Automated Information Extraction from Gene Summaries

Thierry Charnois, Nicolas Durand, and Jiří Kléma

GREYC, CNRS - UMR 6072, Université de Caen
Campus Côte de Nacre, F-14032 Caen Cédex France
{Forename.Surname}@info.unicaen.fr

Abstract. Automated extraction of links among biological entities from free biological texts has proven to be a difficult task. In this paper we propose and solve a modified task in which we extract the links from short textual gene summaries collected automatically from NCBI website. The main simplification lies in the fact that each summary is unambiguously attached to a single gene. The agent part of binary biological interactions is thus known by default, the goal is to identify meaningful target parts from the summary. The outcome is a structured representation of each summary that can be used as background knowledge in consequent mining of gene expression data. As the gene summaries highly interact with the other structural information resources provided by NCBI website, these resources can be used as an annotation tool and/or a feedback for performance optimization of the system being developed. In particular we use the gene ontology terms in order to evaluate and improve the information extraction process.

Keywords: genomics, text mining, biological information extraction.

1 Introduction

As availability of textual information related to biology increases, research on information extraction (IE) is rapidly becoming an essential component of various bio-applications. It is expected that text mining in general, and IE in particular, will provide tools to facilitate the annotation of a large amount of genetic information, including gene sequences, transcription profiles and biological pathways. The biological function of cells, tissues and organisms can be understood by examination of interactions among proteins or between DNA and proteins.

The main interest has been devoted to MedLine abstracts, however there is also a vast effort to exploit full-text journal articles [20]. Applying IE to genomics and more generally to biology is not an easy task because IE systems require deep analysis methods to extract the relevant pieces of information. That is why we propose a modified task in which we extract the links from short textual gene summaries collected automatically from NCBI website. The main simplification lies in the fact that each summary is unambiguously attached to a single gene. The agent part of binary biological interactions is thus known by default, the goal is to identify meaningful target parts from the summary.

This work has started with the intention to develop a meaningful measure of interaction inside a closed set of genes in order to support consequent mining of gene expression data. Such a measure can be used in many ways. The measure can complement the gene distance measure based immediately on the expression data when the genes are clustered [15]. It can be used to select biologically meaningful patterns from the overwhelming pattern sets that technically appear in the expression data [17] or it can help in feature extraction and selection when a classification task is solved [24].

Public databases contain vast amount of rich data that can be used to create and evaluate both direct and indirect interactions among biological entities. Of course, the most straightforward way is to utilize the structured information such as gene ontology (GO) or Entrez's link files. The rationale sustaining the GO based measure is that the more GO terms the genes share, and the more specific the terms are, the more likely the genes are to be functionally related. [19] defines a distance based on the Czekanowski-Dice formula, the methodology is implemented within the GOProxy tool of GOToolBox [1]. [22] uses Entrez's link files in order to create a general entity graph. The authors also provide a measure that assesses the strength of a link between an arbitrary pair of vertices.

Nevertheless, the structured databases can hardly summarize all the available knowledge and text mining outcomes can reasonably complement the information gained from the knowledge sources mentioned in the previous paragraph. Tagging gene and protein names in free biomedical text has proven to be a difficult task [23]. Automated extraction of direct links among biological entities is even more difficult [10]. In this paper we restrict to a corpus of gene textual summaries. Possible interaction among a closed set of genes is studied indirectly. The main aim of the paper is to develop a structured and tagged representation of gene summaries. This structured representation can later serve to assume on interaction or similarity among the genes from multiple points of view. The proposed structured representation also seems to be promising with respect to its further generalization. The developed system provides an insight into relations among biological entities and it can be adjusted to extract arbitrary interactions among biological entities from abstracts or whole texts.

This paper is structured as follows. Section 2 briefly introduces the data we worked with. Section 3 gives an overview of related methods. Section 4 describes a tool LinguaStream that we have used, discusses the developed extraction rules and gives examples of real outputs. Section 5 provides two ways of evaluation – the first is based on a limited corpus of human annotated summaries, the second evaluates the full corpus with respect to GO terms.

