Web Service Programming for Biological Text Mining

Moustafa Ghanem
Department of Computing
Imperial College London
180 Queens Gate London SW7 2AZ
+44 20 7594 8357
mmg@doc.ic.ac.uk

Alexandros Chortaras
Department of Computing
Imperial College London
180 Queens Gate London SW7 2AZ
+44 20 7594 8357
ac901@doc.ic.ac.uk

Yike Guo
Department of Computing
Imperial College London
180 Queens Gate London SW7 2AZ
+44 20 7594 8357
yg@doc.ic.ac.uk

ABSTRACT

In this paper we present a tool for conducting distributed text mining over bioinformatics data. Our approach is based on enabling non-programmers to construct visual workflows that coordinate the execution of distributed data and text mining components through a web services interface. We describe the motivation of our approach and the design of the tool and present examples of its application to the analysis of bioinformatics data.

Categories and Subject Descriptors

H.2.8. [Database Management]: Database Applications—data mining. **H.3.1 [Information Storage and Retrieval]:** Content Analysis and Indexing—Linguistic processing.

General Terms

Algorithms, Languages.

Keywords

Data Mining, Text Mining, Bioinformatics, Web Services, Workflow, Discovery Net.

1. INTRODUCTION

Within the data-analysis and life science community a large body of work has been devoted recently to investigating the use of text mining in extracting useful knowledge automatically from unstructured text sources. For example, a large number of studies aimed to identify and extract biological entities (genes, proteins, small molecules, chemical compounds, diseases, etc.) mentioned in the literature, e.g. [7,23,30], other studies aimed to extract the relationships between such entities (protein-protein interactions, gene-disease correlations, etc.), e.g. [1,19,30]. Furthermore, studies such as those reported in [22] aimed to investigate how text mining can be used to validate the results of analytical gene expression methods in identifying significant gene groupings. Also, recently a large number of shared international tasks have been defined in the area of text mining and bioinformatics. Such tasks include the TREC genomics track [32] that concentrates on the evaluation of information retrieval systems in the genomics domain, the BioLink and BioCreative initiatives [3] which concentrate on assessing and evaluating different methods for finding mentions of biological entities in text, and the KDD CUP 2002 Challenge [36] that concentrated on the analysis of a large collection of documents to identify documents with valid relationships between biological entities.

Our aim here is not to perform a comprehensive survey of such text mining efforts, but rather to highlight the wealth of tools that has been developed by various researchers recently to enable text mining over bioinformatics literature. We note that whereas there have been specific efforts dedicated towards collecting and annotating bioinformatics literature [3,21] to enable the definition and sharing of corpora and text bases, little effort has been devoted to enabling the interoperability between the wealth of tools that operate over them. Our motivation is thus to investigate how to enable the integration of these different tools within a unified framework, allowing their re-use in different contexts.

Our approach, which we detail in this paper, is based on the use of visual programming and web services technologies. Our computational framework is based on allowing users to carry out mixed data and text mining over distributed data and computational resources through a workflow co-ordination paradigm.

The contribution and novelty of work lies, not only in the individual algorithms implemented, but also at the system level by providing novel mechanisms allowing a flexible framework into which the text mining and data mining components can be integrated, and through allowing varying levels of abstraction and metadata in the system to allow the text mining components to be rapidly deployed in new areas, and to be used in conjunction with other data analysis and knowledge discovery tools.

2. VISUAL PROGRAMMING AND TEXT MINING

Our approach to text mining in bioinformatics is mainly based on enabling user definable processing workflows that mix text processing, natural language processing, statistical feature extraction operations and data mining operations. The utility of such workflow paradigm has been noted by a number of authors (see for example [10,25]), even if only as an abstraction, in a variety of text mining settings. Within most text mining applications, text

documents are first passed through a combination of text pre-processing operations that perform activities such as text cleaning, NLP parsing, regular expression operations, entity extraction operations, etc. This is typically then followed by the application of statistical operations on the results to represent the features of the documents in vector form, where counts are recorded for the keywords, patterns, gene and disease names, etc. identified in the previous stages. Finally, the vector form, which is amenable to numerical analysis, is fed to traditional data mining and machine learning techniques such as classification, clustering, dimensionality reduction and association analysis.

