Integrating Text Mining into Distributed Bioinformatics Workflows: A Web Services Implementation

Rob Gaizauskas Neil Davis George Demetriou Yikun Guo Ian Roberts
Department of Computer Science
University of Sheffield, UK
{*R.Gaizauskas, N.Davis, G.Demetriou, G.Yikun, I.Roberts*}*@dcs.shef.ac.uk*

Abstract

Workflows *are useful ways to support scientific researchers in carrying out repetitive analytical tasks on digital information.* Web services *can provide a useful implementation mechanism for workflows, particularly when they are distributed, i.e., where some of the data or processing resources are remote from the scientist initiating the workflow. While many scientific workflows primarily involve operations on structured or numerical data, all interpretation of results is done in the context of related work in the field, as reflected in the scientific literature.* Text mining *technology can assist in automatically building helpful pathways into the relevant literature as part of a workflow in order to support the scientific discovery process. In this paper we demonstrate how these three technologies – workflows, text mining, and web services – can be fruitfully combined in order to support bioinformatics researchers investigating the genetic basis of two physiological disorders – Graves' disease and Williams syndrome.*

1. Introduction

1.1. Workflows and Bioinformatics

Workflow management systems [6] are useful computational models not only for business processes, for which they have been primarily developed, but also for scientific research when it requires the repeated execution of a series of complex analytical tasks that each involve computation. For example, a biologist using micro-array techniques to uncover the genetic basis of a disease will repeatedly map the site of a reactive spot in the micro-array output – signifying a potentially significant difference in genes expressed in healthy versus diseased persons – to its gene sequence, use a sequence alignment tool, such as BLAST [2], to find protein or DNA sequences of similar structutre ("homologues"), mine information about these homologues from a variety of sources, typically remote databases or archives on the web, and annotate the unknown sequence with information from these mined sources. Bundling these steps into a single workflow which can be repeatedly executed by a workflow enactment engine for each new set of inputs can lead to significant savings in time and labour. This is particularly true if individual steps in the workflow involve access to remote data or processing resources which require substantial expertise to exploit, for example determining the value of a specific field in a remote database or running an analytical process on a remote server.

1.2. Bioinformatics Workflows and Web Services

Web services may be defined as processing resources that are available on the Internet, use standardised messaging formats, such as XML, and enable communication between applications without being tied to a particular operating system or programming language [13]. Because they support interoperability between services on different platforms, Web services are predicted to revolutionise and dominate the communication infrastructure of distributed resources in the coming years. This is already proving to be especially useful in bioinformatics research where the typical cycle of literature search, experimental work and article publishing usually requires access to data which is: (1) heterogeneous in nature – database records, numerical results, natural language texts; (2) distributed across the internet in research institutions around the world; (3) made available on a variety of platforms and via non-uniform interfaces.

Already a number of workflow systems for bioinformatics researchers have been produced, both by academic and commercial organisations. As a partner in the myGrid project [12] we have carried out the work reported here using workflow tools developed within the project. These include: a language for specifying workflows (Scufl), a tool for designing workflows (Taverna), and a workflow enactment engine (Freefluo), as well as an integrated information

model designed to hold the results of running workflows [1]. While nothing in these tools commits one to any particular service delivery technology, existing distributed workflows developed using the tools have all been implemented via web services.

1.3. Text Mining and Web Services

Text Mining is a term which is currently being used to mean various things by various people. In its broadest sense it may be used to refer to any process of revealing information – regularities, patterns or trends – in textual data, and includes more established research areas such as information extraction (IE), information retrieval (IR), natural language processing (NLP), knowledge discovery from databases (KDD), and so on. In a narrower sense it requires the discovery of *new* information – not just the provision of access to information existing already in a text or to vague trends in text [5]. We shall use the term in its broadest sense in the following, as we believe that while the end goal may be the discovery of new information from text, the provision of services which accomplish more modest tasks, such as the recognition of entities (e.g. genes or proteins) in text, are important components for more sophisticated systems which may utilise these components in the pursuit of genuine discovery. These components are therefore part of the text mining enterprise.