2 Entrez Gene Summaries

Entrez Gene is the gene-specific database at the National Center for Biotechnology Information (NCBI), a division of the National Library of Medicine (NLM). Entrez Gene provides unique integer identifiers for genes and other loci for a subset of model organisms. It tracks those identifiers, and is integrated with the Entrez system for interactive query, LinkOuts, and access by E-utilities [25].

The information that is maintained includes nomenclature, chromosomal localization, gene products and their attributes (e.g. protein interactions), associated markers, phenotypes, interactions, and a wealth of links to citations, sequences, variation details, maps, expression reports, homologs, protein domain content and external databases.

As mentioned in Section 1 the long term goal is to develop a meaningful measure of interaction inside a closed set of genes. In our experiments we deal with the SAGE human gene expression dataset downloaded from [4]. Only the unambiguous tags (corresponding to genes) identified with RefSeq were selected, leaving a set of 11082 tags (expressed in 207 biological situations).

To access the gene annotation data for every tag considered, RefSeq identifiers were translated into EntrezGene identifiers [3]. The mapping approached 1 to 1 relationship. There were only 11 unidentified RefSeqs, 24 RefSeqs mapped to more than 1 id and 203 ids still appeared more than once. Knowing the gene identifiers, the annotations were automatically accessed through hypertext queries to the Entrez Gene database [4] and sequentially parsed by the method stemming from [28]. The non-trivial textual records were obtained for 6,302 ids which makes 58% of the total amount of 10,858 unique ids. 3,926 genes had a short summary, 5,109 had one abstract attached at least. 6,824 genes had at least a single GO term attached, which makes 63% of the total amount of genes.

3 Information Extraction and Methodology

Many approaches have been proposed for extraction of biological information from scientific texts. These approaches can be classified into two broad categories [8]: machine learning based and linguistic analysis based. That is the latter one, and more precisely IE technique, which is used in this paper. Some of the IE systems use similar approaches with the Natural Language Processing (NLP) understanding systems of the seventies/eighties and IE is often seen as a NLP understanding system. In fact, they do not share the same goal. IE aims at extracting very precise information from a restricted domain while the goal of the NLP systems was the whole understanding of all aspects of the text. For this purpose, extensive knowledge and linguistic resources were needed, and deep analysis was necessary (syntactic, semantic and pragmatic analysis).

Taking advantage of the restricted domain, some biomedical IE systems adopt this NLP based architecture [11, 14]. Nevertheless, the syntactic analysis still remains a difficult task. Actually the accuracy of the complete parsing can be estimated roughly about 50% of the analysed sentences (see [11]). Other works attempt to use “shallow parsing”, a robust method, although less precise performing a partial decomposition of a sentence structure to identify phrasal chunks or entities of interest and relations between these entities. Generally, these kinds of systems are designed for extracting protein-protein relations, such as protein-location relations, binding relations, gene-gene interactions, etc. [21]. A common point of the IE systems is that they utilize resources, biological databases, ontologies, such as UMLS, LocusLink ...

Some other papers are devoted to a preliminary task: the recognition of gene/protein names and families. Difficulties are well known: multi-sense words, no formal criterion, multi-word terms, variations in gene/protein names. Different NLP methods are used for this like rule-based approach [12], or/and dictionary/knowledge approach [16, 18].

Our system differs from the approaches previously mentioned in several ways. For example, [14] carries out a terminological parsing, using a biological knowledge database, syntactic, semantic and discoursive analysis, using a domain model (ontology) to get a predicate argument representation and to fill an extraction template. Instead of those kinds of classical NLP techniques, we design simple declarative extraction rules, making the implementation process "light and quick". Let us note that they are domain-specific, but by no means corpus-specific. One of the aims is to reach similar results as the "heavy" methods published in the literature.

