CONCURRENCY AND COMPUTATION: PRACTICE AND EXPERIENCE
Concurrency Computat.: Pract. Exper. 2014; **26**:1771–1786
Published online 18 September 2013 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/cpe.3130

SPECIAL ISSUE PAPER

Discovering biological knowledge by integrating high-throughput data and scientific literature on the cloud

C. Spampinato [1,*,†], I. Kavasidis [1], M. Aldinucci [2], C. Pino [1], D. Giordano [1] and A. Faro [1]

[1] *Department of Electrical, Electronics and Computer Engineering, University of Catania, 95125 Catania, Italy*
[2] *Computer Science Department, University of Torino, 10149 Torino, Italy*

SUMMARY

In this paper, we present a bioinformatics knowledge discovery tool for extracting and validating associations between biological entities. By mining specialized scientific literature, the tool not only generates biological hypotheses in the form of associations between genes, proteins, miRNA and diseases but also validates the plausibility of such associations against high-throughput biological data (e.g. microarray) and annotated databases (e.g. Gene Ontology). Both the knowledge discovery system and its validation are carried out by exploiting the advantages and the potentialities of the Cloud, which allowed us to derive and check the validity of thousands of biological associations in a reasonable amount of time. The system was tested on a dataset containing more than 1000 gene–disease associations achieving an average recall of about 71%, outperforming existing approaches. The results also showed that porting a data-intensive application in an Infrastructure as a Service cloud environment boosts significantly the application's efficiency. Copyright © 2013 John Wiley & Sons, Ltd.

Received 28 February 2013; Revised 15 July 2013; Accepted 19 July 2013

KEY WORDS: text mining; knowledge discovery; bioinformatics

1. INTRODUCTION

A huge amount of biomedical information is hidden in millions of scientific articles published in the last 25 years, and this quantity is exponentially increasing. This overwhelming quantity of information in the scientific literature compels, therefore, the need for new methodologies to discover new knowledge available in the published papers in order to support biologists in their strive toward understanding/analyzing biological data. One of the most effective and explored approaches to uncover this hidden knowledge is by mining the scientific literature [1–3], especially for finding gene–gene [4], gene–disease [1] and protein–protein [5] associations. However, usually, the number of inferred associations (especially in the approaches which retrieve also first-order associations) can be massive, thus making the analysis and interpretation of such information as complex (and probably more cryptic) as reading all the scientific papers the associations were extracted from. Therefore, issues such as validity, plausibility and feasibility of the inferred associations arise, and for this reason, methods [6] to filter out the obtained associations in order to distill the derived information and to propose it as significant scientific hypotheses have been investigated. A significant support to meet this goal comes from the massive publication on the Web of annotated chemical, genomic, clinical and other types of databases, which could provide evidence and validate specific

[*] Correspondence to: C. Spampinato, Department of Electrical, Electronics and Computer Engineering, University of Catania, 95125 Catania, Italy.

[†] E-mail: cspampin@dieei.unict.it

hypotheses. If, on one hand, experimental data may support the literature mining process, on the other hand, scientific literature may support the interpretation of such data, for example, list of up and down regulated genes.

Recent knowledge discovery systems, such as PathExpress [7], GenCLIP [8], CoPub [9], ENDEAVOUR [10], GeneWizard [6], G2D [11] and BioWizard [12], have exploited this integration between the literature and experimental data (biological, chemical, medical and drugs databases) for hypothesis generation and validation. However, discovering hidden knowledge in massive volumes of biological data necessitates unprecedented quantities of storage and data processing. In fact, with the amount of data growing and the increasing complexity of bioinformatics algorithms and tools, it is becoming highly demanding the introduction of advanced computational techniques to enable efficient knowledge discovery from data following the new wave of scientific development, that is, data-driven. However, it is rather complex, nowadays, for institutions to build up and sustain large computational infrastructures for data processing. A recent solution to support institutions is the cloud computing [13], which conveys computation and storage services as virtual resources via the Internet, thus representing an important alternative to ensure high performance data processing and easy management of complex tools in different areas of bioinformatics [14, 15], data and text mining [16]. As a consequence of this, the number of cloud resources is increasing at an accelerating pace, with service-based cloud environments provided by Microsoft[‡], Google[§], Amazon[¶], SGI[‖] and more, lending an unprecedented opportunity to evaluate the capabilities of the Cloud for sustainable and large-scale data processing in bioinformatics.

