Extracting Biological Information from Full-length Papers
Technical report RN/03/17

David P.A. Corney[1], David T. Jones[1]*, Bernard F. Buxton[1], William B. Langdon[1], Joanne Charlwood[2] and Peter M. Woollard[3]

[1] Department of Computer Science, University College London, Gower Street, London, WC1E 6BT, UK
[2] GlaxoSmithKline, Park Road, Ware, Hertfordshire, SG12 ODP, UK
[3] GlaxoSmithKline, Gunnels Wood Road, Stevenage, Hertfordshire, UK, SG1 2NY, UK

Abstract

Converting the vast quantity of free-format text found in journals into a concise, structured database makes the researcher's quest for information easier. Recently, several information extraction systems have been developed that attempt to simplify the retrieval and analysis of biological and medical data. Most of this work has used the abstract alone, owing to the convenience of access and the quality of data. Abstracts are generally available through central collections with easy direct access (e.g. PubMed). The full-text papers contain more information, but are distributed across many locations (e.g. publishers' web sites, journal web sites, and local repositories), making access more difficult.

In this paper, we present BioRAT, a new information extraction (IE) tool, specifically designed to perform biomedical IE, and which is able to access and analyse both abstracts and full-length papers. BioRAT is a **Bio**logical **R**esearch **A**ssistant for **T**ext mining, and incorporates a document search ability with domain specific IE. We show first, that BioRAT performs as well as existing systems, when applied to abstracts; and second, that significantly more information is available to BioRAT through the full-length papers than via the abstracts alone. Typically, less than half of the available information is extracted from the abstract, with the majority coming from the body of each paper. Overall, BioRAT achieved 39% recall with 48% precision.

1 Introduction

The rapid and ongoing growth in the number of biological and medical publications means that researchers can no longer read more than a small proportion of the field. Yet interesting and useful information, relevant to the researcher, could appear in papers they have not read and therefore be missed entirely. Accompanying this growth in literature is the increasing proportion of electronically available papers, as many publishers now produce on-line versions of their journals. But while this may ease access, there is still a vast quantity that a researcher may feel they should read, with no concomitant increase in their ability to do so.

One way to ease the reader's difficulty is to consider this is as an "information filtering" problem, where we want to select a subset of the possible information available, for the researcher's inspection. One increasingly popular approach is to sort the documents into a number of pre-specified categories (Sebastiani 2002).

While this may help reduce the number of papers a researcher feels that they should read, it still leaves them with a large number of papers to read. Information extraction (IE) goes one stage further, and analyses the papers on behalf of the researcher. IE systems achieve this by identifying semantic structures in the text, and in so doing, distill an entire document down to the key facts.

BioRAT can be regarded as a research assistant that is given a query, and autonomously, finds a set of papers, reads them, and highlights the most relevant facts in each. BioRAT uses natural language processing techniques and domain-specific knowledge to search for patterns in documents, with the aim of identifying interesting facts. These facts can then be extracted to produce a database of information, which has a higher "information density" than a pile of papers. This is similar to an information extraction system (SUISEKI) that has recently been developed by Blaschke et al. (2001, 2002), and which will be discussed in more detail below.

*Corresponding author

There have been several attempts to apply IE techniques to scientific papers, but these have been restricted to use only the abstract of each paper. Example applications include protein-protein interactions (Thomas, Milward, Ouzounis, Pulman & Carroll 2000); using machine learning to classify biological relationships (Craven & Kumlien 1999); and protein structure and residues (Gaizauskas, Demetriou, Artymiuk & Willett 2003). In contrast, BioRAT uses the full length paper whenever it is available, instead of just the abstract.

Abstracts are readily available in large numbers (e.g. through PubMed[1]), are available in plain text, and typically have no superscript or subscript characters, no footnotes, and so on. This avoids potential difficulties in interpreting unusual symbols, Greek letters etc. However, the abstract is only ever a summary of the paper in question; the full text will typically include more detail that may be of direct interest to the reader. Therefore, in this work, we compare information extraction from abstracts and from full-length papers.

IE has been applied successfully to analysing financial publications, such as the *Wall Street Journal* and other news reports (Cowie & Wilks 2000). The highly structured nature of articles in these fields simplifies the use of templates (described below), on which IE is often dependent. The different writing styles encountered academic journals, in comparison with, say, those found in newspapers may cause problems. For example, identifying company names from newspaper reports of mergers may be more straightforward than identifying biological concepts across a wide range of journals. Researchers write papers that are typically aimed at very specific audiences, and so these authors can assume considerable prior knowledge, unlike mainstream newspaper journalists. In general, the writers' aims may not correspond to the readers' needs, and almost certainly do not correspond to the requirements for ease of machine processing.

In the next section, we describe the BioRAT system, and its key components. We then discuss the documents that BioRAT processes, and the conversion of PDF papers to plain text. Next, we discuss two experiments which evaluate BioRAT using a public-domain database (DIP), and discuss the results. Blaschke & Valencia (2001) specifically recommends the use of DIP as a "realistic scenario for the comparison of IE systems". The report concludes with some example results, and a brief discussion.

2 BioRAT System Outline

Figure 1 shows the general outline of the BioRAT system. We designed BioRAT to give people with no IE experience a powerful tool to help them locate and analyse research papers. The two paths shown in the diagram highlight how the full length paper can be analysed if it is available, or else the abstract alone can be processed. One feature of our design is that we include IE methods, so that the system extracts interesting and relevant facts from the documents it finds, instead of just presenting the user with a set of papers to read.