The design of the rules can be seen as a simplification of the "contextual exploration method" [26]. This approach aims at locating contexts in a corpus (i.e., linguistic indicators) from which some rules for identifying relevant textual segments are triggered. For example, linguistic indicators as "our conclusion", "consequently" or "so" found in a corpus can allow the extraction of conclusive sentences. That is the idea of our system: triggering extraction rules only if a context is located while avoiding the whole-corpus analysis. Another similarity with our work is that no syntactic analysis is processed. However, unlike our approach, this method is domain-independent, so linguistic indicators and extracted informations are general (causality, thematic announcement, conclusive sentences, . . .).

Another important point is that our method is endogenous: no resources such as knowledge base or dictionary are needed at the beginning. The resources are constructed on the fly – the system learns new terms (which can be new terms in the domain or missing in the databases) to be used later or in other biological corpora and/or in other text mining applications.

Finally, our system is not designed to focus only on a specific aspect of gene/protein description but it is designed to identify protein/family/name and general biological function about the gene/protein involved. Actually four *types* of information are distinguished and annotated in the corpus: gene/protein name, family name, location and biological function.

4 Method

The presented approach consists in definition of extraction rules, and has been implemented using the LinguaStream platform.

4.1 LinguaStream

LinguaStream [2, 7] is an integrated experimental environment targeted to NLP researchers. It allows complex experiments on corpora to be realised conveniently, using various declarative formalisms.

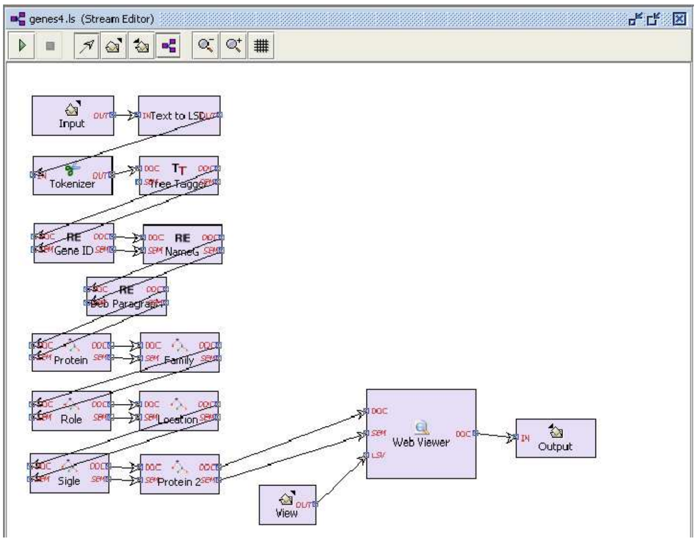


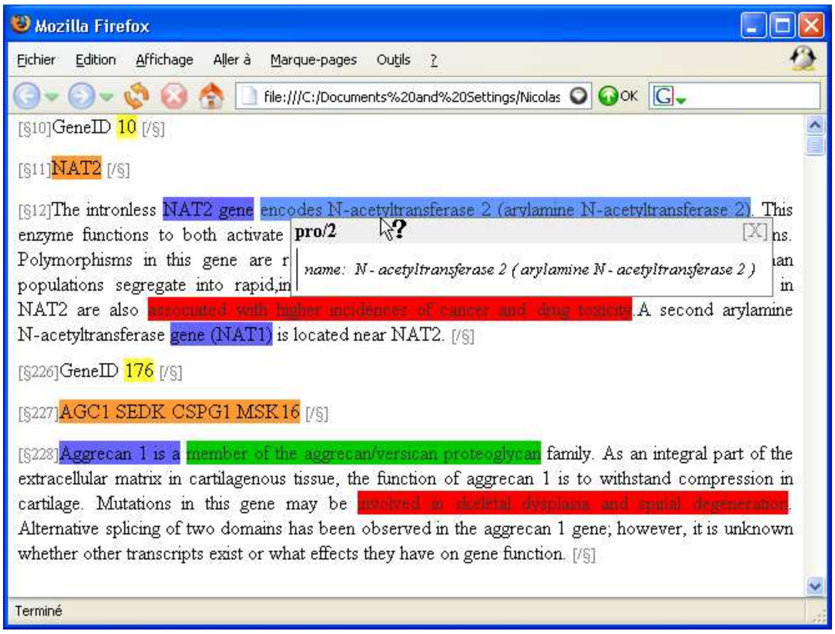
Fig. 1. Processing stream of the implemented rules in LinguaStream.

