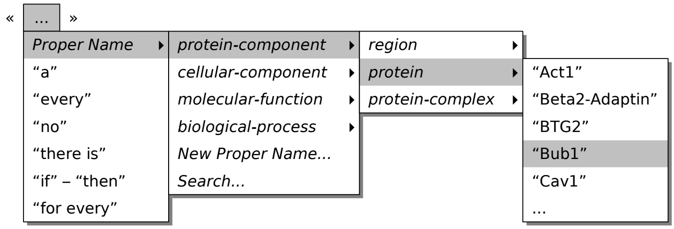
- The tool would help the user to comply with the standard nomenclature. The user would only be allowed to use the defined words. It would also prevent typing errors.
- It would make sure that the created sentences comply with the ACE syntax. At every stage, the tool would allow to proceed only in a way that leads to a correct ACE sentence. Thus the user would not need to know about the syntax of ACE.
- The tool would be aware of the structure of the ontology. In this way it would make sure, for example, that the domains and ranges of roles are respected.

We give now an example how this tool could be used. Suppose that an author of the article that is shown in Fig. 4 wants to write down the fact that the protein *Bub1* interacts with the protein *β2-Adaptin* via its *trunk domain*.

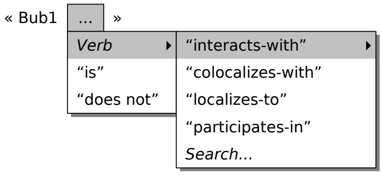
The sentences are created step by step by a simple menu. At the beginning there is just an empty sentence that might look like this:



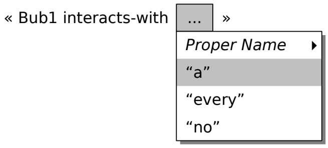
The quotes indicate the beginning and the end of the sentence and the box in the middle is used to create the content. If the user clicks on it, then a menu is displayed that shows the different options for beginning a sentence. Since we want to talk about the protein *Bub1* we first insert 'Bub1' as a proper name. This looks as follows.



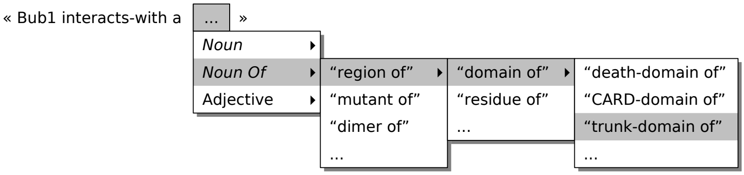
Proper names are hierarchically structured and the menu allows to navigate through this hierarchy. Alternatively, we can use the search option to find a certain term, or we can create a new proper name on-the-fly. In the next step we get



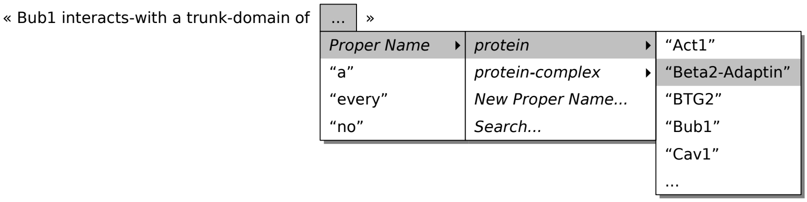
where the proper name 'Bub1' is now fixed as the beginning of the sentence, and we have a new menu with different entries. We want to express the interaction with another protein, and thus we choose the verb 'interacts-with'. Like proper names, verbs are hierarchically structured and we can navigate through this hierarchy. In the next step we get


« Bub1 interacts-with ... »

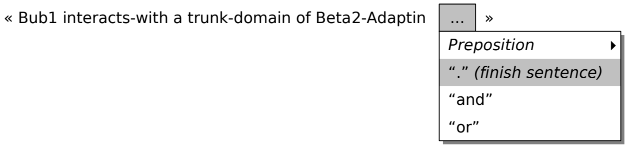
where we can define the second participant of the protein-interaction. Since we want to state that the interaction goes via a *trunk domain* of the protein *β2-Adaptin*, we first have to add the article 'a'. Then we get


« Bub1 interacts-with a ... »

where we can choose the 'trunk-domain of'-relation. Like proper names, such of-relations are structured in hierarchies through which we can navigate (the same holds for nouns and adjectives). After that we get


« Bub1 interacts-with a trunk-domain of ... »

where we can specify the second protein 'Beta2-Adaptin'. Finally we get


« Bub1 interacts-with a trunk-domain of Beta2-Adaptin ... »

where we could use prepositions to add context information, e.g. to specify the organism in which the interaction takes place. In our example, we now finish the sentence with a full stop.

For the creation of this sentence we did not need any further knowledge about ACE. Every person that is familiar with English and knows how to handle a simple menu, is able to create ACE texts. However, to make such a tool really user-friendly we will need a lot of usability testing, as it is done – with promising results – for the *Ginseng*-system [1].