Generally, the cloud has been mainly used in the areas of economics, health and the entertainment industry, whereas its application in bioinformatics has been mainly oriented to the field of comparative genomics, for example, the Sanger Institutes fast matching and alignment algorithm to assemble full human genome [17], Cloud Burst [18] to map next-generation sequencing data [19] and Cloud Blast, a ‘clouded’ implementation of NCBI BLAST [20, 21].

The main contributions of this paper to the research on bioinformatics are the following:

- A review of the existing Cloud-based services, approaches and tools in bioinformatics.
- One of the first examples of Cloud-based knowledge discovery system, which generates biological hypotheses in the form of associations between biological entities by mining scientific papers and then validates the inferred associations against experimental data.

The remainder of the paper is as follows: Section 2 reviews the existing Cloud services and infrastructure that might be adopted in the bioinformatics research; Section 3, instead, describes the platform, a knowledge discovery tool that employs a natural language processing (NLP) based approach for mining the literature and deriving associations between biological entities, which are further validated against high-throughput data. Because the text and data mining processes are computationally expensive, Cloud Foundry, a platform for development, deployment and operation of cloud applications, is used. In Section 4, some experimental results are given, and Section 5, finally, discusses on the conclusions and draws some future lines for the research on bioinformatics by using the Cloud.

2. CLOUD TECHNOLOGIES IN BIOINFORMATICS

The rise of Cloud technologies is an incredible opportunity for bioinformatics in order to satisfy its needs of processing large amounts of heterogeneous data and of storing massive amount of data.

[‡] http://www.windowsazure.com/en-us/.
[§] https://cloud.google.com/.
[¶] http://aws.amazon.com/ec2/.
[‖] http://www.sgi.com/solutions/internet/.

DOI: 10.1002/cpe

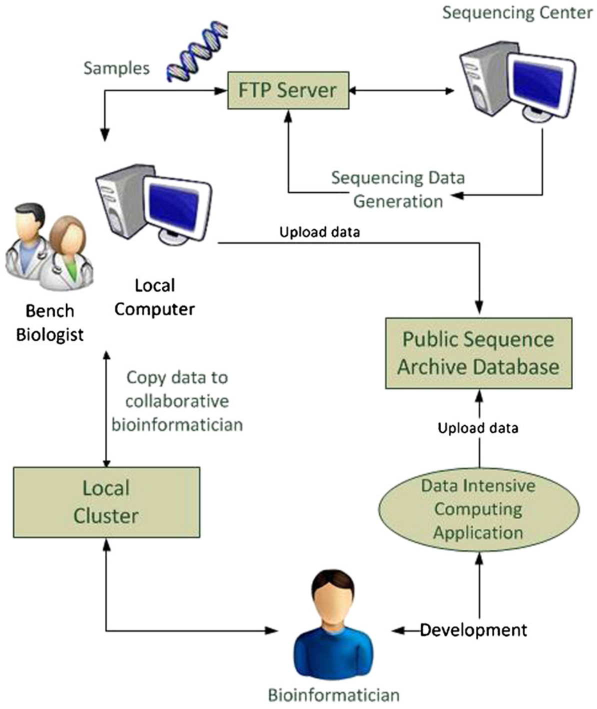

Sequencing Center
Samples
FTP Server
Sequencing Data Generation
Upload data
Local Computer
Bench Biologist
Public Sequence Archive Database
Copy data to collaborative bioinformatician
Upload data
Local Cluster
Data Intensive Computing Application
Development
Bioinformatician

Figure 1. Example of the typical workflow in bioinformatics.

