Pacific Symposium on Biocomputing 5:538-549 (2000)

Automatic Extraction of Protein Interactions from Scientific Abstracts

James Thomas,[*,†] David Milward,[#] Christos Ouzounis,[*] Stephen Pulman[#,†] and Mark Carroll[*]

[#] *SRI International, 23 Millers Yard, Mill Lane, Cambridge, CB2 1SD*
(`http://www.cam.sri.com`*)*

[†] *University of Cambridge Computer Laboratory, New Museums Site,*
Pembroke St, Cambridge, CB2 3QG

[*] *Computational Genomics Group, The European Bioinformatics Institute, EMBL*
Cambridge Outstation, Cambridge, CB10 1SD

This paper motivates the use of Information Extraction (IE) for gathering data on protein interactions, describes the customisation of an existing IE system, SRI's Highlight, for this task and presents the results of an experiment on unseen Medline abstracts which show that customisation to a new domain can be fast, reliable and cost-effective.

1 Introduction

There is now a vast corpus of molecular biology literature available electronically, e.g. the abstracts in Medline (PubMed). However, much of this data is only stored in free text format which means that querying for specific information is not very efficient. Using keywords to narrow the search often produces far more candidates than can be properly read (or processed). While some abstracts *are* converted to a form of database record, the information in them may not be represented in a structured way, and the increasing volume of data means that large numbers of abstracts are not processed at all. Research on protein-protein interactions in particular has generated large volumes of information that are not accessible in a computer-readable form, e.g. [2,6,7].

The availability of protein-protein interactions in a structured form should avoid duplication of research efforts: not only because interactions of interest can be located easily if the work has already been performed, but also because it will be possible to search more accurately for information about results on the same protein in other species. It should then be feasible to build a database which can be used to discover regularities and connections which do not emerge by looking only at isolated pieces of information. Species-specific knowledge bases, such as EcoCyc [5] which represent the genome and metabolism of model organisms, have not incorporated data on all known protein-protein interactions, partly because of the difficulty of obtaining this information. Recently, however, there has been growing interest in the use of automatic extraction and analysis of protein-protein interactions, e.g. [2,3,8].

2 What is Information Extraction?

Information Extraction (IE) is an application of natural language processing that takes a piece of free text and produces a structured representation (a template consisting of slots to be filled) of the points of interest in it. This representation can then be easily transformed to a database record, a row in a table, or some other convenient notation. The input text is syntactically and semantically analysed to locate the entities of interest and the properties ascribed to them which are then extracted and used to fill in the template slots.

Information Extraction is *not* Information Retrieval (IR), in which key words are used to select relevant *documents* from some corpus (or the Internet, e.g. Altavista [a]), but could easily post-process IR output. Successful applications of IE can be found in several large industries, as well as in some military and intelligence areas. Information extraction systems take over a task that would otherwise have to be performed by hand: for example, analysing incoming news feeds for a certain type of incident and creating summaries. Although the accuracy of these systems may not rival that of a human expert, they are able to process larger quantities of text very quickly and economically and provide data for other processes such as data mining and statistical analysis.

3 Highlight

Highlight is a general-purpose IE engine developed at SRI Cambridge for use in commercial applications. It incorporates several of the techniques used by SRI Menlo Park's Fastus [4], a leading performer in the MUC evaluations [b] of information extraction. The summary in Table 1 was extracted automatically from free text by processing a series of newswire articles looking for gas or oil company projects (e.g. wells, pipelines etc.), and the companies involved. The templates are sorted via countries and locations so that the user can easily find out about competitor activities in a particular area. By viewing the summary in a browser it is possible to click on the *Source* entries which hyperlink to

[a] http://www.altavista.com/

[b] http://www.muc.saic.com/

Country	**Location**	**Project Type**	**Partners**	**Source**
Arctic	western Canadian Arctic	the pipeline	Shell Canada Ltd.; Gulf Canada Resources Ltd.; Esso Resources Canada Ltd.	*1753*
USA	offshore Louisiana	the well	Diamond Shamrock Offshore Partners	*2312*
USA	south Texas	17 oil and gas fields	Texaco Inc.	*1399*

(wsj_1753.txt) A Canadian government agency conditionally approved proposed exports to the U.S. of natural gas from big, untapped fields in the Mackenzie River delta area of the western Canadian Arctic. Three companies, Esso Resources Canada Ltd., Shell Canada Ltd. and Gulf Canada Resources Ltd., applied to the Canadian National Energy Board to export 9.2 trillion cubic feet of Mackenzie delta natural gas over 20 years starting in 1996. To be economically feasible, the pipeline requires almost a doubling of natural gas export prices.