In a typical session, the user enters a query in the form of one or more keywords, with an optional restriction on publication dates. BioRAT then passes the query to PubMed, which returns a set of document identification numbers (PMIDs), and supplementary information, such as paper titles and author lists. Where available, PubMed also returns the URL of the full-length paper, typically on the web site of the corresponding journal. This summary is displayed to the user, allowing them to quickly skim through the titles of a number of papers, and view any abstracts they wish to. The paper's URL can be passed to the "spider" component of BioRAT, described below, which obtains the full length version of the paper.

The "Entrez Programming Utilities"[2] provide a straightforward web-based interface to the PubMed database. For our purposes, this consists of two tools: the first returns a list of PMIDs that match a given query; the second returns information relating to a particular PMID, such as the authors' names and the URL of the paper.

2.1 Web Spidering

One distinctive feature of BioRAT is that it automatically obtains full length papers wherever possible, instead of just using abstracts. It does this via the Internet, by following a series of hyperlinks to find each target paper. To find a particular paper, BioRAT starts with a URL (web address) provided by the PubMed database. It then goes to that web page and identifies the hyperlinks there, and recursively

[1] `http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=PubMed`

[2] `http://eutils.ncbi.nlm.nih.gov/entrez/query/static/eutils_help.html`

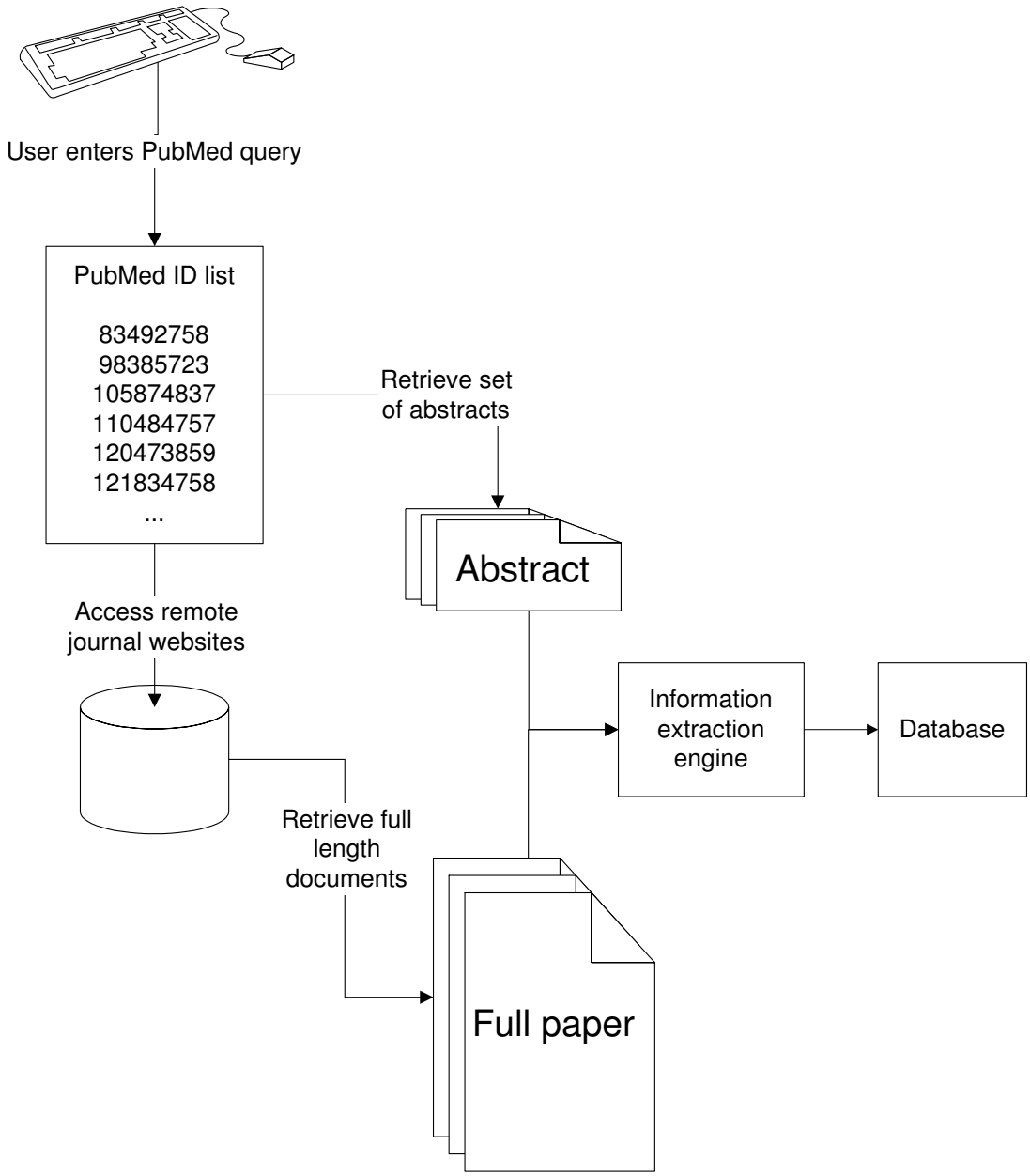

Figure 1: BioRAT system outline. BioRAT can be used to access abstracts or full-length papers, and to extract information from them.

follows links until it finds the target paper, in PDF format. This is downloaded and converted to a text-only version, ready for the IE engine.