2.1 A Biological Document Classification Example

To highlight the use of such workflows, consider the task of bio-informatics document categorization based on the KDD CUP 2002 competition [36]. The task dealt with building automatic methods for detecting which scientific papers, in a set of full-text genetics papers from FlyBase [9], contained experimental results about gene products (transcripts and proteins), and also to identify and score, within each paper, which of the individual genes mentioned had experimental results about these products mentioned in the paper. A manually pre-labelled document collection of about 800 documents was made available by the task organizers together with a gene name dictionary to help contestants develop and train their automatic classification methods.

Figure 1 shows one possible workflow to develop and train an automatic classification system for the FlyBase task. This workflow is a variation on our early work presented in [11]. As with traditional document classification methods each document is represented as a feature vector that records how many times a particular feature occurs in a document (see Figure 2). However, as opposed to traditional approaches that use simple word tokens as the basic features to be counted, we base our features on the use of regular expression patterns. These patterns capture, within short segments of text, frequently co-occurring combinations of gene names and other general words that are mentioned within the text.

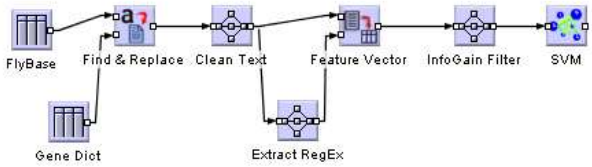

Figure 1: Document Classification Training Workflow: Note that "Clean Text", "Extract RegEx" and" InfoGain" Filter nodes represent macro nodes that are implemented using other components.

To generate these regular expression patterns, we first extract and normalize the mentions of biological entities within each document using a "Find & Replace" component to replace all gene names and their aliases with unique tags identifying a gene name ('genexx'). Each document is passed through a series of text pre-processing components ("Clean Text") that perform traditional text cleaning operations (e.g. stemming and stop-word removal), components that identify frequently occurring keywords.

The regular expression patterns are then generated automatically ("Extract RegEx" component) by considering combinations of the unique tag and up to 4 other words. An example of these patterns is

interact\s([a-z]*(\s)+)*genexx[a-z]+\s([a-z]*(\s)+)*bind\s

ID	Pat_1	Pat_2	…	…	Pat_k	Class
ID1	0.8	0.12	0.3	0.12	0.97	Y
ID2						N
…						…
IDN	…	…				…

Figure 2: Example of a Feature Vector Table

Only regular expressions that pass a predefined count threshold are kept and used to generate the feature vector for each document. This feature vector is normalized based on a *tf.idf* approach and passed through a statistical algorithm (based on Information Gain) to keep only the most discriminating patterns.

It is interesting to note, that for the FlyBase document collection, this automatic regular expression generation and evaluation method retained only regular expressions that contained variants of biology-related keywords (e.g. variants of keywords such as 'transcription', 'interaction' and 'binding'). Indeed our final set of feature vectors contained only about 300 regular expression patterns.

Finally, the simplified feature vectors are then used to train a traditional classification algorithm based on support vector machine (SVM) [27]. In order to use the result of the SVM model to classify unseen documents, each new document has to undergo similar preprocessing to represent it as a feature vector that can be passed to the model. Our document classifier, based on this method, proved to produce more accurate results than approaches relying solely on using keyword-based features and secured our team an honourable mention in the competition [11,36].

2.2 Properties of Biological Text Mining Workflows

It is important to note three important properties of biological text mining workflows such as exemplified by the one described above:

1. **Requirement for using a mix of text processing and machine learning components:** A very large number of workflows may be generated to address similar text mining tasks. The structure of such workflows and the components used within them are not fixed, and the choice of which combination to use depends on the task in hand. For example, different users may choose to extract different features from the documents using other methods (we had experimented with different features extraction methods including variations of traditional bag of words approach), and may also wish to use different data mining algorithms (Bayes vs. SVM, etc).
2. **Requirement for integrating background biological information in the analysis:** In most cases effective analysis of biological text documents requires the use of background biological information for identifying biological entities (gene and protein names, disease names, etc) mentioned in the documents. In the classification workflow described above, gene names were simply identified through a lookup operation using a gene and alias list. However, different automatic tools have been suggested recently in the literature to automatically address the named entity recognition task using methods that avoid

problems associated with clashes on ambiguous gene names, and different users may wish to use alternative methods to identify and extract such biological entities.