Table 1: An example

the corresponding section of the original text. The first entry in the summary comes from the article below the table.

Highlight carries out processing in several different stages. First, an input text is tokenised (i.e. broken into separate words and sentences). Each word is then 'tagged' (using Hidden Markov Models) with an appropriate part of speech such as 'noun' or 'verb'. Next, sequences of words are grouped into phrases of various types by cascaded finite state machines. Then phrases referring to the entities or events of interest are recognised using pattern matching and statistical methods. Groups of coreferring phrases are identified and linked. Finally, templates and schemata are instantiated so as to contain the desired components of information to be extracted.

The usual measures of success in IE are *recall* and *precision*. The former is the percentage of possible templates that are found while the latter is the percentage of extracted templates that are correct. For example, if we know that there are 100 references to gas projects in some corpus and our system finds 60 of which 45 are correct, i.e. 15 were found in error—perhaps not gas projects at all—then the precision is $45/60 = 75\%$ and recall is $45/100 = 45\%$.

It is usually possible to sacrifice higher precision for better recall and vice versa, e.g. for an application involving extraction of information concerning competitors the system is tuned to provide good recall i.e. not to miss required information, but not so good precision i.e. there may be inappropriate records produced which can be pruned out by hand if necessary.

4 Customising Highlight for biological literature

Similar to the gas example above, the project described here aimed to extract occurrences of protein interactions from Medline abstracts, producing a database of protein pairs characterized by a type of interaction. This was done by customising the existing Highlight system tuned to produce high precision (accuracy) but lower recall (coverage), a suitable strategy because there is such a large volume of material to be analysed that if an interaction is missed in one abstract the likelihood is that it will be found elsewhere. Customising an existing system, rather than constructing a bespoke program, can potentially reduce development time and costs by, for example, re-use of tried-and-tested components.

The main effort in customising a system like Highlight is in (a) adapting the natural language (NL) component so as to be able to correctly recognise the relevant entities and events, (b) developing a set of templates or outlines of the kinds of information that is of interest, and (c) developing the patterns that will decide how to slot the items and events into the templates. We discuss (a) in Section 4.1 and (b,c) in Section 4.2.

4.1 Adapting Highlight's NL components

Adaptation of the natural language processing components requires a corpus of texts representative of those that will be encountered in the field. On the basis of initial processing of these texts, we will generally need to add new vocabulary, e.g. acronyms, abbreviations, or technical terms characteristic of the domain and syntactic constructs not already covered in the general purpose analysis engine. Also, methods for 'reference resolution' have to be refined. These locate alternative references (e.g. by pronouns or phrases like 'the protein') to the same entity and link them together.

Although the system guesses unknown words fairly reliably, we added extra vocabulary such as *mutagenesis, osteocalcin* and *retinoid* and also took account of the lexical peculiarities that occur in protein names including symbols such as ' \ - () in proteins like *eIF-4a, EPIY', FTZF2/HK* and *GLYS(A)* along with numbers and so on which would not normally appear in proper names.

Similarly, given that this version of the Highlight tagger was trained on financial newspaper reports, we had to correct tagging errors for words such as *associates* which was tagged as a noun (*Smith's associates*) but which we would prefer to tag as a verb (*Brf associates with TFIIB*).

Customising Highlight for a new domain is made easier by the design of the system which separates domain-specific information from a central core of linguistic processing which is general enough to be applicable in most domains.

4.2 Finding protein interactions

After customising the linguistic element, we turned to the templates: outline summaries of the information in the text that is of interest. For the information extraction technology to work effectively there must be sufficient detail in the input texts for the contents of the templates to be recognised explicitly. A set of rules (essentially pattern matching rules with a statistical component) for assigning the entities and events recognised by the natural language component to slots in the templates was written and tested.