Finding the target paper is non-trivial for such a tool. The URL provided by PubMed (and ultimately, by the journal publishers), typically points to a web page that contains a summary of the paper (e.g. abstract and authors), with hyperlinks to the full-length paper as a PDF file. However, this link is usually just one of many links within the page, with other links pointing to different papers in the same volume; to the references contained in the target paper; to copyright information; to publishers' homepages; and so on. The spider's task is to find the correct link, and follow it to the target paper, with the minimum of steps.

The system works by first downloading the web page as an HTML document to a local temporary cache. The HTML is then parsed to extract all the hyperlinks, along with corresponding screen text. Each link is evaluated by a simple string matching routine, that gives high scores to links containing words or phrases such as "full text" or "PDF", and low scores to links pointing to "privacy" or "e-mail alert". Where a link is not text but is a button, then the filename of the graphic being displayed is used instead. Often such button graphics have names such as "full_link.gif".

In the current version of BioRAT, keywords are used to evaluate the links. A pre-defined list of keywords has been chosen by hand, and each word in the list has a score associated with it, which can be positive or negative. Each link is given a score equal to the sum of the scores of each keyword contained within the link. For example, a link labelled "Download full length paper as PDF" might score 50 points for "full length", 30 for "PDF" and 10 for "paper", while a link labelled "Help" might score -100. In a similar web search tool, Rennie & McCallum (1999) used reinforcement learning as part of a document classification and retrieval system. A similar learning scheme could be applied here to adjust the weights of the keywords, and to select new keywords.

Having evaluated every link on the web page, the highest-scoring link is followed, and the corresponding page is downloaded. The process continues iteratively, until either the target paper is located, or a threshold number of links followed is passed, representing a failure. For example, if after following 10 links the target paper has not been found, the system stops looking. This failure threshold is not critical, and the value here has been chosen after some experimentation. If the spider follows the wrong link, then it may download the wrong paper. To check for this, the abstract is downloaded from PubMed, and a simple fuzzy string matching routine is called to compare the abstract with the first part of the document.

Ideally, the same sequence of words will appear in the PubMed abstract, and the start of the downloaded document. However, some PDF journal articles contain the end of the previous journal article at the start of the file. To further complicate matters, the PDF file may contain extra words that do not appear in the PubMed abstract, such as page numbers, copyright information and author addresses. These must be ignored by the matcher, or else the correct PDF article may be wrongly rejected. To allow for these problems, the fuzzy string matcher searches from the beginning of the PDF file until it finds the first word of the abstract. It then looks forward from that point for the second word of the abstract, and so on. If it finds most of the words from the PubMed abstract in the correct order, then the PDF file is accepted as being the correct paper. Otherwise, it is rejected. Several parameters control the detail of the fuzzy matching, such as how far ahead to search for each word, how far through the document to allow the search to continue, and what fraction of the abstract must be matched. These parameters can be set by the user; the parameters default values have been chosen by using a substantial test set (100 pairs of documents, of which 50 were correct abstract-PDF file matches, and the rest were false matches), and gave 100% correct performance on this sample.

A formal evaluation of the web spider component has yet to be completed. However, preliminary work suggests that the correct paper is located and downloaded approximately 70% of the time, from comparison of the number of times that the failure threshold was reached (i.e. that 10 links were followed), with the number of PubMed ID's produced by a sample query. The figure is obtained on the assumption that, if the full paper passes comparison with the original abstract obtained from PubMed, then the desired paper has been correctly located. It also assumes that the target paper is available electronically, and that any necessary journal subscription is in place.

Having obtained some relevant documents, the system then attempts to extract interesting facts from them.

2.2 Information extraction engine

Information extraction (IE) is a key part of BioRAT's functionality. The aim of IE is to extract from a set of documents the key facts about prespecified types of events, objects and relationships. These

facts are then used automatically to populate a database. This can then be used to ease on-line access.

The heart of BioRAT is an IE engine, based on the GATE toolbox[3], produced at Sheffield University (Cunningham, Maynard, Bontcheva & Tablan 2002). GATE is a general purpose text engineering system, whose modular and flexible design allows us to use it to create a more specialised biological IE system. GATE has been applied to numerous problems; those of most interest to us include automatic annotation and indexing of a digital library (Bontcheva, Maynard, Cunningham & Saggion 2002) and extracting enzyme interaction and protein structure information from journals (Humphreys, Demetriou & Gaizauskas 2000). Two components of GATE that must be modified for our domain-specific application are gazetteers and templates, which we shall now discuss in turn.

2.2.1 Gazetteers

One task in IE is "named entity recognition", which aims to identify key concepts within text. For example, we may want to identify words that are people's names, company names, proteins, genes and so on. Once identified, these words or phrases can then be matched by the templates. One simple approach is to use a gazetteer.

A gazetteer is a list of words identifying members of a particular category. For example, one gazetteer may list names of proteins, while another lists names of people. BioRAT incorporates gazetteers from three sources, namely MeSH[4], Swiss-Prot[5], and hand-made lists.

The top two levels of the MeSH hierarchy contain a total of approximately 120 entries, each of which was used to define a separate gazetteer. Each of the almost 22,000 entries in MeSH was extracted and added to the appropriate gazetteer(s). A larger number of more refined gazetteers could have been created, if lower branches of the MeSH hierarchy had been used. But if the subclasses had been used to create separate gazetteers, then we would gain precision at the cost of requiring more complex templates. The current compromise of using just the first two levels of MeSH produces satisfactory results with modest overheads.