Its integrated environment allows processing streams to be assembled visually (see Figure 1), picking individual components from a "palette". Some components are specifically targeted to NLP, while others solve various issues related to document engineering (especially to XML processing). Annotations made on a single document are organized in independent layers and may overlap. Thus, concurrent and ambiguous annotations may be represented in order to be solved afterwards, by subsequent analysers. The platform is systematically based on XML recommendations and tools, and is able to process any file in this format while preserving its original structure. When running a processing stream, the platform takes care of the scheduling of sub-tasks, and various tools allow the results to be visualised conveniently.

IE from a raw text is composed of tokenization, POS tagging (using TreeTagger [5]), extraction and output generation which adds the final XML wrapper. Among fundamental principles, the platform allows the **declarative representations** to be used. Furthermore, the **modularity** of processing streams promotes the **reusability** of components in various contexts: a given module, developed for a first processing stream may be used in other ones. Section 4.2 demonstrates their utility.

4.2 Extraction rules

We have defined a set of rules to identify, extract and annotate relevant multi-word terms from gene summaries. The results are given in a form of XML file containing the whole text where the recognized areas are highlighted and clickable (see Figure 2), and another XML file with the extracted information only (see Figure 3). Let us refer to these files as the *interactive* and *extracted* output.


Mozilla Firefox
Fichier Edition Affichage Aller à Marque-pages Outils ?
file:///C:/Documents%20and%20Settings/Nicolas
OK
Terminé

[§10]GeneID 10 [/§]

[§11]NAT2 [/§]

[§12]The intronless NAT2 gene encodes N-acetyltransferase 2 (arylamine N-acetyltransferase 2). This
enzyme functions to both activate
Polymorphisms in this gene are r
populations segregate into rapid,in
NAT2 are also associated with higher incidences of cancer and drug toxicity.A second arylamine N-acetyltransferase gene (NAT1) is located near NAT2. [/§]

pro/2 [X]

name: N - acetyltransferase 2 (arylamine N - acetyltransferase 2)

ns.
ıan
in

[§226]GeneID 176 [/§]

[§227]AGC1 SEDK CSPG1 MSK16 [/§]

[§228]Aggrecan 1 is a member of the aggrecan/versican proteoglycan family. As an integral part of the extracellular matrix in cartilagenous tissue, the function of aggrecan 1 is to withstand compression in cartilage. Mutations in this gene may be involved in skeletal dysplasia and spinal degeneration. Alternative splicing of two domains has been observed in the aggrecan 1 gene; however, it is unknown whether other transcripts exist or what effects they have on gene function. [/§]

Fig. 2. Example of XML result.

The rule definition is decomposed into the following steps:

- Observe the corpus (in fact the training corpus) in order to get regularities and identify some contexts. For example, the expression "this gene encodes the X protein" has numerous occurrences in the corpus, so "encode" is a good context – a trigger word – to identify a protein name;
- Design rules from the contexts previously identified (a particular example is shown later).
- Implement the rules. It is a straightforward process using DCG Prolog and unification on feature structures thanks to GULP [9].
- Review the results after processing the rules and backtrack if necessary. This can be changing rules, or adding rules while possibly reusing the terms/knowledge already recognized/learned.

We have defined 4 sets of rules allowing the system to recognize 4 types of information: protein names, family names of proteins, roles / biological functions (including diseases, interactions, . . .), and location (components, . . .).