3. **Requirement for high performance computing resources:** Due to the size of the full-text documents used, the size of the generated feature vectors and the computational intensity of the methods used, the execution of the training phases of the workflows can take several hours even on a small document collection using a desktop machine. Access to dedicated servers or high performance computing (HPC) machines to execute such tasks is usually advisable.

2.3 Automating the Execution of Text Mining Workflows

Our initial implementation to the FlyBase document classification task [11] was based on the use of a mixture of Perl scripts. To address the requirements of text mining workflows discussed above, we have recently ported our implementation to a visual programming workflow environment in the context of the Discovery Net project [5,24]. The visual programming components used are built on top of the KDE knowledge discovery platform and data mining components from InforSense Ltd [15]. The environment allows end user to drag and drop icons representing different processing components and to connect them together as executable workflows. We have extended the environment by implementing a wide range of text processing components that enable end users to manipulate the input text documents and interactively define which features to extract, allowing then to experiment quickly with the suitability of different feature selection methods as well as with the suitability of the different data mining and statistical analysis components available within the system.

Porting the implementation of the classification workflows to the visual programming environment has significantly contributed to better maintenance of the code. It has also improved our ability to experiment with many different feature selection methods and classifiers, as well as to address a number of new problems (e.g. document clustering and co-occurrence analysis) by re-using our existing components in different settings. We have also used the system to integrate and evaluate the use of various 3rd party text-mining tools within our workflows (e.g. POS taggers, named entity recognition components, terminology identification components, etc).

The use of the Discovery Net framework and its component model (Figure 3) also allows us to make use of high performance computing resources, such as clusters of workstations, to speedup the processing of various components within our workflows. This is enabled through the underlying high performance computing architecture of the system [2].

3. BIOINFORMATICS AND WEB SERVICES

One approach to integrating various 3rd party tools within the Discovery Net text mining systems is to download their executable code (when available), and use the Discovery Net tools to develop wrappers over such components (Figure 3) so that they conform to the visual programming interface and to enable data to be passed between them. Our experience using such approach has been extremely successful.

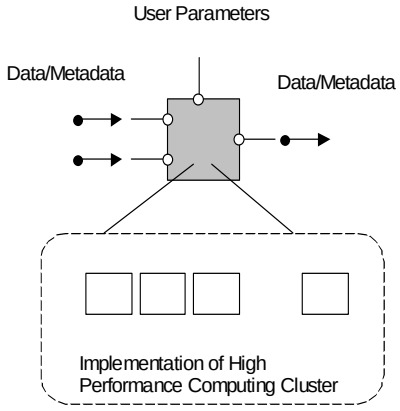

Figure 3: Web service interface / HPC implementation of a text mining component

An alternative approach to the integration of such 3rd party tools is to invoke them remotely using standard inter-operability methods, such as web services. Web Services technology [4,20] has arisen from the W3C consortium as a solution to the problem of inter-operability of distributed software components within a service-centric computing framework, and to provide the common open standard for communication between computational service producers and service consumers.

A Web Service defines explicitly in an XML-based language (Web Service Description Language, WSDL [34]) the interfaces for describing a computational service to a potential client, or user, of the service. This explicit mechanism allows any client to inspect and understand this interface definition and to invoke a particular method of the service using XML invocation mechanism. Different services can be registered within a registry service (e.g. UDDI [32]) thus allowing clients to search for services, and each service is identified by its Universal Resource Identifier (URI), which can be used to resolve access to the service through a particular transport protocol, usually HTTP. The SOAP protocol [28], although not compulsory, is the usual encoding mechanism for message exchanged with a service.

Although web services efforts have targeted e-business applications initially, they are currently becoming popular within scientific applications. For example, over the past few years, the use of web services has become popular for building bioinformatics applications (e.g. [16,24,26,33]) since they provide standardised access methods to remote bioinformatics databases and services.

Within the information retrieval community, search engines such as Google [12] have also started providing web service APIs allowing other programs to post queries to their servers and obtain results automatically. Within the bioinformatics text mining community, several research groups (e.g. [8,10]) have started making terminology servers available through a web service interface. For example, the terminology server of Chang et al [10] provides an XML-RPC interface [35] that allows its services to be invoked remotely through a programmatic interface. Two services are provided, the first returns abbreviations found in a document and the

second returns the gene names and protein names found in it. Each service receives its input as a string containing the document text, and returns an array containing the found entities, their location within the document and score for the confidence of the result.