Besides MeSH, further gazetteers were derived from Swiss-Prot. Each entry from Swiss-Prot describes a single protein, but proteins often have many synonyms, all of which are included in the relevant gazetteer. Also, some authors refer to proteins in terms of the genes that encode them, so the gene names were also extracted, and used to create another gazetteer.

To supplement these two sources, further gazetteers were created by hand by the authors. These consisted of words or phrases that covered concepts of interest that were not already in other gazetteers, such as "words describing the interaction of proteins" (e.g. "bind", "down-regulate", "interact" and so on).

This current set of gazetteers could be easily extended, perhaps by incorporating GO[6] or UMLS[7] terms.

2.2.2 Templates

A template is a representation of a text pattern that allows us to automatically extract information. It consists of a number of predefined slots to be filled in by the system from information contained in the text. One of the simplest templates from BioRAT is:

"interaction of" (PROTEIN_1) "and" (PROTEIN_2)

Here, "PROTEIN_1" and "PROTEIN_2" are slots to be filled with names of proteins, as defined by a gazetteer. The contextual phrase ("interaction of") is a fixed string: only phrases containing those exact words will be matched by this particular template.

A rather more complicated template is:

(EXPRESSION) "of" (WORD)? (WORD)? (WORD)?
(PROTEIN_1) (WORD)? (WORD)? (WORD)?
("by" | "to"| "with")
(WORD)? (WORD)? (WORD)?
(PROTEIN_2) "and" (PROTEIN_3)

[3]General Architecture for Text Engineering – `http://gate.ac.uk/`
[4]Medical Subject Hierarchy — `http://www.nlm.nih.gov/mesh/`
[5]`http://www.expasy.org/`
[6]Gene Ontology — `http://www.geneontology.org/`
[7]Unified Medical Language System — `http://www.nlm.nih.gov/research/umls/`

```
<resultsList>
   <templateResult>
     <filename>/DIP/papers/ 10087260.txt</filename>
     <context>Genetic evidence for the interaction of Pex7p and Pex13p is
provided by the observation that overexpression of Pex13p suppresses a loss
of function mutant of Pex7p.</context>
     <rule>DIPinteraction1</rule>
     <protein1>Pex13p</protein1>
     <protein2>Pex7p</protein2>
   </templateResult>
   <templateResult>
       ⋮
   </templateResult>
</resultsList>
```

Figure 2: Extract from a typical XML output file produced by BioRAT

Here, "EXPRESSION" refers to a gazetteer containing words relating to protein expression and interaction, such as "bind" and "inhibit". The slot (WORD)? is a wildcard that matches any word, but is optional, so the sequence (WORD)? (WORD)? (WORD)? matches between zero and three consecutive words of any type. As before, the three (PROTEIN_x) slots match protein names, and the quote strings must be matched exactly. The | character is a logical OR. For example, this template matches part of the sentence "Specific binding of Rna15 in complex with Hrp1 and Rna14 creates a polymerase pause site at the sixth nucleotide of the A-rich element", and identifies two interactions: (Rna15 $\leftrightarrow$ Hrp1) and (Rna15 $\leftrightarrow$ Rna14), with the expression type "binding".

As with comparable IE systems, the templates in BioRAT are written by hand. Although there have been attempts at automatic template creation (Collier 1998), these have not been broadly applicable. Although template design takes time and requires some practice, it does allow the user to maintain full control over what information is extracted, and allows experts to incorporate their knowledge within the system.

The first of the two templates defined above was used to generate the example XML record in Figure 2. Such XML files can be readily imported into existing database query systems. The same data is produced simultaneously as HTML and as a comma-separated list, for viewing in applications such as a web-browser or a spreadsheet, if that is more convenient for the user. Each record in the resulting database represents a single completed template.

2.3 Comparison with SUISEKI

The SUISEKI system was developed by Blaschke & Valencia (2001, 2002), and, like BioRAT, it uses gazetteers derived from Swiss-Prot (and DIP itself) to identify protein names. It uses "frames" to extract information. Frames are similar to BioRAT's templates in that they define patterns of language that form the basis for IE. However, they use less sophisticated linguistic knowledge, and more use of statistics. For example, frames distinguish between nouns and verbs, but do not recognise conjunctions, adjectives or any other parts of speech. Also, they count the number of words occurring in a phrase, and aim to find patterns in short phrases rather than longer ones.

One issue that arose during the development of BioRAT was that many protein (and gene) names are easily mistaken for common words. For example, the Swiss-Prot database includes entries with names "mice", "was" and "alpha", as well as 26 single-letter gene names. The problem is then to distinguish whether the word "was" refers to a gene or is simply the past tense of the verb "to be". Sometimes, this can be resolved by considering the case of the letters, but this is not reliable. Instead, BioRAT applies a filter based on the part of speech of the word, and rejects determinants, conjunctions etc. as not being proteins. This provides one possible advantage of BioRAT over SUISEKI.

3 Source Documents

PubMed contains abstracts of more than 14 million papers, and is continuing to grow rapidly. Increasingly over the last 5–10 years, the papers cited in PubMed have also been made available electronically, with hyperlinks from PubMed to the publishers' web sites. Similar databases include ChemWeb[8] for chemistry publications; arXiv[9] for physics and maths publications; and CiteSeer[10] for computer science. Alternatively, a local store of documents can be used by BioRAT.