4 The Benefits of Our Approach

The preceding sections showed what needs to be done to express scientific results about protein interactions in ACE. Now it is time to take a look at the benefits.

Today there are many databases that contain life science data, but they are mostly unsynchronized, incomplete, and often not up-to-date. With our approach it would be much easier to provide complete and consistent databases.

Imagine that all the scientific papers about protein interactions summarize their results in ACE. We could use these formal summaries to build up a dynamically growing knowledge base about protein interactions. Of course, we would also have to collect all the knowledge that is contained in old papers. For these, we still need some form of classical text mining. But once we have such a knowledge base that is continuously updated with the results of new papers, then we would be able to answer many questions. We present now some examples.

Are some results consistent with an existing knowledge base or with other papers? We can check, whether an ACE summary is consistent with an existing knowledge base. If this knowledge base contains common knowledge, then the results should be consistent, or otherwise it can be seen as an appeal against the common knowledge.

Without formal declarations, it is impossible to check a paper for consistency. Probably there exist scientific papers that contain results which are inconsistent with the common knowledge. But since this can be very difficult to find out, neither the author nor the readers might realize the special status of the results.

In the same way we can check, whether there exist papers that contradict a certain paper. That would mean that different researchers claim contradictory results. Being aware of such a contradiction might lead to a dialogue between the corresponding scientists, which might entail better and consistent results.

Are some results (or parts of it) already known? With our formal approach we can check whether a certain result, or a part of it, is already known. Results that are already considered common knowledge are usually not worth to be described as results of scientific papers (unless they contain more detailed information or if additional evidence is given). Thus it is very valuable to be able to run a check, whether a certain result is already contained in the knowledge base or not.

Furthermore a researcher might want to check, whether there exists scientific literature that has arrived at the same or similar results. Altogether our approach would help the researchers to save a lot of time, since they would not need to search "manually" for the relevant literature.

Is there a known answer for a certain question? If someone – researcher or not – has a specific question about the domain (e.g. protein interactions), then we would be able to give automatically an answer.

Indeed, there exist already systems like *MEDIE*[13] that provide some sort of answer extraction using natural language processing. But such systems have serious shortcomings: There is always a trade-off between precision and recall, and only very simple queries are allowed. Furthermore, we cannot find answers that are spread over multiple articles.

[13] `http://www-tsujii.is.s.u-tokyo.ac.jp/medie/`

type	Bub1
supertypes	Kinase – Enzyme – Protein – Molecule
subtypes	BubR1
interacts directly with	Beta2-Adaptin, Cdc20, Mad3
interacts indirectly with	Mad2, APC
associates with	Cdc20, Mad3
phosphorylates	Bub1, Bub3
localizes to	Kinetochore, Chromosome
participates in	Cell-Communication, Signal-Transduction

Fig. 6. Overview over the object 'Bub1'

What is known about a certain object of interest? In some cases we do not want to ask a specific question, but we rather want to get an overview of a single object of interest (e.g. the protein *Bub1*). If we ask for information about such an object then we might get something as shown in Fig. 6. Such an overview could be used for a dynamic hypertext representation. This would allow us to navigate through the whole knowledge base, e.g. with an ordinary web browser. New papers that are submitted can be integrated *automatically* and thus such a web interface would be always up-to-date.

How are some objects of interest related? Instead of focusing on one single object, we might want to have an overview of the interrelations of a certain group of objects. We could extract, for example, the *interacts-with*-relations of all proteins and use this data for further examination, like the detection of clusters or hot-spots. Such examinations are already common in the research on proteins (e.g. [10], [16]), but only with restricted data. With our approach we could consider every interaction that has been published.

5 Outlook

We suggest an approach of using controlled natural language for making the results of scientific papers readable and – to some degree – understandable by computers. But in order to achieve this goal, there is still a lot of work to do. For example, we need an authoring tool as sketched in Sect. 3.3, that would support the authors of scientific papers in the creation of ACE summaries. A prototype of such a tool does already exist. Furthermore, we need tools for the definition of ontologies and for the collection and management of knowledge.

Besides all these technical requirements, there are also political ones. There must be a commitment among the scientists of the corresponding field of research – or at least among a large part of them – that scientific articles get summarized in ACE. If such a summary is optional then there is little hope that it gets established.

This is the point where the publishers and editors have to come into play. The publishers would have to make ACE summaries a mandatory part of the articles, and the editors would have to check whether these summaries are correct and complete. The creation of a formal summary should be an additional requirement to consider when writing a scientific article, besides all the requirements that already exist today (e.g. about the abstract, the keyword list, and the reference list). The formal summaries can also be seen as a robust indicator for the value of a scientific paper. Information that is already known and redundant information could be ignored automatically, and wrong statements are likely to be detected at some later point in time. Thus we could use the formal summaries to quantify and qualify the contribution of a certain author, institute, or journal.