4. BIOINFORMATICS, TEXT MINING AND INTEROPERABLE METADATA

With the growing popularity of web services in bioinformatics we expect to see such protocols to become widely used also within the text mining for bioinformatics community in the coming years. However, we believe that although the use of web services is essential within a service-centric setting, they are not necessarily sufficient on their own to allow full flexible interoperability between remote services. The missing two ingredients are a mechanism that allows co-ordinating the execution of remote services, and a mechanism that allows higher-level information to be exchanged easily between co-operating services.

4.1 Web Service Co-ordination of web services

One popular approach to co-ordinating the execution of remote web services is through the use of workflow engines. Within the e-business application domain, a popular example is BPEL4WS. Within the scientific application domain, examples include efforts such as Discovery Net [24] and myGrid [16]. Whereas a visual programming environment such as the one described in section 2 allows users to visually construct workflows, a workflow engine is responsible for the co-ordination of the execution of distributed services. The combination of both, a visual programming environment and a workflow engine, provide end users with a higher-level framework for distributed programming that, in many cases, does not require them to develop any conventional code for building their distributed applications.

4.2 Information Abstraction and Metadata

Web services do not restrict what a remote component can accept as input or produce as output; they only provide a standardised mechanism for representing these inputs and outputs together with a mechanism for invoking the component remotely. Our own experiences in integrating 3rd part tools as web services, and also in developing and using our own web services within the Discovery Net project quickly led us to identify a stumbling block in their use. The complete flexibility provided with respect to data structures that can be passed between remote services makes the modelling of data a crucial element when constructing re-usable and efficient text mining workflows, and web service technology on its own does not provide a straightforward solution.

To highlight the problem using a simple example, consider the remote terminology service [10] discussed above. It has been designed to operate in isolation, taking as input only a text document represented as a string, and hence ignoring any biological entities that may have been marked up in the document by previous components within the workflow. We believe that recording the results of previous stages is crucial since it allows components not only to invoke one another, but also to share information about their results.

The solution to the problem is to attach annotations and metadata with the documents being passed between the different software components. Such metadata not only describes the entities found within a document but also any other information that is useful to be passed between services, including providing a basic profile for the document and statistical information about each entity in the text that may be needed by other components in the same workflow.

To address this problem, we have investigated the use of XML-based biological text annotation methods (e.g. [21]). The main limitation we faced with these approaches was that they are based on a hierarchical representation of the text, and thus are not suitable for representing overlapping information structures (e.g. multiple entities identified by different tools) within a single document. In effect, they assume that only one logical view can be imposed on a document based on one specific parsing method. Such a model breaks when different tools, developed by different authors, are integrated within the same analysis workflow.

For text mining, we thus based our annotation metadata approach on a variation of the Tipster Document Architecture model [13] that uses the notion of a document annotation model where a single document is represented by two entities: the document text, which corresponds to the plain document text and the annotation set structure.

The annotation set structure provides a flexible mechanism for associating extra-textual information with certain text segments. Each such text segment is called an annotation, and an annotation set consists of the full set of annotations that a make up a document. A single annotation is defined uniquely by its span (or co-ordinates), and has associated with it a set of attributes. The role of the attributes is to hold additional information, e.g. about the function, the semantics, or other types of user-defined information related to the corresponding text segment.

Table 1: Annotated Text Example

Text	*Annot. Type*	*Attributes*
Insulin	token	pos:noun, stem:insulin
resistance	token	pos:noun, stem:resist
Insulin resistance	compound token	disease:insulin resistance
plays	token	pos:verb, stem:plai
major	token	pos:adj, stem:major
role	Token	pos:noun, stem:role

Each annotation also has a type which in our system, unlike in the original Tipster Architecture, is a low level notion limited to defining the role of the annotation as a constituent part of the document, i.e. as a single word, as a sequence of words, etc. Any type of additional information is represented by the associated attributes. This distinction prevents the conflicting use of annotation types and attributes as means of information holders by reserving this role only to the attributes. Thus each annotation has a unique type and may have any number of attributes. Each attribute has its own type denoting the type of information that the attribute is representing, and a value. Typical examples include attributes that represent the results of a natural language processing operation

such as part-of-speech (POS) tagging, stemming and morphological analysis, the results of dictionary lookups or database queries for certain annotations or the results of a named entity or terminology extraction process. Table 1 shows a simple example of an annotated text. Although shown there as a table, the same information may be easily stored using an XML format for easier exchange between heterogeneous software components. The annotations can also be easily viewed within a text document viewer by highlighting the relevant text as shown in Figure 4.