Most past work has been restricted to abstracts, such as SUISEKI (Blaschke & Valencia 2002) and PASTA (Gaizauskas et al. 2003). As noted in Section 1, abstracts are generally available in plain text format, which makes things easy, as no translation is required, there are no unusual symbols or formulae, no tables or figures etc. The full text of a paper is typically available in PDF format, which embeds figures and tables, with varying fonts, line breaks, super- or sub-scripts, Greek letters and other symbols etc. All of these get lost or corrupted when PDF is converted to the more limited ASCII text. IE systems generally work on plain text, because they are processing the natural language. Some papers are also available in HTML, but this need not be any easier: symbols may be included as graphics files (e.g. GIF format); different sections of the paper may appear on different web pages; and extra hyperlinks may be inserted (e.g. to footnotes, to reference lists, to tables, or to other papers). These are designed to aid human navigation, but may confuse robotic textual analysis.

Many journal articles are published in a two-column format, and this extends to electronic versions of the papers. Some PDF-to-text conversion tools read right across the page, and so conflate columns. This may cause problems to linguistic analysis of the resultant text. A further complication is that the title and abstract may be in a one column format, even if the rest of the paper is in two columns. Most generic conversion tools either read across columns, or else attempt to maintain the page layout. In either case, some post-processing is required in order to recognise column and page breaks, and so to produce a text-only document with all the words in the correct order. However, Adobe Acrobat (version 5.05 and later) includes a manual PDF-to-text conversion facility that deals with columns correctly, reading down the left-hand column of one page, then the right-hand column of the same page, before moving on to the next page. Note that we assume the paper is written in English.

In some cases, the PDF-to-text conversion failed completely. This occurs when the source document was created by scanning in a paper, as opposed to using conventional PDF authoring tools to generate a file from text. In effect, the document contains a picture of some words, rather than the words themselves. One approach to process these papers would be to use optical character recognition (OCR), but we have not attempted this to date. In our experience, less than 5% of PDF files downloaded via BioRAT have failed to convert to plain text for this reason. OCR could also be used to scan in older publications, which may never have been made available electronically. Examples include the British Library's "Collect Britain" project[11] and the Royal Society's archive[12].

4 Using DIP for recall experiments

Having described the BioRAT system, and considered the documents that it can be used to analyse, we now turn to a particular study to test the usefulness of the system. For this, we will use the Database of Interacting Proteins (DIP[13]; (Xenarios, Salwinski, Duan, Higney, Kim & Eisenberg 2002)). Blaschke & Valencia (2001) recommends using DIP as a way of evaluating biological IE systems, because it represents a realistic problem of practical interest to biological researchers. IE researchers can use their systems to extract protein-protein interactions, and then compare these with the known records in DIP. By re-creating (part of) DIP, IE researchers can calculate the recall of their systems, and compare these results between different systems. The recall (or "sensitivity") is the fraction of records that the IE system correctly re-creates.

Each record in DIP defines a pair of proteins that interact with each other, and provides citations of papers that describe the interaction. Proteins are defined by entry keys to Swiss-Prot, GenBank or PIR. For simplicity, we only consider DIP records containing two Swiss-Prot identifiers. The DIP records also define the experimental method used (e.g. "two hybrid test" or "immunoprecipitation"),

[8] `http://www.chemweb.com/`
[9] `http://arxiv.org/`
[10] `http://citeseer.nj.nec.com/cs`
[11] `http://www.collectbritain.co.uk/`
[12] `http://www.royalsoc.ac.uk/dservea`
[13] `http://dip.doe-mbi.ucla.edu`

and whether the experimental scale was "small scale" or "genome wide". We use this information when selected records for our experiments, as we describe below.

Some papers describe just one interaction; others describe thousands. For example, one paper[14] has 4425 entries in DIP. However, when a paper has such a large number of entries in DIP, this usually means that the paper describes a large scale study, whose detailed results are *not* quoted in the paper, but are perhaps available in a supplementary database or web site. Obviously, IE systems can only extract information that is present, so we will exclude papers that have more than 5 entries in DIP. In this way, we can be reasonably sure that the papers do explicitly describe the interactions.

We are further limited to articles that are available electronically. For example, this excludes most papers that were published before the mid-1990s, when most journals were paper-only. Also, the experiments described here were carried out using computers at UCL, and so we could only access journals to which UCL subscribes, and those which are freely available.

For each experiment, we start by selecting a subset of DIP. BioRAT can analyse papers rapidly, typically taking just a few seconds to complete it's analysis a full-length paper. However, for our experiments, the results need to be manually checked in order to calculate the recall rate, and this time-consuming task forces us to limit the targets to a manageable subset of DIP. We then select the corresponding papers (abstract and full-text) mentioned in the selected DIP records. We process these papers using BioRAT, and manually compare the output of BioRAT to the source DIP records to measure the recall. For each record in DIP, we search through the output of BioRAT corresponding to the same paper, and check to see if the interaction mentioned in DIP has been identified. Throughout this work, we use the January 2003 version ("dip20030105") of DIP, the March 2003 version of Swiss-Prot, and the 2003 edition of MeSH.