Text mining is particularly relevant to bioinformatics applications, where the explosive growth of the biomedical literature over the last few years has made the process of searching for information in this literature an increasingly difficult task for biologists. Depending on the difficulty of the task, text mining systems may have to employ a range of text processing techniques, from simple information retrieval to sophisticated natural language analysis with the use of algorithms developed either manually or by machine-learning methods, some of which may be freely available. However, exploration of the potential of text mining systems by prospective technology integrators has so far been hindered by the non-standardised data representations, the diversity of processing resources across different platforms at different sites and the fact that linguistic expertise for developing or integrating natural language processing components is still not widely available. All this suggests that, in the current era of information sharing across networks, an approach based on Web services may be better suited to rapid system development and deployment.

Compared to the amount of research and development in text mining, Web services technologies are still relatively new, so it is not surprising that work on integrating text mining with Web services is still limited. Some examples of generic language components available as Web services can be found at the RemoteMethods website [10]. But perhaps closest to the work described in this paper is the proposal by [7] who describe text mining middleware for Medline based on Web services. The middleware is implemented as a two-layer architecture, with the first layer providing the API to the text mining components and the second layer providing the Web services. The text mining subsystem is based on the text analysis system by [8]. The middleware provides facilities such as identification of biomedical terms and extraction of relations based on noun-verb-noun sequences. Although there are similarities between our system and that of [7], such as the use of Medline as the data source, our system differs in that in our case the Web services are run from within a biomedical workflow on the grid rather than being tightly coupled to a specific piece of middleware. In this way we allow the external workflow to specify the arrangement of inputs and outputs of the individual Web services rather than use a pre-arranged graph of services in the middleware as [7] do. [9] have developed a set of Web services to support operations for machine translation and terminology management. These services include full and partial dictionary lookup and terminology extraction from bilingual corpora. Direct comparisons with the system we describe below cannot be made as both the domain and the target applications are different.

1.4. Putting it all Together

The foregoing has motivated the use of and shown the connections between workflow systems, text mining and web services, especially in the domain of bioinformatics. Exploiting these natural connections, we have been developing text mining services to support biomedical researchers as part of the myGrid and CLEF e-Science projects [4]. An on-going issue has been how to integrate these services into the workflow model. Most text mining functionality, while it may process text off-line in batch mode, is designed to support interactive operations, such as querying, searching and browsing text collections. The workflow model adopted in myGrid, however, envisages workflows as sequences operations to be run start to finish, with user browsing of results only at the end. In this model, what are the inputs to text mining to be? One answer is that the inputs should come directly from the user, i.e. that text mining should simply support *ad hoc* querying by the user in conjunction with whatever workflow he/she is running. This effectively denies the utility or possibility of tighter integration. An alternative is to explore ways in which an implicit query can be inferred or synthesized from information in a workflow.

In this paper we describe how for two biological research scenarios that are being used to drive myGrid development we have identified points in the workflow where links to the scientific literature can be used to deliver to the biologist a

variety of textual information that may assist in interpreting the results of the workflow. The services are still under-active development with the aim of offerring considerably richer text mining capability than currently available. However we believe that the model that we describe below in section 3 for delivering web services-based text mining services as part of a bioinformatic workflow is sound and will remain at the core of any extended text mining capability.

2. Two Case Studies in the Genetic Basis of Disease

In order to develop and to validate our approach to integrating text mining services into bioinformatics workflows we have chosen to concentrate on two case studies, both of which involve biological researchers investigating the genetic basis of human disorders. These examples are highly typical of many other similar investigations. We describe them briefly here to provide context for the discussion that follows.

2.1. The Graves' Disease Scenario

Grave's Disease is a an autoimmune condition primarily affecting tissues in the thyroid and orbit, giving rise to physical symptoms of an enlarged thyroid gland and protruding eyes, along with agitation from high levels of thyroid hormones. The exact cause of Grave's disease is unknown and though there is genetic predisposition, the genotype of the disease and its interaction with environmental stressors is not well understood. The genetic factors implicated in the disease are being investigated using the sort of micro-array methods described above in section 1.1. The key step at which text mining can play a role follows the BLAST search – BLAST reports contain, amongst other things, references to records for homologous proteins in the SWISSPROT protein database and these records in turn contain the ids of abstracts describing these proteins in the Medline abstract database. These abstracts can be mined directly for information of interest to the biologist, or can be used as "seed" documents to assemble a set of related abstracts from which clues about possible gene/protein function can be gleaned.