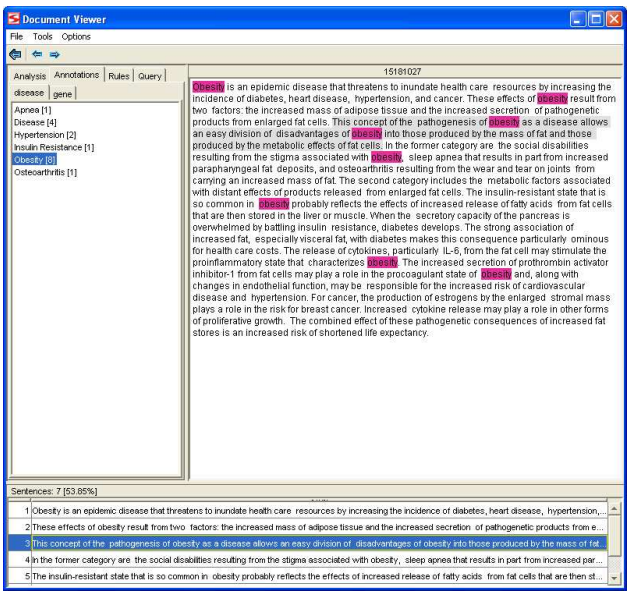

Figure 4: The Document Viewer

An example of the type of available functionality allowed by the inclusion of the metadata is shown in the simple workflow of Figure 5, which computes a co-occurrence matrix between the genes and diseases that appear in a set of PubMed abstract texts as well as extracts feature vectors for the same abstracts. The imported document data are in MedLine format, from which the extraction component extracts the texts of the abstracts. Then, the next three components in the upper branch of the workflow create an annotation set structure for the document collection, and they use a local gene and disease dictionary to identify matching annotations in the abstracts. An index is then generated over the annotation set structure that allows a fast computation of the desired co-occurrence matrix. An extract of the computed co-occurrence matrix is presented in Table 2.

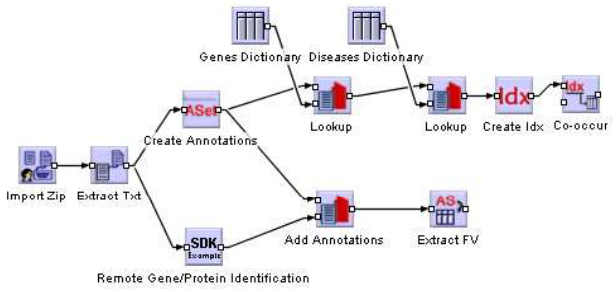

Figure 5: Co-occurrence and Feature extraction workflow

Table 2: Co-occurrences Table (Diseases vs. Genes).

	CRP	AGT	TF	INS	VHL	AA	GC
Hirsutism	0	0	0	4	0	0	0
Hyperandrogenism	1	0	0	6	0	0	1
Hyperinsulinemia	0	0	0	4	0	0	0
Hyperlipidemia	0	0	0	1	0	0	0
Hyperplasia	0	0	0	1	0	0	0
Hypertension	0	1	0	4	0	0	0
Hypertrichosis	0	0	1	0	0	0	0
Hypertrophy	0	0	0	0	0	0	0
Hypopituitarism	0	0	0	1	0	0	0
Infertility	0	0	0	3	0	0	0
Inflammation	1	1	0	2	0	0	1
Insulin Resistance	1	0	0	19	0	0	1

4.3 Distributed Text Mining Workflows

Within a web services setting, the different components used within the workflow can be executing remotely, and independently, and a user may easily decide to replace our simple gene and disease look-up components by calls to more accurate gene and disease look-up web services, that accept the same metadata format. In Figure 5 this is achieved by the lower branch of the workflow in which a component contacts the terminology server [8] using its XML-RPC interface and then based on these results extracts features from the documents.