We first analysed around 200 abstracts by hand to find common ways of describing interactions. We examined approximately 30 different verbs (including *activate, inhibit, modulate, suppress, isolate, promote, characterise* ...) and decided to concentrate on *interact (with), associate (with)* and *bind (to)* since these three, of those which occur frequently, all appear in relations directly between proteins rather than between a protein and some process. For example: *TR-beta inhibits the assembly of a functional transcription preinitiation complex* does not give us a relation between two proteins in the way that *the interaction between Tat and TFIIB* does.

Several patterns for each verb have been implemented. They are generally written at the syntax rather than lexical level which allows us to collapse multiple related patterns into a single pattern. Patterns act as filters on the tagged and parsed text, i.e. if a continous segment of the text matches the input (the top line of the pattern) and the conditions hold, then the text is rewritten as shown on the bottom line of the pattern. The pattern in Table 2 looks for a noun phrase followed by a verb and particle then another noun phrase. If it is found and the verb and particle are of interest (e.g. *interact with*) then a unique identifier is generated and a template indexed by the identifier containing the required information (the two noun phrases and the relation) is created and stored. A tag (`tvbio`) which causes later processing to make a hyperlink to the relation from the summary tables is added to the output.

We also added a method of ranking templates in order to give some mea-

```
%% A interacts with/binds to/associates with B

[NP1/tag(np,Id1), VG/tag(vg,headed_vg,Word), PP/in,   %% input pattern is
 NP2/tag(np,N2,Id2)]:                                 %% NP1, V, Particle, NP2.

    verb_particle_pair(Word,PP,VbPart),               %% desired verb-particle?
    make_new_id(Id),                                  %% new Id if input matches.
    make_template(NP1/tag(np,Id1),NP2/tag(np,Id2),VbPart,Id)
                                                      %% make template and store
   ==>                                                %% indexed by Id.
[NP1/tag(np,Id1),
 [VG/tag(vg,headed_vg,Word), PP/in]/tvbio(Id),        %% output pattern with
 NP2/tag(np,Id2)].                                    %% indexed tag.
```

Table 2: A pattern

sure of the confidence we have that the template is correct, i.e. is a desired relation between two proteins rather than an unwanted relation between two non-proteins. Several factors might form part of this score:

- The context in which the relation is found. This would include verbs such as *prove, show, suggest* ..., phrases such as *It is probable that* ..., *X suppresses interaction of Y and Z* and so on. A further option is to add a note of the "modality" of the relations in another column. This might include negation, possibility and so on.

- The confidence we have that the NP arguments are proteins. By scoring highly those relations with lexical protein name arguments we can prefer relations which we are sure of as opposed to definite descriptions or pronouns which stand a chance of being unresolved or incorrectly resolved.

- The number of times a relation occurs. This measure might provide a way of measuring, across a whole corpus, the reliability of any given relation.

For simplicity's sake, we have rated according to the second criterion and, in fact, we merely check that the NP arguments are proper names. In our task, it turns out that if we have already found an *interaction* and the items involved are proper names then there is a good chance that they are proteins. Our scoring strategy is given in Table 3. The scoring allows us to filter the templates so that we only return those with the greatest chance of being correct, i.e. to keep precision high.

NP	**Score**
proper name	50
compound noun, containing proper name	50
compound noun, not containing proper name	30
definite description, linked to proper name	25
definite description, unlinked	15
all other	0

Table 3: NP scores

5 Results

We tested the system by analysing 2565 unseen abstracts extracted from Medline with the keywords *molecular, interaction* and *protein* for year 1998 (560k words). This resulted in 2359 templates of which 782 scored 100% and 454 scored 80% which means that around 1 in 2 texts have a relation that we are interested in.

We estimate recall and precision values for the whole corpus of abstracts by taking three samples of 30 abstracts each and analysing them by hand. The amount of time and effort required to analyse a substantial sample (e.g. 10%) was prohibitive but by taking 3 samples of around 1% and comparing the results, we get a good indication of the overall population recall and precision. There is no intersection between the samples, but otherwise the abstracts that make up each set were chosen at random.