General structure of the rules We do not use patterns in the sense of the IE, that is without an a priori on the form of the expressions. Figure 4 presents the structure of the rules. From a "context", an expression (generally a multi-word term, a nominal phrase) is recognized until a stop phrase is encountered. The context is a set of "trigger" words. The stop phrases can be words, symbols, verbs, punctuation, ... They depend on the rule type.

```
<gene>
<id>10</id>
<name>NAT2</name>
<protein>encodes N-acetyltransferase 2 (arylamine N-acetyltransferase 2)</protein>
<role>associated with higher incidences of cancer and drug toxicity</role>
</gene>
<gene>
<id>176</id>
<name>AGC1 SEDK CSPG1 MSK16</name>
<protein>Aggrecan 1 is a</protein>
<family>member of the aggrecan/versican proteoglycan</family>
<role>involved in skeletal dysplasia and spinal degeneration</role>
</gene>
```

Fig. 3. The extracted results.

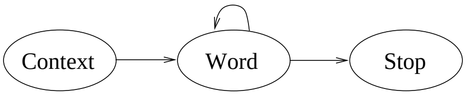


Fig. 4. Structure of the rules.

Let us take an example. The term "encodes N-acetyltransferase 2 (arylamine N-acetyltransferase 2)." is extracted by using the following set of rules:

```
protein(type:pro..name:N) --> @lemma:encode, np(N).
np(N) --> @tag:dt, namepro(N).
np(N) --> namepro(N).
namepro(N) --> ls_token(N,_), end.
namepro(N) --> ls_token(N,_), namepro(N2), {concat(N1,N2,N)}.
end --> punctuation ; verb ; relative_pronoun ; trigger_word.
```

In this example, "namepro" stands for the name of the protein to be extracted, "ls_token" a terminal symbol (a token), "end" the indicator for cutting out the recognition of a multi-word term. The trigger phrase is "encode" (i.e., also encoded, encodes atc.). Let us remark that "dt" corresponds to a possible determiner just before the name of the protein. The rules "namepro" allow the system to recognize the multi-word terms. The end phrase is a punctuation symbol here.

Context and stop phrases Currently, the identification of the context has been done manually, however an automatic learning of the context can also be considered. We have manually detected special phrases for each type of information (proteins, families, . . .) on an excerpt of the corpus. We have noted the corresponding trigger words and the stop phrases. Table 1 presents some examples of trigger phrases for each type. The common stop phrases are the trigger words, some punctuation symbols and the relative pronouns.

The final rulebase consists of 186 rules – 74 for proteins, 46 for families, 27 for roles, 39 for locations.

Special processing Another set of rules benefits from the reusability principle of LinguaStream. It enables us to use the information (tokens, trigger phrases, ...) recognized earlier within the current processing stream. Some protein names/families are recognized using entities already identified. These entities are considered as lexical units (tokens) in the rules. For instance, "`X are class of FAMILY`" where `FAMILY` is the entity previously recognized as a protein family, and `X` is the new extracted information: here an other protein family.

A special process for recognizing protein names expressed by an acronym is done. All acronyms are marked by a few special rules that extract words with upper cases, numerical figures and/or special symbols as in [12]. Then, the acronym context is used to decide whether the acronym corresponds to a protein name. For example, "protein CEBP-alpha" is detected using these specific rules. Here the rule is: the word "protein" followed by an acronym. We also use particular rules to filter out false or misleading expressions. For instance, the term "the secreted protein" must not be extracted as a protein name.

proteins	encodes an ...	@lemma:encode, @tag:dt
	the product of this gene is ...	@lemma:product, $'of', $'this', $'gene', @lemma:be
families	belongs to the ...	@lemma:belong, $'to'
	is a member of the ...	@lemma:member, $'of'
roles	an important role in ...	$'role', $'in'
	is involved in ...	$'involved', $'in'
locations	found in ...	$'found', $'in'
	located in ...	$'located', $'in'

Table 1. Examples of detected contexts.

4.3 Outcome

The corpus is about 2.33MB and contains 64,308 lines. There are 10,858 genes. In order to learn the contexts and the stop phrases, we have looked over 200 genes (1.8% of the corpus).

Type	No. text areas
proteins	3,058
families	3,056
roles	4,303
locations	1,023

Table 2. The number of recognized terms (text areas) according to their type.