2.2. The Williams Syndrome Scenario

Williams Syndrome is a congenital disorder resulting in mild to moderate mental retardation. The syndrome is caused by the deletion of genetic material on the seventh chromosome. The area in which these deletions occur is not currently well characterised. Sequence information for the area of interest is becoming available, but must be organised into a contiguous sequence and genes isolated. The workflow developed to support Williams syndrome researchers involves running gene finding software over new sequence information as it becomes available then performing BLAST searches against new putative genes, again to identify homologues whose function may be known, yielding insights into the function of the gene of interest. As with the Graves' disease scenario the BLAST reports provide links to abstracts in the literature, in this case via links to the GenBank database, which can effectively be treated as queries to begin a search for relevant related information.

The Williams-Beuren syndrome workflow is currently in use at the University of Manchester in active research. The time taken for data searches is being cut from (typically) two weeks to approximately two hours, along with a reduction in manual errors. In addition to the reduction in time taken to search for data and resources, and the increase in the precision of results, the workflow model is proving itself to be simple to use and adapt by personnel without a computer science background. This has not been the case when using bespoke scripts.

3. Text Services

3.1. Architecture

In figure 1 is a simplified illustration of a setup in which text mining has been integrated into a distributed workflow environment. The setup comprises the following components: (1) a client from which a user launches a workflow and browses results (2) a workflow server that enacts the workflow itself consisting of any number of computational steps, each of which may involve accessing remote resources, including remote process invocation (3) a text database server holding the results of prior text analysis and indexing and making these results available via web services to external callers.

We believe this three-way division of labour is a sensible way to deliver text mining capability in a distributed workflow environment, and will be applicable to a wide range of other problem areas. Providers of electronic text archives, such as Medline, will want to make their archives available via services interfaces. However, the amount of functionality they will be willing to provide will be limited, both because of the need to be efficient and because they cannot support a potentially unlimited number of specialised user requirements. Specialist workflow designers will want to add value to the basic services offered by archive providers in order to meet the needs of researchers in their organisation or community. End users, certainly naive end users, will want to execute pre-defined workflows, most likely using a familiar, light client such a browser.

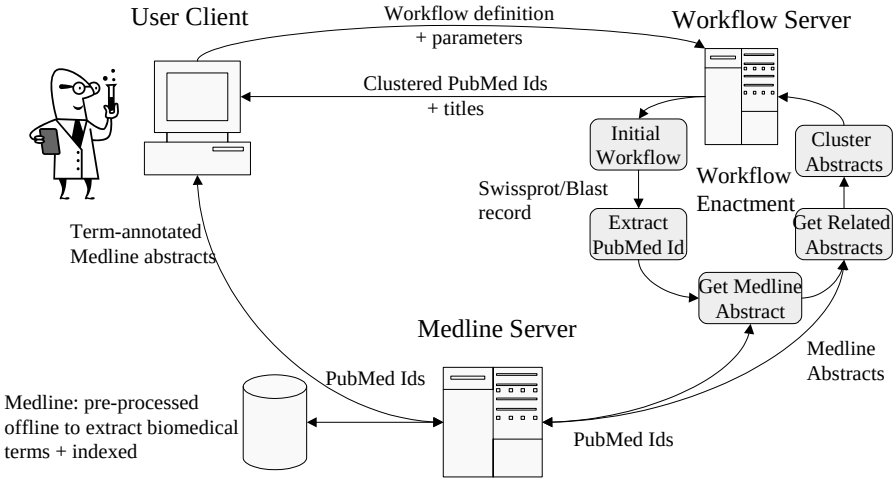

Figure 1. Text Services Architecture

3.2. Workflow Text Services

The text services architecture we have developed allows a user to invoke stored workflows which may contain zero or more text processing steps. For simplicity in figure 1 we show multiple initial steps in the Graves' disease or Williams syndrome workflows as a single component, called "initial workflow" in the diagram. This pseudo-component, which we have introduced solely to simplify the exposition here, encapsulates an arbitrary number of possibly very complex workflow steps. Following this step are a number of text processing steps, each of which has been implemented as a web-service. All workflow steps are run by the FreeFluo enactment engine when it executes the workflow. In the diagram, no workflow steps follow the text processing steps, but in the general case there is no reason why this needs to be so. Other workflow steps could follow on from the text processing steps, perhaps using outputs from text processing as inputs; or another sequence of steps could proceed in parallel, if there are no dependencies.