5 Experiment 1 — Comparison with SUISEKI

In this section, we compare BioRAT with the existing SUISEKI information extraction engine described by Blaschke & Valencia (2001, 2002). We compare the performance of BioRAT to that of their system by measuring the recall of BioRAT on a sample of papers from DIP that were also used by Blaschke & Valencia (2001). This provides a suitable benchmark for BioRAT.

There were 415 records from DIP, which meet the following conditions:

1. They were used by Blaschke & Valencia (2001);
2. The DIP record refers to two Swiss-Prot records; and
3. The paper describes small-scale studies, and mentions less than 5 protein-protein interactions.

These 415 DIP records relate to 273 PubMed citations. We randomly selected 150 of these DIP records, covering 99 abstracts, to analyse the BioRAT results by hand.

We used a total of 19 hand-built templates, initially based on the SUISEKI frames, but subsequently modified; and 127 gazetteers, derived from MeSH and other sources, as described earlier. The templates and gazetteers used here can be accessed from the same website as the BioRAT software, i.e. `http://bioinf.cs.ucl.ac.uk/biorat`.

5.1 Results

Table 1 shows the recall from these abstracts by BioRAT, namely 19.33%. This may seem to be a low recall score, but is similar to that achieved by SUISEKI. Although this study is relatively small, the results are comparable with the larger study reported by Blaschke & Valencia (2002), where 190 DIP interactions were correctly detected, from a possible set of 851 interactions, giving a recall score of 22.33%.

We can compare the "abstract" results (Table 1) to the results in Blaschke & Valencia (2002), if we assume the results follow a binomial distribution. We want to test the hypothesis that both systems have the same recall.

Our recall of 19.33% from 150 trials gives a variance of $\sigma^2 = 150 \times 0.1933 \times (1 - 0.1933) = 23.39$ and hence a standard deviation of $\sigma = 4.837$. Blaschke et al. quote a recall of 190 cases from 851 trials, giving a recall rate of $190/851 = 0.2233$. If they had achieved the same rate on our smaller sample, we would expect them to achieve $150 \times 0.2233 = 33.49$ successes. This is within one standard deviation of our success score, so we can say that both systems are performing with approximately the same recall.

[14] PMID 11283351

	BioRAT		SUISEKI	
Result	Cases	Percent	Cases	Percent
Match	29	19.33	190	22.33
No match	121	80.67	661	77.67
Totals	150	100	851	100

Table 1: Comparison of BioRAT and SUISEKI on recall from abstracts. BioRAT results from 150 DIP records, derived from 99 abstracts. SUISEKI results from 851 DIP records, derived from 514 abstracts. The former set of records is a subset of the latter.

Result	Cases	Percent
Match in abstract	14	15.91
Match in full text (*but not in abstract*)	20	22.73
(Total match)	(34)	(38.64)
No match	54	61.36
Totals	88	100

Table 2: BioRAT recall results from 88 DIP records, derived from 60 full-length papers

In the next section, we compare information extraction from abstracts to that from full-length papers.

6 Experiment 2 — Abstracts vs. Full-length papers

In this experiment, we want to assess the benefits of using the full-length version of a paper, rather than just the abstract. The experiment follows the same pattern as that described above. This time, we take 88 records, based on 60 different documents, where full text and abstract are both available, and less than 5 interactions are cited for each paper. We no longer restrict ourselves to those records that (Blaschke & Valencia 2002) used. We used the same set of templates and gazetteers as above.

Table 2 shows the results. The information extraction rate obtained was over 38%, with more than half of the information coming from the body of the paper, and the rest from the abstract. This clearly shows the benefit for our system of analysing the full text of a paper, rather than restricting information extraction to just the abstract.

Using a similar binomial analysis to that used earlier, we can also test whether this improvement is significantly better than the information extracted from just the abstracts in the results obtained from SUISEKI, as reported in Blaschke & Valencia (2001).

The recall of 38.64% from 88 papers gives a standard deviation of $\sigma = 4.568$. Over 88 records, we would expect SUISEKI to recall 19.65 records, which is $\frac{34-19.65}{4.568} = 3.14$ standard deviations below the best estimate of our recall. I.e. BioRAT is significantly better.

Table 3 shows an analysis of the errors made by BioRAT in this second, larger experiment. This includes three records listed as "DIP entry erroneous", where the authors believe that the DIP record is inconsistent with the corresponding article, or at least, the article is ambiguous. For example, the DIP record DIP:455E describes a link between protein "ATPA_ECOLI" (SWP:P00822) and "ATPB_ECOLI" (SWP:P00824) in one particular paper[15]. The two Swiss-Prot entries referred to by DIP describe the proteins "ATP synthase alpha chain" and "ATP synthase beta chain" respectively. However, the paper describes an interaction involving ATP synthase, without specifying the alpha/beta chains. Whether this process is a protein-protein interaction in the conventional sense is a matter of some debate. In any case, it would be impossible for BioRAT (or any IE system) to recreate this DIP record, as the information is not contained in this paper, but rather it relies on implicit biological knowledge.

PDF-to-text conversion failed in two cases, whilst of the remaining 49 failures in Table 3, 15 were caused by failure to identify the correct proteins. Each protein is typically known by several different names, and may also be referred to by its associated gene, which itself may have several distinct names.

[15]PMID 10576729

Cause	Cases
Template insufficient	34
Protein identification failure	15
Text conversion problems	2
DIP Entry erroneous	3
Total	54

Table 3: Analysis of reason for failure by BioRAT during Experiment 2

Furthermore, long names may be abbreviated by the authors for parsimony, producing further non-standard ways of referring to the protein. The gazetteer used in these experiments included more than 230,000 gene names and more than 99,000 proteins, but still failed to recognise a large number of proteins.