The number of marked text areas (i.e., multi-word terms or pieces of extracted information) is presented in Table 2. Let us note that in a gene summary, the information about proteins, families, ... is not always present. We observe that the number of "roles" is larger than the number of "proteins". As a matter of fact, a single gene may have several "roles" because the type role contains biological functions, diseases and also different interactions. As regards families, we capture families and also subfamilies and superfamilies, if they are indicated.

The system has recognized 3,058 protein names. By rule of thumb, this is a relatively good result since there are 6,932 genes without summaries (i.e. without any chance to extract information). On average, there is nearly one protein name extracted per existing summary. A more detailed evaluation of the performance is given in the next Section.

5 Evaluation

Two types of experiments have been carried out. First, we have evaluated the precision and the recall of the method using an excerpt of the data. The second experiment is a direct comparison between our extracted terms and the GO terms annotating the individual genes.

5.1 Evaluation on a human annotated corpus

We have evaluated our approach using 100 genes (and the corresponding summaries) randomly chosen. This excerpt has been annotated by two local experts to form the reference. We have computed the classical measures of precision and recall [8] to assess the performance of our system.

Table 3 presents the results and relates them to the results obtained by other existing methods published in literature. The comparison is illustrative only as the methods were not applied to the same corpus. The precision and recall values cannot be compared directly, but they may give an estimation of the performance. As we can see, the results of our system are comparable to the existing scores, without using a "heavy method" nor resources.

method		recall	precision
existing methods: [12, 18, 13, 27]		73-99%	73-95%
our approach:	proteins	73,6%	78,8%
	families	71,6%	93,4%

Table 3. Results.

Distinguishing various term types, a good precision and recall has been reached for families. Actually, for the family names and the locations, the implemented rules are appropriate to extract information from summaries. Moreover, we have recently improved the rules by using the results of this evaluation and by observing the information not recognized to define new contexts.

As for biological functions, the important point is to have a relatively complete list of the commonly used verbs. Our "list" is good enough, and it is easy to add new verbs to capture more cases. For this, ideas from specific works like [6] can be used.

The main problem concerns the recognition of proteins names. The current rules are able to capture the majority of the names, however some particular linguistic problems are not treated yet: anaphoras and coordination. For instance, in the phrase "the related proteins CEBP-alpha, CEBP-delta, and CEBP-gamma" all the three acronyms are recognized but only the first one ("CEBP-alpha") is identified as a protein name (the rule given above). The others would need to take the coordination problem into account.

5.2 Comparison with GO

A great part of GO terms associated with genes also appear in their summaries. In other words, if a gene is annotated with a GO term, this term (or its semantically equivalent phrase) often appears in the summary of the given gene too. This significant overlap between GO terms and summaries gives us a chance to utilize GO terms as an annotation tool for gene summaries. The quality of information extraction can be tested with respect to recall of the known GO terms. The main advantage of such an experiment is that it enables us to automatically evaluate the system over whole the corpus of gene summaries.

The basic assumption of this evaluation is that all the GO terms represent meaningful terms to be extracted. Then, the recall is estimated as the ratio between the number of GO terms identified within the extracted XML annotation and the number of GO terms that appear within the original summaries. The ideal case occurs when all the GO terms that appear within the gene summary of the given gene remain also in its XML record – recall would be 1 here.

The main and difficult problem is to identify the GO terms within free text of gene summaries. First, let us see what is the percentage of GO terms that appear in gene summaries immediately – as exactly the same term or phrase. The simple search for substrings suggests that only 7% of GO terms associated with the given gene co-appear in its summary immediately. These are mainly one word terms since for longer phrases the exact match is less likely – e.g., the GO term "amino acid metabolism" appears in the summary as an expression "function in the catabolism and salvage of acylated amino acids". That is why we have also applied a simple form of approximate match for longer phrases. If at least one of the stemmed words from the GO phrase appears in the gene summary exactly, we search for an approximate match of the other words in the same summary sentence. We use the bigram approximate string comparison for this purpose. The phrase is found if and only if the average of best-match values – we search for the nearest counterpart for all the words from the GO phrase – reaches a certain threshold. This simple approximate match reveals that 18% of GO terms associated with the given gene co-appear in the respective summary.

matching	original summaries	extracted XML	recall
exact	7%	3.9%	56%
approximate	18%	8.2%	46%

Table 4. Recall of GO terms – the exact and approximate match.