In our example scenarios, from the initial steps of a Grave's Disease or Williams syndrome workflow we get a SwissProt or BLAST report, respectively. A seed set of PubMed identifiers are retrieved using this report. For each of these identifiers the associated Medline record is retrieved. We want to use these records and the abstracts they contain to generate a set of relevant related abstracts. To do this a wide range of potentially useful techniques is available, including text clustering, automatic annotation of the abstracts with terms from domain ontologies, such as the Gene Ontology (GO), distillation from the abstracts of a query based on traditional IR mechanisms such as tf.idf term weighting [3], etc. One simple technique, which we have implemented to build the simple demonstrator reported here, is to use the keyword terms manually assigned to each Medline abstract by human indexers – the MeSH (Medical Subject Heading hierarchical controlled vocabulary) terms. These terms are retrieved and compiled into a list with any repeated terms removed. This list of MeSH terms is filtered to remove any that are considered to be non-discriminatory (e.g. the MeSH term "Mutation" is the major MeSH term for 42176 papers in the PubMed database and returning a list of this size for the user to look through would not be helpful.) The filtered MeSH term list is then used as a set of keys to pull back the PubMed identifiers of the related papers in the PubMed database where each member of the MeSH term list is a major MeSH term for that paper. This produces a list of associated papers which can be clustered in various ways or simply presented to the user. In our initial implementation we cluster the results according to the hierarchy in the MeSH ontology and present them graphically to the user as described in section 3.4.

In more detail, the Graves' Disease and Williams Syndrome workflows are constructed as a series of RPC style SOAP services. Prototypes and descriptions of these services are as follows (a diagram of the Graves' Disease workflow as constructed using the Taverna workflow editor is shown in figure 2):

1. ParseBLAST(*string*) : *string[]*

   (Williams syndrome only) The ParseBLAST service takes a single BLAST report as input, from which all the listed gene ID records are extracted. For each extracted gene ID a call is made to the National Library of Medicine (NLM) servers and the appropriate gene record is retrieved. Each gene record is parsed as it is retrieved and any PubMed identifiers (zero or more) are mined. The mined PubMed identifiers are com-

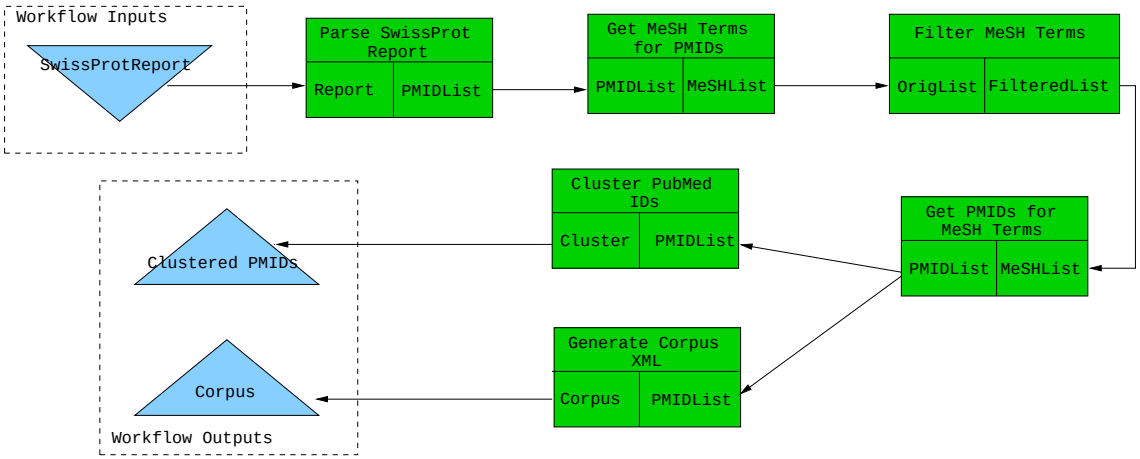

Figure 2. Graves' Disease Scenario Workflow

piled into a list with duplicates removed and returned to the caller.