One example of this protein identification failure comes from DIP record DIP:43E. The corresponding Swiss-Prot entry (P15172) refers to the protein "Myoblast determination protein 1", and lists synonyms "Myogenic factor 3" and "Myf-3", with gene names "MYOD1" and "MYF3". However, the paper in question[16] refers repeatedly to "MYOD". While this is clearly the same protein, a slightly different abbreviation has been used by the author than those included in Swiss-Prot. The gazetteer used by BioRAT is derived principally from Swiss-Prot, and so BioRAT failed to recognise this protein, and hence failed to extract the interaction.

Another example of the difficulties of protein identification comes from Swiss-Prot record P15260, which names the protein "Interferon-gamma receptor alpha chain", with gene name IFNGR1. One paper[17] refers to this protein as "IFN-gamma R alpha". As before, the gazetteer method used by BioRAT to recognise proteins fails, and so BioRAT fails to recreate the corresponding DIP record (DIP:728E).

The remaining failures listed in Table 3 are due to imperfections in the set of templates used by BioRAT. Although these errors could no doubt be reduced by improving the templates, there is no clear way to achieve this without a significant manual effort. Thus template design remains a major issue in information extraction research (Cowie & Wilks 2000).

Even when BioRAT fails to extract the relevant information, it may still highlight the correct piece of text. For example, DIP record DIP:800E defines an interaction between proteins p53 and UBE2I. BioRAT failed to identify this interaction, but did extract the following sentence[18]:

> Since the tumor suppresser protein p53 and a newly identified ubiquitin-like protein (UBL1) are implicated in the RAD51/RAD52 complex. . . , we further tested their associations with UBE2I.

Note that BioRAT correctly identified the above sentence as defining the interaction between RAD51 and RAD52, even though it missed the target interaction.

7 Precision

Above, we used IE techniques to re-create DIP records, allowing us to measure and compare the recall of different IE systems, where recall is defined as the proportion of positive instances that were correctly identified. In contrast, precision is a measure of how many false positive predictions are made. It is defined as the number of correct positive predictions divided by the total number of positive predictions.

The recall (or "sensitivity") is the fraction of target records that the IE system correctly re-creates:

$$\text{Recall} = \frac{\#\text{true positives}}{\#\text{true positives} + \#\text{false negatives}} .$$

Precision is a measure of how much of the output of an IE system is correct, and is defined as the ratio of the number of *correct* positive predictions to the total number of positive predictions made:

$$\text{Precision} = \frac{\#\text{true positives}}{\#\text{true positives} + \#\text{false positives}} .$$

[16] PMID 9184158
[17] PMID 10860730
[18] PMID 8921390

Result	Cases	Percent
Correct extraction	121	48.4
Template mistake	39	15.6
Protein identification mistake	87	34.8
Text extraction mistake	3	1.20
Totals	250	100

Table 4: Precision analysis. Here, "correct extraction" refers to records where the interaction information was extracted correctly, regardless of whether that interaction is in DIP.

In our experiments, precision is harder to measure than recall, because we need an estimate of the false positive rate. If a record produced by BioRAT is not found in DIP, it could be that a) it is a false-positive example, reducing the precision of BioRAT; or b) the record is missing from DIP.

For the second experiment described earlier, BioRAT produced a total of 867 records, derived from the 60 papers it analysed. We randomly sampled 250 of these records for analysis by hand, with no reference to DIP. We counted how many of BioRAT's predictions were correctly extracted from the text, and what sort of mistakes it made. Table 4 shows the results. Almost half (48.4%) of all records produced by BioRAT are correct, in the sense that the information contained in the papers was correctly extracted, whether or not the information is in DIP. When information is incorrectly extracted, it is more likely to be caused by incorrectly identifying a protein than by incorrectly applying a template.

8 Example output

From one paper[19], BioRAT found the interaction (Swi6 ↔ Hrr25), which corresponds to the DIP entry DIP:250E. BioRAT quoted the following sentence:

> Swi6 was also phosphorylated by Hrr25 kinase immunoprecipitated from yeast extracts with a HA tag (Fig. 2B).

A similar, but slightly more complex template can recognise two interactions at once. The following sentence[20] correctly lead BioRAT to produce two records for the interactions (Pcf11 ↔ Rna14) and (Pcf11 ↔ Rna15).

> Since Pcf11 interacts simultaneously with Rna14 and Rna15, its role in vivo may also be to stabilize their interaction.

A less successful example comes from the following sentence:

> Many interactions between nucleoporins and nuclear transport receptors have already been identified; however, we were unable to detect a biochemical interaction between Cse1p and Nup2p.

BioRAT incorrectly predicted that Cse1p interacted with Nup2p, whereas the text is less conclusive.

Other sentences are very difficult to interpret without being fully aware of the context. This can cause problems for BioRAT, as shown in the following sentence, which led BioRAT to predict an interaction between Cmd1p and Spc29p.

> GST-Spc110p/Cmd1p interacts independently with either Spc29p-Flag or Spc42p-2PY.

9 Discussion

As expected, the density of "interesting" facts found in the abstract is much higher than in the full text. This is at least in part because full-length papers include background discussion, a description of the method, references and so on. While these are necessary to set the work in context, and to provide supporting evidence, that may not contain the kind of information that BioRAT is attempting to extract.