5. SUMMARY AND CONCLUSIONS

Our work towards mixed data and text mining has been conducted with the context of the Discovery Net project funded under the UK e-Science programme. Using the various tools produced as part of Discovery Net, generating a reusable distributed application workflows becomes the task of selecting the required components and services and connecting them into a process using a workflow model represented in an XML based language, Discovery Process Mark-up Language (DPML) [13]. A workflow created in DPML is reusable and can also be encapsulated as a new executable service through a web service interface for access by other users.

A large number of applications have already been developed using Discovery Net. In the bioinformatics domain, these applications include genome and protein annotation [24] and virus evolution analysis [6]. Within the environmental modelling domain, Liu and Ma [18] describe how they use the system for performing image-mining activities over earthquake satellite images.

The Discovery Net system is generic in terms of the visual workflow model for co-ordination of remote web services and access to high performance computing facilities. However, for each new application domain where the system is used, a domain-specific metadata model is engineered to allow domain-specific informa-

tion sharing between the software components used. For genome and protein sequence annotation, this is the FASTA format, for image mining applications this is a specialised annotation format.

In this paper, we have described how the Discovery Net approach can benefit the development of distributed text mining workflows for bioinformatics. We have described how the infrastructure has been re-engineered to support mixed data and text mining activities and presented the rich metadata model used to support the interaction between different distributed text processing components. We believe that the use of proposed methods offer an appealing and long term solution that would enhance the interoperability between the different text mining tools allowing their re-use in various applications and scenarios.

Acknowledgments

Parts of this work have been supported under the EPSRC funded UK e-Science Project Discovery Net. We thank and acknowledge support from InforSense Ltd. for using their software and visual tools in our development activities. We would also like to thank our colleagues in both the Data Mining Group at Imperial College and at InforSense Ltd. for their support and encouragement. We would especially like to thank Grace Leung for implementing the XML-RPC interface component.

REFERENCES

1. L.A. Adamic, D. Wilkinson, B.A. Huberman, and E. Adar. A Literature Based Method for Identifying Gene-Disease Connections. *IEEE Computer Society Bioinformatics Conference*, 2002.
2. S. AlSairafi, F. S. Emmanouil, M. Ghanem, N. Giannadakis, Y. Guo, D. Kalaitzopoulos, M. Osmond, A. Rowe, J. Syed and P. Wendel. The Design of Discovery Net: Towards Open Grid Services for Knowledge Discovery. Special issue of The *International Journal on High Performance Computing Applications* on Grid Computing: Infrastructure and Applications, Vol. 17 Issue 3. 2003.
3. C. Blaschke, L. Hirschman, A. Valencia. *BioCreative: Critical Assessment for Information Extraction in Biology Workshop*. Granada, Spain, March 28-March 31, 2004.
4. F. Curbera, R. Khalaf, N. Mukhi, S. Tai and S. Weerawarana. The next step in Web services. *Communications of the ACM*. 2003. Volume 46. No. 10.
5. V. Curcin, M. Ghanem, Y. Guo, M. Kohler, A Rowe, and P. Wendel Discovery Net: Towards a Grid of Knowledge Discovery. In *Proceedings of KDD-2002, the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. July 23-26, 2002 Edmonton, Canada.
6. V. Curcin, M. Ghanem and Y. Guo. SARS Analysis on the Grid. *UK e-Science All Hands Meeting*, Nottingham UK, September 2004.
7. J.T. Chang, H. Schutze, and R.B. Altman. Creating an online dictionary of abbreviations from Medline. *Journal of American Medical Informatics Association.* 2002 Nov-Dec;9(6):612-20.
8. J.T. Chang, H. Schutze, and R. B. Altman Biomedical Abbreviation Server. http://bionlp.stanford.edu/abbreviation
9. FlyBase: http://www.flybase.org
10. R. Gaizauskas, H. Cunningham, Y. Wilks, P. Rodgers, and K. Humphreys. GATE: An Environment to Support Research and Development in Natural Language Engineering. In *Proceedings of the 8th IEEE International Conference on Tools with Artificial Intelligence (ICTAI-96)*, pages 58-66. IEEE Computer Society, October 1996.
11. M.M. Ghanem, Y. Guo, H. Lodhi, and Y. Zhang. Automatic Scientific Text Classification Using Local Patterns: KDD CUP 2002 (Task 1), *SIGKDD Explorations*, 2002. Volume 4, Issue 2.
12. Google: http://www.google.com/apis/
13. R. Grishman. TIPSTER Text Architecture Design. http://www.itl.nist.gov/iaui/894.02/related_projects/tipster/docs/arch31.doc.
14. H. Harkema, R. Gaizauskas, M. Hepple, A. Roberts, I. Roberts, N. Davis and Y. Guo. A Large Scale Terminology Resource for Biomedical Text Processing. *NAACL/HLT 2004 Workshop on Linking Biological Literature, Ontologies and Databases: Tools for Users*, 2004.
15. InforSense Ltd. http://www.inforsense.com
16. R. de Knikker R, Y. Guo, J. L. Li, A. K. Kwan, K. Y. Yip, D. W. Cheung and K. H. Cheung. A web services choreography scenario for interoperating bioinformatics applications. *BMC Bioinformatics* 2004. Volume 10 No 5.
17. D. Landau, R. Feldman, Y. Aurmann, M. Fresko, Y. Lindell, O. Liphstat, and O. Zamir. TextVis: An Integrated Visual Environment for Text Mining. *Second European Symposium on Principles of Data Mining and Knowledge Discovery PKDD '98*, Nantes, France, 1998.
18. J. G. Liu and J. Ma. Imageodesy on MPI & GRID for Coseismic Shift Study Using Satellite Optical Imagery. *UK e-Science All Hands Meeting*, Nottingham UK, September 2004.
19. E. M. Marcotte, I. Xenarios, and D. Eisenberg. Mining literature for protein-protein interactions, *Bioinformatics*. 2001 Apr;17(4):359-63.
20. J. Martin, A. Arsanjani, P. Tarr and B. Hailpern. Web Services: Promises and Compromises. 2003. *ACM Queue* Volume 1. No. 1.
21. T. Ohta, Y. Tateisi, H. Mima, and J. Tsuj. GENIA Corpus: an Annotated Research Abstract Corpus in Molecular Biology Domain. In the Proceedings of the *Human Language Technology Conference (HLT 2002*, 2000.
22. S. Raychaudhuri, H. Schutze, R. B. Altman. Using Text Analysis to Identify Functionally Coherent Gene Groups. Genome Research, p. 1582–1590. 2002 http://www.pdg.cnb.uam.es/BioLINK/workshop_BioCreative_04/handout/index.html
23. T.C. Rindflesch, L. Tanabe, J. N. Weinstein, and L. Hunter; EDGAR: Extraction of Drugs, Genes and Relations from the Biomedical Literature *Pacific Symposium on Biocomputing* 5:514-525. 2000.