2. ParseSwissProt(*string*) : *string[]*

   (Graves' Disease only) The ParseSwissProt service takes a single SwissProt record as input. All of the PubMed identifiers (zero or more) listed in the SwissProt record are mined and returned to the caller.

3. GetMeSH(*string[]*) : *string[]*

   The GetMeSH service takes a list of one or more PubMed identifiers and retrieves a list of the MeSH terms associated with the paper that each PubMed identifier refers to. The list of retrieved MeSH terms are compiled into a unique list and returned to the caller.

4. FilterMeSH(*string[]*) : *string[]*

   The FilterMesh service takes a list of one or more MeSH terms and removes from that list any MeSH terms that are deemed to be non-discriminatory. This is done by calculating the number of papers for which the MeSH term in question is a major topic header. If the number of papers falls above a pre-specified threshold the MeSH term is removed from the working list. The original list of MeSh terms with any non-discriminatory values removed is returned to the caller.

5. MeSHtoPMID(*string[]*) : *string[]*

   The MeshtoPMID service takes a list of one or more MeSH terms and retrieves the PubMed identifiers of all the papers that have the MeSH term in question as a major topic header. The list of retrieved PubMed identifiers is compiled into a list with duplicates removed and returned to the caller.

6. Cluster(*string[]*) : *string*

   The Cluster service takes a list of one or more PubMed identifiers and orders them using the MeSh tree as an organisational hierarchy. An XML representation of the MeSH tree with the supplied PubMed identifiers inserted into the appropriate nodes in the tree is returned to the caller as a string.

7. generateCorpus(*string[]*) : *string*

   The generateCorpus service takes a list of one or more PubMed identifiers and generates an XML representation of the information held in the pre-processed Medline database about the paper corresponding to each supplied PubMed identifier. These XML representations are returned to the caller as a string.

A demonstration workflow in the XSCUFL language can be downloaded from: `http://don.dcs.shef.ac.uk/mygrid-/demo/xscfl/graves_demo.scufl` and a sample input SwissProt record can be downloaded from: `http://don.dcs.shef.ac.uk/mygrid/demo-/input/SwissProt.demo`. To run the demonstrator the Taverna workbench and FreeFluo workflow enactment engine are also required and can be downloaded from `http://taverna.sourceforge.net/`.

3.3. Text Collection Server

As noted in section 3.1 there are good reasons for separating a set of services that provide general access to a text archive from services that may be developed as part of a workflow and use workflow context together with basic archive services to build bespoke text mining capability for workflow users. In the Graves' Disease and Williams