Further, we present 4 different measures of precision and recall:

ALL across all *interact, associate, bind* relations regardless of score (including relations that we would like to be able to get even though we don't currently try to get them because we have no patterns for them yet). There are 72 such relations in sample 1.

PAT across only those relations which we have written patterns for regardless of score. There are 41 such relations in sample 1.

TOP1 The PAT relations which score 100%, i.e. are between two proteins directly (as opposed to by reference resolution). There are 16 such relations in sample 1.

TOP2 The PAT relations which score 100%, i.e. are between two proteins directly (as opposed to by reference resolution), or 80%, i.e. between a protein and a compound noun which is probably a protein but doesn't conform to our criteria. There are 27 such relations in sample 1.

	Sample1		**Sample2**		**Sample3**	
Set	**Recall**	**Prec.**	**Recall**	**Prec.**	**Recall**	**Prec.**
ALL	30	65	24	72	33	72
PAT	54	65	50	72	50	72
TOP1	63	77	56	71	53	80
TOP2	63	81	38	60	54	81

Table 4: Recall and Precision (percentages) for our samples

Set	**Recall**	**Prec.**
ALL	29	69
PAT	51	69
TOP1	58	77
TOP2	55	77

Table 5: Overall results

When calculating precision in the results in Table 4 [c] we imposed a strict criteria that a template is incorrect if any element is incorrect. This includes incorrect reference resolution for NPs. However, when an NP was unresolved, e.g. *the protein*, then this was counted as correct. Some example output is given in Tables 6, 7.

Using standard hypothesis tests at the 95% level we are able to say that there is no significant difference between the precision and recall values across the 3 samples which gives us confidence that they reliably predict the actual precision and recall values of the whole test set. We can calculate the overall results in Table 5 by combining the results for each of the three samples, e.g. set TOP1 recall is calculated by the following sum $(10+8+5)/(16+15+9) = 58\%$, i.e. a grand total of recall across all three samples.

5.1 Related work

Sekimizu et. al.[8] attempt to generate automatic database entries containing relations extracted from Medline abstracts. The relations they are interested in come from the verbs *activate, bind, interact, regulate, encode, signal* and

[c] For PAT, TOP1 and TOP2 the reported recall values are based on the "true" number of occurrences of relations of the type of PAT, TOP1 and TOP2 respectively, rather than on the "true" number of all occurrences of potential relations. Thus a recall of 63% for TOP1 in sample 1 means that 63% of the potential 100% scoring relations (in the system as it stands) were recalled.

Entity	**Relation**	**Entity**	**Score**
CUL1	interact	SKP2	100
Nun	interact	RNAP	100
HSP105	associate	HSC70	100

Table 6: Examples of accepted templates

Entity	**Relation**	**Entity**	**Score**
KIV9 can	interact	LDL	100
PS	bind	beta2GPI and the binding of PS	100
others	interact	UbcH7	80

Table 7: Examples of rejected templates

function. This forms part of a larger project which includes automatic SGML tagging of abstracts before IE is performed. Their approach is to parse, determine noun phrases, spot the commonly-occurring verbs and choose the most likely subject and object from the candidate NPs in the surrounding text. They use a corpus of 898k words extracted from Medline and report precision results which range from 67.8% to 83.3% across the different verbs.

Blaschke et. al.[1] attempt to do without NL technology such as parsing and present a simple matching approach to extracting protein interactions from Medline texts. The text is broken into clauses, those which contain two proteins and an "action" verb are extracted and simple order information is used to predict the relation, e.g. *protein1 action protein2* makes *protein1* the subject, *protein2* the object and *action* the relation. The verbs they investigate include *acetylate, activate, destabilise, inhibit, phosphorylate, suppress* and *target*. They simplify their task by assuming that all protein names are already known and present no quantitative results.

5.2 Comments

Sekimizu et. al. only report precision results and these are broadly comparable with those reported here. Their system uses standard linguistic processing but otherwise has been specifically developed for this domain. Blaschke's approach is simple and gives some interesting results, but without recall and precision figures it is difficult to compare to any other approach. It is obvious, however, that it will not be able to easily cope with, for example, parenthetical commas, relative clauses and so on which distance a subject or object from a verb. Also, it has a closed list of protein names which will inevitably lead to missed relations (i.e. false negatives.)