Table 4 gives an overview of the recall for exact and approximate GO terms. Precision of the IE cannot be revealed in this way as we do not search for GO terms only. Let us remind that we are interested in any biological terms and the goal is not to confine ourselves to the limited dictionary of GO terms. Nevertheless, the recall presented in Table 4 should be evaluated with respect to the condensation that the extraction process brings. The content of the extracted output makes 29.2% of the content of the original gene summary files. It also has to be considered that the sentences in gene summaries can be quite long while the extracted tags are quite compact. The chance that the GO phrase is scattered by tag tokens is thus increased.

6 Conclusion

In this paper, we presented an original approach for extracting and exploiting information from biological domain. The approach gives promising results on a specific but wide corpus. It is suitable for extracting biological information as well as to acquire knowledge. It also seems to be promising with respect to its further generalization. The developed grammar provides an insight into relations among biological entities and it can be adjusted to extract arbitrary interactions among biological entities from abstracts or whole texts using the acquired information such as terminological resources. The second investigation will consist in enhancing the grammar by an automated learning of context [6]. The process has not been designed yet but the terms already learned can guide and accelerate the learning process (to annotate the other corpus, etc.).

Another work is to propose and test a gene similarity measure based on the developed structured representation. While the measure itself is more or less obvious – the more overlap two genes show in their corresponding type fields the more they interact – the main effort will be to show a possible difference in comparison with the measures that are immediately based on the GO annotations [19] or a vector representation of whole summaries [17].

Acknowledgements. The authors thank A. Widlöcher and F. Bilhaut (the Linguastream team), and the CGMC Laboratory (CNRS UMR 5534, Lyon, France) for providing the gene expression database. This work has been partially funded by the ACI "masse de données" (French Ministry of research), Bingo project (MD 46, 2004-2007).

References

[1] GOToolBox website: http://crfb.univ-mrs.fr/gotoolbox/.

[2] LinguaStream website: http://www.linguastream.org/.

[3] MatchMiner website: http://discover.nci.nih.gov/matchminer/.

[4] NCBI website: http://www.ncbi.nlm.nih.gov/.

[5] TreeTagger website: http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/.

[6] P. Bessières, G. Bisson, A. Nazarenko, C. Nédellec, M. Ould Abdel Vetah, and T. Poibeau. Ontology Learning for Information Extraction in Genomics Bibliography - the Caderige Project. In *Journées IMPG Ontologie et Extraction d'Information en Génomique*, Grenoble, France, May 2001.

[7] F. Bilhaut and A. Widlöcher. LinguaStream: An Integrated Environment for Computational Linguistics Experimentation. In *the European Chapter of the Association of Computational Linguistics (Companion Volume)*, Trento, Italy, 2006.

[8] K. B. Cohen and L. Hunter. *Artificial Intelligence Methods and Tools for Systems Biology*, volume 5, chapter Natural Language Processing and Systems Biology. Springer Verlag, 2004.

[9] M. A. Covington. GULP 3.1: An Extension of Prolog for Unification-Based Grammar, 1994.

[10] J. Cussens and C. Nédellec, editors. *Proceedings of the 4th Learning Language in Logic Workshop (LLL05)*, Bonn, August 2005.

[11] N. Daraselia, S. Egorov, A. Yazhuk, S. Novichkova, A. Yuryev, and I. Mazo. Extracting Protein Function Information from MEDLINE Using a Full-Sentence

Parser. In *ECML/PKDD Workshop on Data Mining and Text Mining for Bioinformatics*, Pisa, Italy, Sept. 2004.

[12] K. Fukuda, A. Tamura, T. Tsunoda, and T. Takagi. Toward Information Extraction: Identifying Protein Names from Biological Papers. In *Pacific Symposium Biocomputing (PSB'98)*, pages 362–373, Hawaii, Jan. 1998.