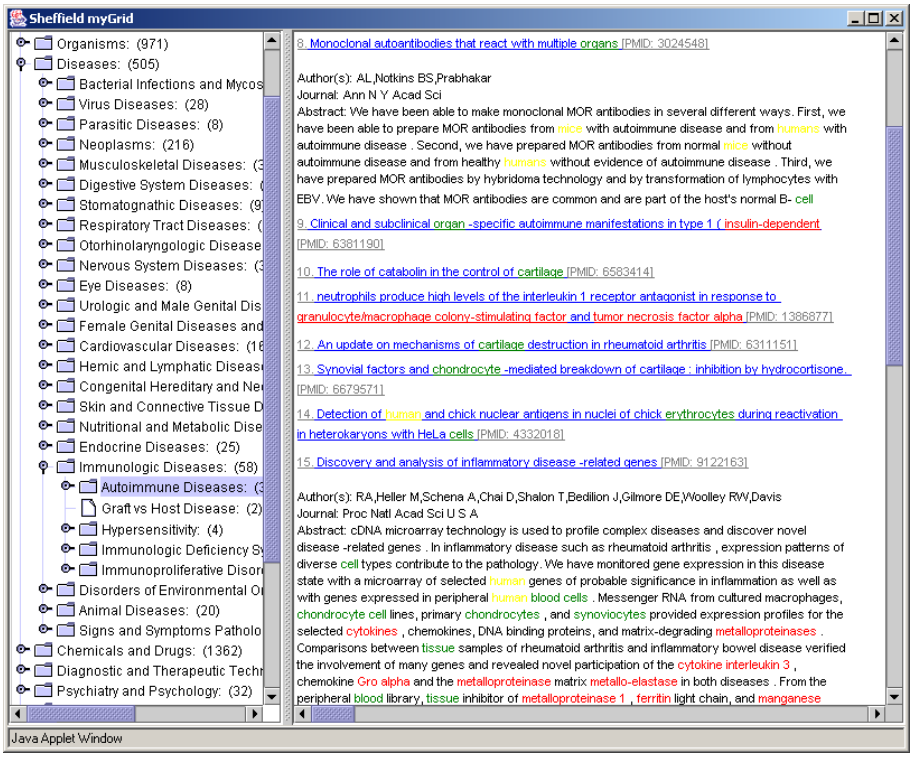

Figure 3. Text Services Interface

Syndrome scenarios the text database is all of Medline (15 million abstracts of biomedical journal papers). We have pre-processed selected portions of this database using the AMBIT system [4] to identify references to biological entities (genes, proteins, chromosomes, cells, organs, tissues, etc) and certain relations between entities (e.g. relations of containment, such as that a specific gene is found in a certain chromosome). Inverted indices have also been built to support access to all the extracted entities, and also to all the words, as in conventional free text search engines. These indices and the associated Medline abstracts have all been stored in a mySQL database to which a web-services interface has been added, enabling this data to be accessed by remote clients, such as a workflow server or a browser.

The fact that the back-end application is supported by a mySQL database ensures a high level of system scalability in terms of handling increased load due to multiple active clients. Time savings have been achieved because the fetching of a document is just a database operation in the local filesystem, rather than an Internet transaction as would be the case if the Pubmed website was queried in order to retrieve documents for annotation. The processing of Medline is the most “intelligent” part of the text mining technology in the current system, but since it is not the focus of the present paper we do not discuss it further here (see [4] for further details).

3.4. Interface

We provide a web-based front end to Freefluo, which allows users to enact predefined workflows without needing to understand the underlying workflow definition. The input data may be supplied by the user. To aid user understanding of the output information generated by the workflow we have also developed a graphical user interface (see Figure 3) which aims: (i) to map the information generated by the workflow operations, usually in XML or RDF format, into a textual representation that can be easily read and understood by the user, and (ii) to enable navigation through this information, supporting the user in knowledge exploration and discovery. Two issues in the development of the interface are platform-wide accessibility and efficient integration with the background text services. The former has been achieved by providing the interface as a Java Swing applet that can be invoked from a Web browser. For the latter, we have developed coordinated mechanisms between interface and workflow that allow for efficient access to text mining results without requiring the user to know or access the workings of the underlying components (specifically the functionality in step 7 of the workflow presented in section 3.2 has been added to the workflow so as to minimize delays experienced by the user while navigating the clustered, related document set; given the clustered document *ids* the

interface could have assembled the corpus on demand by calls to the archive server, but smoother browsing can be guaranteed by pre-assembling the results corpus as part of the workflow, which is expected to take fractionally more time in any case).

The interface offers both inter-and intra-document browsing facilities. Inter-document browsing is possible through the visualisation of hierarchical clustering of documents. The clustering is based on two kinds of features: (i) Medical Subject Headings (MESH), and (ii) biomedical terms extracted directly from the texts such as the names of genes, diseases, species etc. The tree-like visualisation of such clusters on the interface allows for easy navigation through the results of a particular experiment by expanding or collapsing term nodes that link to subsets of the retrieved document set. Each document subset is represented as a list of titles in a text pane, with each title being a hyperlink which, if selected, displays the Medline abstract for that particular document. The abstract is displayed with the annotated terms highlighted in various font colours with correspondence to the term classes of interest. We believe that this visual form provides value-added information compared to services such as Pubmed, as it should be easier for the user to quickly browse and judge the relevance of the text content.

In addition, we plan to provide term filtering functionality, in a way that will allow the user to further reduce a document cluster to a subset that includes only specific terms of interest such as, for example, 'alcohol dehydrogenase', whose selections are activated by check buttons on the interface. These filters act as labels for terms that have been extracted by the text mining back-end and they can be visualised in tabular or tree format. The availability of both document clustering and filtering methods and the combination of these methods in the same visual environment offers the user a powerful set of possibilities for information searching, browsing and navigation.