This importance is witnessed by the ever growing number of bioinformatics applications (from DNA sequencing [22], to sequence alignment and similarity search [23], data mining [24], biological systems modeling [25] and knowledge discovery [16]) relying on Cloud services. However, cloud computing not only serves for large-scale computation but also provides a set of services that are changing radically the traditional way of doing research leading to a new era of bioinformatics [26]. For example, the typical workflow of DNA sequencing foresees that biologists design the experiments and send samples to sequencing centers, which make available raw data (through specific services, such as FTP, HTTP) to biologists, who have to download in their local institutions terabytes of data and, according to the research plans, publish these data in public databases. At the same, biologists copy the data into local machines for being used by bioinformaticians for the subsequent data analysis. Bioinformaticians, on the other hand, when process biologists' data have also to download data from public databases. Therefore, this typical flowchart (Figure 1) implies that huge quantities of data are moved several times from sites to sites, thus slowing down the analysis and the interpretation of the results.

The Cloud, instead, aims at creating an infrastructure (Figure 2) where sequencing centers store their data into the Cloud, public databases are built on the top of the infrastructure, biologists access these data directly from the Cloud and share what they need with bioinformaticians, who will develop large-scale applications directly on the Cloud whose results will be made available to the biologists for the interpretation. This new architecture will reduce the times data are transferred and also will allow laboratories and institutions to cut down the expenses to carry out experiments and data analysis.

In the next sections, the existing Cloud services and solutions for bioinformatics will be reviewed according to the Cloud's service model categorisation: Platform as a Service (PaaS), Software as a Service (SaaS) and Infrastructure as a Service (IaaS). The Data as a Service (DaaS), which is a Cloud service model that concentrates in distributing data on-demand instead of software applications or hardware resources and the Network as a Service (Naas), which is a Cloud service model that provides network infrastructure and resources on demand, are omitted because they are out of the scope of this work.

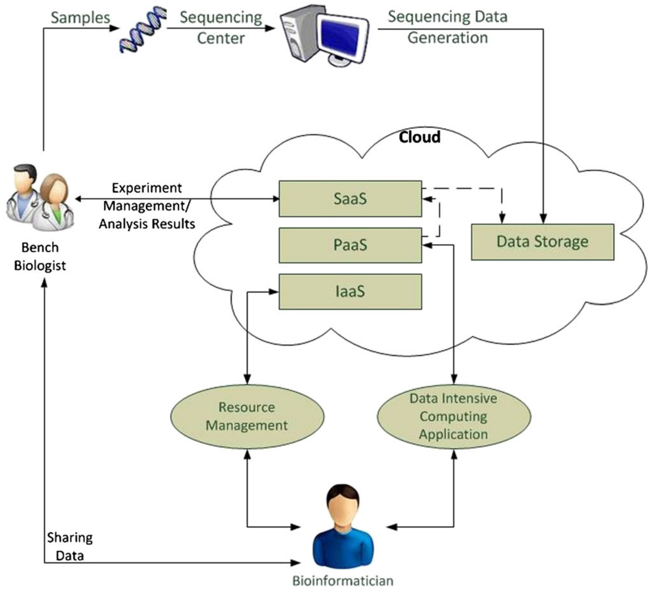


Figure 2. Changes on the bioinformatics workflow with the introduction of the Cloud computing.

2.1. *Platform as a Service*

Platform as a Service offers a development environment that allows users to create and run their applications using specific programming languages and frameworks available in the platform itself. Examples of PaaS environments are Google App-Engine[**] and Microsoft Azure[††]. However, to perform large-scale data analysis in bioinformatics, it is necessary that Cloud-based environments support the communication of parallel tasks in order to make full use of the available computation and storage resources. To address this need, most of the existing PaaS services are provided with an additional abstraction level implementing the map-reduce programming model [27]. The map-reduce computational paradigm divides the main application into many sub-applications, each executed or re-executed on a node of the Cloud infrastructure, and consists of two main steps. During the first step (map), the master node takes the input, divides it into smaller sub-problems and distributes them to worker nodes. The worker nodes process the smaller problems and pass the answer back to its master node. In the second step (reduce), the master node collects the answers to all the sub-problems and combines them to form the output. There exist many frameworks that implement the map-reduce paradigm and also provide jobs management functions for data-intensive computing such as Apache Hadoop[‡‡] or Microsoft's Dryad[§§]. Some of these frameworks are:

- Apache Hadoop framework, in which beyond the implementation of the map-reduce model, provides a distributed file system, the Hadoop Distributed File System [28], for effective and very low latency data storage on the worker nodes. In addition, there are many projects built on top of Hadoop such as Pig[¶¶], which is a high-level data-flow language and execution framework whose compiler produces sequences of Map/Reduce programs for execution within Hadoop, or Hive [29], which is a data warehouse framework built on top of Hadoop, developed at Facebook, used for ad hoc querying with an SQL type query language and also used for more complex analysis.

[**] http://code.google.com/appengine/.
[††] http://www.microsoft.com/windowsazure/.
[‡‡] http://hadoop.apache.org/.
[§§] http://research.microsoft.com/en-us/projects/dryad/.
[¶¶] http://pig.apache.org/.

DOI: 10.1002/cpe

- Microsoft Dryad, developed by Microsoft Research, allows developers to write parallel applications executing on the Cloud by modeling a directed acyclic graph (DAG). The DAG consists of a set of vertices describing the operations to be performed, which are distributed at runtime to different execution engines.
- Cloud MapReduce [30] is an implementation of MapReduce model [31] on top of the Amazon Cloud OS. Cloud MapReduce can be considered as an optimized version of the other MapReduce implementations, thanks to an architecture that ensures several advantages in terms of speed, scalability and simplicity.

Recently, PaaS frameworks have been applied with increasing interest to bioinformatics research as demonstrated by the quantity of works employing the map-reduce approach on the Cloud, mainly, for parallel large-scale data processing. In [32], Windows Azure was used in particular for data storage and as Virtual Machine (VM) hosting environment to conduct data mining for computational drug discovery. In [33], Dryad and Hadoop were used to host two bioinformatics applications: Expressed Sequence Tag [34] and Alu Sequencing [35]. An accurate performance evaluation has shown the advantages of the two frameworks with respect to traditional MPI implementations.

To the best of our knowledge, there exist only few applications exploiting Cloud-based map-reduce solutions to perform literature text mining for biological hypothesis generation. Nazareno *et al.* in [36] propose an 'ad-hoc' Cloud infrastructure for identifying molecular interactions by mining the scientific literature, whereas Lin *et al.* in [37] describe, more generally, how to process massively text data by using MapReduce. However, these solutions are at a very early stage, and their implementations cannot be used reliably for massive text processing because mainly of the lack of generalization. In fact, the deployment of these systems is too application-specific and often restricted to a single private Cloud environment.

Unlike text processing, Cloud-based map-reduce methods (mainly based on Hadoop) have been often used for processing high-throughput data analysis. Crossbow [38] proposes solutions executing on Hadoop for whole genome resequencing analysis and single nucleotide polymorphism genotyping from short reads. Contrail [39] uses Hadoop for 'de novo' assembly from short sequencing reads, whereas Myrna [40] proposes a method for calculating differential gene expression from large RNA-seq data sets. On clusters, Myrna uses Hadoop, whereas in the Cloud, it uses Amazon Elastic MapReduce[||||].

Analogously, a few Cloud-based methods for microarray data mining have been proposed as in [41] where the authors developed a MapReduce framework on Hadoop for mining association rules from microarray gene expression datasets. Delmerico *et al.* in [42] provide an extensive performance evaluation of clusters and Hadoop based solutions for computing genes correlations by processing microarray data. The authors state that although the performance of the existing approaches for identifying such correlations are generally improved on clusters, storage, hardware and network (mainly) limitations restrict their scalability, on the contrary of Hadoop, which, instead, provides a significantly better scalability.

However, two are the main downsides of the MapReduce solutions: first of all, the map/reduce frameworks require re-writing most of the existing applications, which, for several reasons, is not appreciated by bioinformaticians and biologists. Second, the current implementations of map/reduce paradigm employ some overly simple mechanisms; for example, the job scheduling is often not (well) supported, thus affecting the tools' performance.