Due to the immense benefits such a system would bring along, we believe in the great potential of our approach. It could be a first step towards better communication and persistence of biomedical knowledge.

References

1. Abraham Bernstein, Esther Kaufmann, Christian Kaiser. *Querying the Semantic Web with Ginseng: A Guided Input Natural Language Search Engine*. Department of Informatics, University of Zurich, 2005
2. Grady Booch, James Rumbaugh, Ivar Jacobson. *The Unified Modeling Language User Guide*, First Edition. Addison Wesley, 1998
3. Aaron M. Cohen, William R. Hersh. *A survey of current work in biomedical text mining*. In *Briefings in Bioinformatics*, 6(1):57-71, 2004
4. Nikolai Daraselia, Anton Yuryev, Sergei Egorov, Svetalana Novichkova, Alexander Nikitin, Ilya Mazo. *Extracting human protein interactions from MEDLINE using a full-sentence parser*. In *Bioinformatics*, 20(5):604-611, 2004
5. Andreas Doms, Michael Schroeder. *GoPubMed: exploring PubMed with the Gene Ontology*. In *Nucleic Acids Research*, 33:W783-W786, 2005
6. Norbert E. Fuchs, Stefan Hoefler, Kaarel Kaljurand, Tobias Kuhn, Gerold Schneider, Uta Schwertel. *Discourse Representation Structures of ACE 4 Sentences*, Technical Report ifi-2006.07. Department of Informatics, University of Zurich, 2006, `ftp://ftp.ifi.unizh.ch/pub/techreports/TR-2006/ifi-2006.07.pdf`
7. Norbert E. Fuchs, Kaarel Kaljurand, Gerold Schneider. *Attempto Controlled English Meets the Challenges of Knowledge Representation, Reasoning, Interoperability and User Interfaces*. The 19th International FLAIRS Conference (FLAIRS'2006), 2006
8. Norbert E. Fuchs, Uta Schwertel, Rolf Schwitter. *Attempto Controlled English – Not Just Another Logic Specification Language*. In *Logic-Based Program Synthesis and Transformation*, Eighth International Workshop LOPSTR'98, Lecture Notes in Computer Science 1559, Springer, 1999, `http://www.ifi.unizh.ch/attempto/publications/papers/LOPSTR98.pdf`
9. Melvin Fitting. *First-Order Logic and Automated Theorem Proving*, Second Edition. Springer, New York, 1996
10. L. Giot, J. S. Bader, C. Brouwer, A. Chaudhuri, et al. *A Protein Interaction Map of Drosophila melanogaster*. In *Science*, 302(5651):1727-1736, 2003
11. Thomas R. Gruber. *Toward Principles for the Design of Ontologies Used for Knowledge Sharing*. In *International Journal of Human-Computer Studies*, 43 (5-6):907-928, 1995

12. Lynette Hirschman, Jong C. Park, Junichi Tsujii, Limsoon Wong, Cathy H. Wu. *Accomplishments and challenges in literature data mining for biology.* In *Bioinformatics Review*, 18(12):1553-1561, 2002
13. Stefan Hoefler. *The Syntax of Attempto Controlled English: An Abstract Grammar for ACE 4.0*, Technical Report ifi-2004.03. Department of Informatics, University of Zurich, 2004,
`ftp://ftp.ifi.unizh.ch/pub/techreports/TR-2004/ifi-2004.03.pdf`
14. Deborah L. McGuinness, Frank van Harmelen. *OWL Web Ontology Language Overview.* W3C Recommendation, 2004,
`http://www.w3.org/TR/2004/REC-owl-features-20040210/`
15. Daniele Nardi, Ronald J. Brachman. *An Introduction to Description Logics.* In *The Description Logic Handbook: Theory, Implementation, and Applications*, Cambridge University Press, 2003
16. Benno Schwikowski, Peter Uetz, Stanley Fields. *A network of protein-protein interactions in yeast.* In *Nature Biotechnology*, 18:1257-1261, 2000
17. Rolf Schwitter, Anna Ljungberg, David Hood. *ECOLE: A Look-ahead Editor for a Controlled Language.* In *Proceedings of EAMT-CLAW03, Controlled Language Translation*, Dublin City University, 141-150, 2003
18. Rolf Schwitter, Marc Tilbrook. *Let's Talk in Description Logic via Controlled Natural Language.* To be presented at: Logic and Engineering of Natural Language Semantics 2006 (LENLS2006), Japan, 2006
19. Craig W. Thompson, Paul Pazandak, Harry R. Tennant. *Talk to Your Semantic Web.* In *IEEE Internet Computing*, 9(6):75-79, 2005
20. Mike Uschold, Michael Gruninger. *Ontologies: Principles, Methods and Applications.* In *Knowledge Engineering Review*, 11(2), 1996
21. Alexander Yeh, Alexander Morgan, Marc Colosimo, Lynette Hirschman. *BioCreAtIvE Task 1A: gene mention finding evaluation.* In *BMC Bioinformatics*, 6, 2005