4. Conclusion and Future Work

We have described how we have integrated text mining services into a workflow environment designed to support scientific discovery. Our model consists of three components: (1) a text archive, possibly pre-processed using any number text analysis or mark-up tools, which is accessible via a web services interface; (2) a workflow enactment engine which executes workflows supplied to it, some of the workflow steps being text mining processes, themselves implemented as web services, which a) communicate with the archive server to retrieve texts, possibly pre-annotated, on the basis of an implicit query constructed from the information produced in earlier steps in the workflow and then b) manipulate these retrieved texts further; (3) a user interface client which initiates workflow execution and through which results are examined.

The text mining functionality we have delivered in the model so far is limited, but here the point is not the current state of this capability, but rather the utility of the general model. The text mining capability in the system is currently being extended in a number of directions. Other approaches to clustering texts in the expanded documents sets the are being investigated including term subsumption [11] and conventional agglomerative clustering and clustering based on assigning gene ontology (GO) codes. Algorithms for relation extraction over the Medline corpus are being developed so that relations such as *interacts with* or *inhibits* can be extracted and pre-stored at the archive server. Direct, *ad hoc* querying capabilities are being developed for the client via a query language that will support combined search over free text and extracted semantic information (entity types and relations) to give the user powerful querying capabilities over mined text. All of these capabilities will significantly extend the scope of text mining capability; but all will build on the model proposed here, which integrates workflows, web services and text mining to further support research biologists in the task of knowledge discovery.

Acknowledgements

This work has been supported by the UK Engineering and Physical Sciences Research Council via the myGrid e-Science project (Grant ref: GR/R67743) and the Medical Research Council via the CLEF e-Science project (Grant id: 60086).

References

[1] M. Addis, J. Ferris, M. Greenwood, D. Marvin, P. Li, T. Oinn, and A. Wipat. Experiences with escience workflow specification and enactment in bioinformatics. In S. Cox, editor, *Proceedings of UK e-Science All Hands Meeting 2003, Nottingham, UK*, 2003. http://www.nesc.ac.uk/events/ahm2003/AHMCD/.

[2] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *Journal of Mol. Biology*, 215:403–410, 1990.

[3] R. Baeza-Yates and B. Ribiero-Neto. *Modern Information Retrieval*. ACM Press Books, 1999.

[4] R. Gaizauskas, M. Hepple, N. Davis, Y. Guo, H. Harkema, A. Roberts, and I. Roberts. AMBIT: Acquiring medical and biological information from text. In S. Cox, editor, *Proceedings of UK e-Science All Hands Meeting 2003, Nottingham, UK*, 2003. http://www.nesc.ac.uk/events/ahm2003/AHMCD/.

[5] M. Hearst. Untangling text data mining. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL99)*, 1999.

[6] D. Hollingsworth. *The Workflow Reference Model*. Workflow Management Coalition, 1995. http://www.wfmc.org/standards/docs.htm.

[7] H. Kubo, N. Uramoto, S. Grell, and H. Matsuzawa. A Text Mining Middleware for Life Science. *Genome Informatics*, 13:574–575, 2002.

[8] T. Nasukawa and T. Nagano. Text analysis and knowledge mining system. *IBM Systems Journal*, 40(4):967–984, 2001.

[9] U. Quasthoff and C. Wolff. Web services in language technology and terminology management. In *Proceedings of the 6th Terminology in Advanced Management Applications Conference (TAMA-2003)*, South Africa, February 2003.

[10] RemoteMethods - Your Guide to Web Services. *Human Language Technology*. http://www.remotemethods.com/-home/valueman/converte/humanlan, 2004.

[11] M. Sanderson and B. Croft. Deriving concept hierarchies from text. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 206–213, Berkeley, CA, 1999.

[12] R. Stevens, A. Robinson, and C. Goble. mygrid: Personalised bioinformatics on the information grid. In *Proceedings of 11th International Conference on Intelligent Systems in Molecular Biology*, pages i302–i304, Brisbane, Australia, 2003. Published as Bioinformatics Vol. 19 Suppl. 1.

[13] World Wide Web Consortium - W3C. *Web Services Activity*. http://www.w3.org/2002/ws/, 2002.