2.2. *Software as a Service*

Software as a Service is a Cloud solution, which coveys software applications through the Internet, and it is commonly used in bioinformatics to support remote access to available tools. For SaaS there is no client side software requirement for the user: the services are reachable through an access point like a web portal or a visualization tool. The main advantage of SaaS is that it enables large-scale data analysis over the web, thus eliminating the need for local installation of a large

[||||] http://aws.amazon.com/elasticmapreduce/.

variety of software tools and also providing up-to-date Cloud-based services for bioinformatic data analysis.

An interesting example of SaaS in bioinformatics is EasyGenomics[***], a key enabling platform providing streamlined bioinformatics services. Basically, most of the available tools implement the map/reduce paradigm for parallelization and scalability, but make it trasparent for the end-users who have to call only the service with no knowledge about the underlying software and hardware infrastructure. Relevant examples are: CloudAligner [43], a full-featured Hadoop MapReduce-based tool for sequence mapping, and CloudBurst [18], an open source optimized tool for mapping next-generation sequence data to the human genome with MapReduce.

However, one of the major requirements of SaaS services in bioinformatics is the interoperability between multiple Cloud systems. The main difficulties to address this need are that the mechanisms for service publishing, searching and subscription are not well-established and the existing technologies (WSDL, UDDI) do not describe sufficiently the semantics of such services [44]. For this reason, the current trend is toward a metadata ontology for service description [45].

2.3. *Infrastructure as a Service*

The IaaS layer aims at offering computer infrastructures, virtualised resources, storage, networks and other fundamental computing resources via self-services to the user. The challenge introduced by bioinformatics on IaaS regards the enhancement of flexibility of Cloud platform for resource management in order to satisfy user needs. The most appropriate approach to ensure such flexibility is via virtualisation that mainly involves either the generation of multiple virtual machine instances to partition the physical resources or multi-tenancy techniques, which enable users to share application instances and treat them as independent ones.

However, currently, the most employed approach is the creation of suitable virtual machine instances according to user requirements. This is a non-trivial task in bioinformatics because of dependence and version matching issues arising when dealing with bioinformatics tools.

Amazon EC2 [46] represents an example of such a service, and it offers a variety of VM images provided with a good variety of bioinformatics tools. Other important examples are Cloud BioLinux [47] and CloVR [48]. The former is a publicly accessible virtual machine for high performance bioinformatics computing. The latter, instead, is a portable virtual machine for automated sequence analysis, and its performance are discussed in [49, 50]

The main limitation of the current IaaS services is that VM creation, update and sharing are too ad hoc and tailored to the specific needs of bioinformaticians and biologists, who, basically, have to create VMs from the beginning. On-demand packaging mechanisms are recently under investigation to allow an automatic creation of virtual machine images provided with all the needed and up-to-date tools with all the dependencies solved.

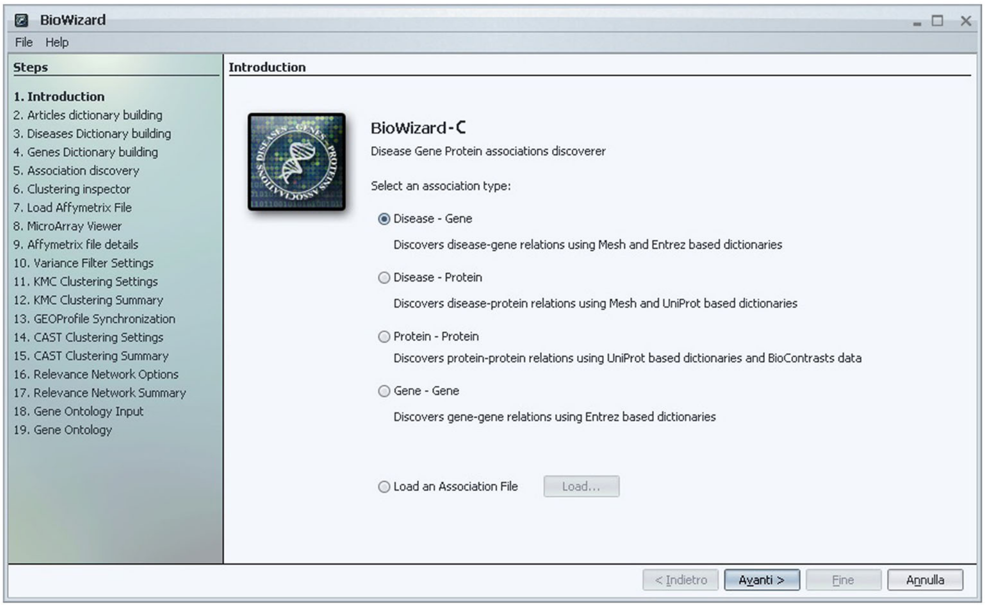
In the next section, the knowledge discovery tool for biological hypothesis generation exploiting an IaaS Cloud service is described as an example on how to execute large-scale data analysis tool on the Cloud.