[19] PMID 9012827
[20] PMID 11689698

The total size of the files containing the 60 abstracts used in the second experiment above was 96 kilobytes, while the total size of the 60 full-length papers[21] was 2857 kilobytes. BioRAT found 14 records from the same abstracts, and a total of $14 + 20 = 34$ from the full length versions. Thus the full length papers are approximately 30 times larger than the abstracts, and contain only 2.5 times the information, as defined by BioRAT's rate of information extraction.

In earlier experiments, we used a different PDF-to-text conversion utility (`ps2text.ps` - part of GhostScript[22]). This utility does not deal sensibly with multi-column PDF files, but reads right across the page, and so produces rather garbled text files. Nonetheless, the recall was almost as good as that reported here. The reason is that the information extracted by BioRAT is very local, spread across only a few words in each case. Most of the templates will only match sentences where the two proteins are just 3–4 words apart, or at most, 6–8 words apart, and in a specific context.

However, even when BioRAT (or another IE system) fails to find a particular relationship, or finds a non-existent one, it is quite possible that it has found an interesting part of an interesting document. In this way, using IE to guide a literature search is perfectly feasible, even if the recall and precision are a long way from the ideal 100%.

The results that BioRAT produces can be stored and retrieved using a variety of interfaces, easing the user's access to information. Furthermore, BioRAT also provides quotes from the source texts, and links directly to the source papers and related databases. In this way, BioRAT behaves like a virtual research assistant, guiding the user towards interesting papers.

10 Conclusions

We have presented BioRAT, an information extraction system specially designed to process biological research papers. A distinguishing feature of BioRAT is that is uses full-length papers, rather than being limited to abstracts as previous studies have been. The recall and precision performance of BioRAT was assessed by use of the DIP database of protein-protein interactions, and the recall was compared with that of a previous system, SUISEKI, which processed only the abstracts. The recall performance of BioRAT on the abstracts alone was similar to that of SUISEKI. Overall, BioRAT achieved 39% recall and 48% precision, showing the benefits of using full-length papers when performing information extraction. Extra time is required to obtain the full-length papers, and there are difficulties in converting them into a usable plain text format; but these costs are outweighed by the fact that more than twice as much relevant information can then be extracted automatically.

Acknowledgments

This work was funded by GlaxoSmithKline.

References

Blaschke, C. & Valencia, A. (2001), 'Can bibliographic pointers for known biological data be found automatically? Protein interactions as a case study', *Comparative and Functional Genomics* **2**, 196–206.

Blaschke, C. & Valencia, A. (2002), 'The frame-based module of the SUISEKI information extraction system', *IEEE Intelligent Systems* **17**, 14–20.

Bontcheva, K., Maynard, D., Cunningham, H. & Saggion, H. (2002), Using human language technology for automatic annotation and indexing of digital library content, *in* 'Proceedings of the 6th European Conference on Research and Advanced Technology for Digital Libraries (ECDL'2002)', Rome, Italy.

Collier, R. (1998), Automatic template creation for information extraction, PhD thesis, Department of Computer Science, University of Sheffield.

Cowie, J. & Wilks, Y. (2000), Information extraction, *in* R. Dale, H. Moisl & H. Somers, eds, 'Handbook of Natural Language Processing', Marcel Dekker, New York.

[21] after converting to text format

[22] `http://www.cs.wisc.edu/~ghost/`

Craven, M. & Kumlien, J. (1999), Constructing biological knowledge-bases by extracting information from text sources, *in* 'Proceedings of the Seventh International Conference on Intelligent Systems for Molecular Biology', Germany, pp. 77–86.

Cunningham, H., Maynard, D., Bontcheva, K. & Tablan, V. (2002), GATE: A framework and graphical development environment for robust NLP tools and applications, *in* 'Proceedings of the 40th Anniversary Meeting of the Association for Computational Linguistics (ACL'02)', Philadelphia, USA.

Gaizauskas, R., Demetriou, G., Artymiuk, P. & Willett, P. (2003), 'Protein structures and information extraction from biological texts: The PASTA system', *Journal of Bioinformatics* **19**(1), 135–143.

Humphreys, K., Demetriou, G. & Gaizauskas, R. (2000), Two applications of information extraction to biological science journal articles: Enzyme interactions and protein structures, *in* 'Proceedings of the Workshop on Natural Language Processing for Biology, held at the Pacific Symposium on Biocomputing (PSB2000), 2000'.

Rennie, J. & McCallum, A. K. (1999), Using reinforcement learning to spider the Web efficiently, *in* I. Bratko & S. Dzeroski, eds, 'Proceedings of ICML-99, 16th International Conference on Machine Learning', Morgan Kaufmann Publishers, San Francisco, USA, Bled, Slovenia, pp. 335–343.

Sebastiani, F. (2002), 'Machine learning in automated text categorization', *ACM Computing Surveys* **34**(1), 1–47.

Thomas, J., Milward, D., Ouzounis, C., Pulman, S. & Carroll, M. (2000), Automatic extraction of protein interactions from scientific abstracts, *in* 'Pacific Symposium on Biocomputing 5', pp. 538–549.

Xenarios, I., Salwinski, L., Duan, X., Higney, P., Kim, S. & Eisenberg, D. (2002), 'DIP: The database of interacting proteins. a research tool for studying cellular networks of protein interactions', *Nucleic Acids Research* **30**(1), 303–305.