[13] K. Fundel, D. Güttler, R. Zimmer, and J. Apostolakis. A Simple Approach for Protein Name Identification: Prospects and Limits. *BMC Bioinformatics*, 6(Suppl 1), 2005.

[14] R. J. Gaizauskas, G. Demetriou, P. J. Artymiuk, and P. Willett. Protein Structures and Information Extraction from Biological Texts: The PASTA System. *Bioinformatics*, 19(1):135–143, 2003.

[15] P. Glenisson, J. Mathys, and B. D. Moor. Meta-Clustering of Gene Expression Data and Literature-Based Information. *SIGKDD Explor. Newsl.*, 5(2):101–112, 2003.

[16] K. Humphreys, G. Demetriou, and R. Gaizauskas. Two Applications of Information Extraction to Biological Science Journal Articles: Enzyme Interactions and Protein Structures. In *Pacific Symposium Biocomputing*, pages 505–516, Hawaii, Jan. 2000.

[17] J. Kléma, A. Soulet, B. Crémilleux, S. Blachon, and O. Gandrillon. Mining Plausible Patterns from Genomic Data. In *the 19th IEEE International Symposium on Computer-Based Medical Systems*, pages 183–188, Salt Lake City, Utah, 2006.

[18] A. Koike and T. Takagi. Gene/Protein/Family Name Recognition in Biomedical Literature. In *Linking Biological Literature, Ontologies and Databases: Tools for Users, Workshop in conjunction with NAACL / HLT 2004*, pages 9–16, 2004.

[19] D. Martin, C. Brun, E. Remy, P. Mouren, D. Thieffry, and B. Jacq. GOToolBox: Functional investigation of gene datasets based on gene ontology. *Genome Biology*, 5(12):R101, 26 Nov. 2004.

[20] S. K. Parantu, P.-I. Carolina, B. Peer, and A. A. Miguel. Information extraction from full text scientific articles: Where are the keywords? *BMC Bioinformatics*, 4:20, 2003.

[21] J. Pustejovsky, J. Castano, J. Zhang, M. Kotecki, and B. Cochran. Robust Relational Parsing over Biomedical Literature: Extracting Inhibit Relations. In *Pacific Symposium on Biocomputing (PSB'02)*, pages 362–373, Hawaii, Jan. 2002.

[22] P. Sevon, L. Eronen, P. Hintsanen, K. Kulovesi, and H. Toivonen. Link Discovery in Graphs Derived from Biological Databases. In *3rd International Workshop on Data Integration in the Life Sciences 2006 (DILS'06)*, Hinxton, UK, July 2006.

[23] L. Tanabe and W. J. Wilbur. Tagging gene and protein names in biomedical text. *Bioinformatics*, 18:1124–1132, 2002.

[24] J.-P. Vert and M. Kanehisa. Graph-Driven Feature Extraction From Microarray Data Using Diffusion Kernels and Kernel CCA. In S. Becker, S. Thrun, and K. Obermayer, editors, *NIPS*, pages 1425–1432. MIT Press, 2002.

[25] D. Wheeler, D. Benson, and S. Bryant. Database Resources of the National Center for Biotechnology Information: Update. *Nucleic Acids Res.*, 33:D39–D45, 2005.

[26] D. Wonsever and J.-L. Minel. Contextual Rules for Text Analysis. In *CICLing '01: Proceedings of the Second International Conference on Computational Linguistics and Intelligent Text Processing*, pages 509–523, London, UK, 2001. Springer.

[27] H. Yu, V. Hatzivassiloglou, C. Friedman, A. Rzhetsky, and W. Wilbur. Automatic Identifying Gene/Protein Terms in MEDLINE Abstracts. *Journal of Biomedical Informatics*, 35(5-6), 2002.

[28] F. Zelezny, J. Tolar, N. Lavrac, and O. Stepankova. Relational Subgroup Discovery for Gene Expression Data Mining. In *EMBEC: 3rd IFMBE European Medical & Biological Engineering Conf.*, November 2005.