24. A. Rowe, D. Kalaitzopoulos, M. Osmond, M Ghanem, Y. Guo. The Discovery Net System for High Throughput Bioinformatics. In *Proceedings of the Eleventh International Conference on Intelligent Systems for Molecular Biology*, 2003. Also appears in *ISMB (Supplement of Bioinformatics)* 2003: 225-231

25. H. Shatkay and R. Feldman. Mining the Biomedical Literature in the Genomic Era: An Overview. *Journal of Computational Biology*. 2003. Volume 10. No. 6.

26. H. Sugawar, S. Miyazaki. Biological SOAP servers and web services provided by the public sequence data bank. *Nucleic Acids Research*. 2003 Volume 31 No. 13.

27. SVM light, http://svmlight.joachims.org/

28. SOAP: http://www.w3.org/TR/SOAP

29. J. Syed, Y. Guo, and M. Ghanem. Discovery Processes: Representation And Re-Use, *UK e-Science All Hands Meeting*, Sheffield UK, September, 2002.

30. L. Tanabe and W. J. Wilbur. Tagging Gene and Protein Names in Biomedical Texts. *Bioinformatics*. 2002 Volume 18, No. 8.

31. TREC Genomics Track. http://medir.ohsu.edu/~genomics/

32. UDDI: http://www.uddi.org

33. M. D. Wilkinson and M. Links. BioMOBY: an open source biological web services proposal. *Briefings in Bioinformatics*. 2002 Volume 3 No. 4.

34. WSDL: http://www.w3.org/wsdl

35. XML RPC.: http://www.xml-rpc.org

36. A. Yeh, L. Hirschman, A. Morgan, Background and Overview for KDD Cup 2002 Task 1: Information Extraction from Biomedical Articles, *SIGKDD Explorations*, 2002. Volume 4, Issue 2.