Of our results, the TOP1 and TOP2 are most interesting initially. They show that we have a high precision on those relations which we are trying to capture. The reason for this is that we deliberately tried to extract as many templates as possible with as few patterns as possible and these patterns are very reliable. The ALL results presented above show that there are ways in which recall could be increased, these include patterns for sentences containing relative clauses (e.g *KyoT, which physically interacts with RBP-J*), appositives (e.g. *GCN4, an activator of HIS*) or plural objects (e.g. *some kind of interaction between these domains*).

The main causes of loss of precision are incorrect reference resolution and NP bracketing (where a compound NP is incorrectly tagged). The former is factored out by only considering the TOP2 and TOP1 scores but the latter is still a problem and accounts for almost all the loss of precision in the results presented above. Other areas where there is scope for improvement include:

- add protein identification: our current method of identifying proteins is unsophisticated and we cannot simply use a large list of protein names since it will become out of date as new names are invented. Fukuda et al.[3] provide an algorithm for spotting proteins in text and initial tests suggest this would improve the precision results for TOP2 (scores of 80 and 100) but not the more restrictive TOP1 (score of 100).

- extending and improving reference resolution: Reference resolution, e.g. finding a referent for the pronoun, *It*, in ***It** interacts with eIF-4A,* underperformed due to the fact that we have not yet developed a specific domain ontology.

The utility of the output might be improved by including further information, such as:

- distinguishing negative vs. positive examples: *GFR alpha-3, which **did not** bind GDNF directly*

- determining level of interaction: *chimpanzee Lp (a) exhibits **poor** lysine-specific interaction with fibrin*

- including conditions: *FREAC-2 was shown to interact **in vitro** with TBP and TFIIB*

- determining degree of confidence: *we **show that** the eukaryotic initiation factor (eIF-5A) associates with the TGase*

As next steps, we might link templates to the appropriate SwissProt entries, use profiling, statistical or visualisation tools on the output or incorporate information from other, structured, resources as well as free text.

The total amount of time spent on this project, including familiarisation with the existing Highlight system and testing was three person-months.

6 Summary

We have shown that we can take a general IE system and customise it to a biological domain in a relatively short time. Much of the customisation work involved coping with the idiosyncrasies of protein names (e.g. *eIF-4a, FTZF2/HK*). Surprisingly little vocabulary needed to be added due to accurate word guessing and the shallow nature of the syntactic processing. The resulting system provides a cost-effective way of populating a database of protein-protein interactions.

References

1. Christian Blaschke, Miguel A. Andrade, Christos Ouzounis, and Alfonso Valencia. Automatic extraction of biological information from scientific text: Protein-protein interactions. (International Conference on Intelligent Systems for Molecular Biology. Heidelberg, 1999 (In press))
2. R. Brent and R. L. Finley Jr. Understanding gene and allele function with two-hybrid methods. *Annu Rev Genet*, 31:663–704, 1997.
3. K. Fukuda, T. Tsunoda, A. Tamura, and T. Takagi. Towards information extraction: Identifying protein names from biological papers. In *Proceedings of the Pacific Symposium on Biocomputing*, pages 707–718, 1998.
4. Jerry R. Hobbs, Douglas Appelt, David Israel John Bear, Megumi Kameyama, Mark Stickel, and Mabry Tyson. Fastus: A cascaded finite-state transducer for extracting information from natural-language text. In E. Roche and Y. Schabes, editors, *Finite State Devices for Natural Language Processing*. MIT Press, 1996.
5. P. D. Karp, M. Krummenacker, S. Paley, and J. Wagg. Integrated pathway-genome databases and their role in drug discovery. *Trends Biotechnol*, 17:275–281, 1999.
6. D. S. McNabb and L. Guarente. Genetic and biological probes for protein-protein interactions. *Curr Opin Biotechnol*, 7:554–559, 1996.
7. E. M. Phizicky and S. Fields. Protein-protein interactions: methods for detection and analysis. *Microbiol Rev*, 59:94–123, 1995.

8. T. Sekimizu, H. S. Park, and J. Tsujii. Identifying the interaction between genes and gene products based on frequently seen verbs in medline abstracts. In *Genome Informatics*. Universal Academy Press, 1998.