3. A CLOUD-BASED PLATFORM FOR BIOLOGICAL HYPOTHESES GENERATION

The developed application allows users to produce new biological hypotheses through an intuitive and guided interface without requiring knowledge of text-mining and data mining methods. It retrieves automatically associations between biological entities (gene–disease , gene–gene, protein–protein, protein–disease, gene–miRNA and miRNA–disease) by mining Pubmed abstracts and Oxford Journals full papers and validates them against biological data. Figure 3 shows all the types of associations (excluded the ones involving miRNA as they are considered as a special case of the ones involving genes) supported by our tool.

[***] www.easygenomics.com.

DOI: 10.1002/cpe



1. Introduction
2. Articles dictionary building
3. Diseases Dictionary building
4. Genes Dictionary building
5. Association discovery
6. Clustering inspector
7. Load Affymetrix File
8. MicroArray Viewer
9. Affymetrix file details
10. Variance Filter Settings
11. KMC Clustering Settings
12. KMC Clustering Summary
13. GEOProfile Synchronization
14. CAST Clustering Settings
15. CAST Clustering Summary
16. Relevance Network Options
17. Relevance Network Summary
18. Gene Ontology Input
19. Gene Ontology

BioWizard-C

Disease Gene Protein associations discoverer

Select an association type:

● Disease - Gene

Discovers disease-gene relations using Mesh and Entrez based dictionaries

○ Disease - Protein

Discovers disease-protein relations using Mesh and UniProt based dictionaries

○ Protein - Protein

Discovers protein-protein relations using UniProt based dictionaries and BioContrasts data

○ Gene - Gene

Discovers gene-gene relations using Entrez based dictionaries

○ Load an Association File Load...

Figure 3. The biological entities associations supported.

As the association retrieval and validation involves large-scale text and data mining procedures, we have used Cloud Foundry[†††], a Cloud computing PaaS and IaaS solution developed by VMware[‡‡‡].

In detail, the main steps performed by the proposed system to generate and validate scientific hypothesis are the following:

- Document retrieval and dictionaries building to create the test set from which the associations are extracted.
- An NLP based approach that, starting from the above set, parses the documents and extracts the associations between the terms of the dictionaries.
- Validation of the derived associations using high-throughput data (e.g. microarray data for gene–disease and miRNA–disease associations, BioConstrast[§§§] and STRING [¶¶¶] for associations involving proteins).
- Execution of the text and data mining algorithms on the Cloud.

Figure 4 recaps the resources and modules used by the system, while Figure 5 shows how each module is distributed in the Cloud. In the next subsections, each module is described in detail.

3.1. Text mining module for hypothesis generation

The text mining approach implemented is based on a natural language processing method, which parses fully syntax and semantics of the retrieved papers. The developed application infers an association between two biological entities $T_1 - T_2$ when it finds a meaningful triple (*noun-verb-adjective*) with *noun* and *adjective* being biological entities (taken from the biological terms vocabulary) and *verb* being a verb, which significantly correlates the two terms (e.g. T1 activates T2). In previous works [6, 12], we adopted co-occurrences processing for deriving associations that, unlike the one herein proposed, produce many noisy associations (high recall, but low precision) making the subsequent validation very time consuming and sometime also useless.

[†††] http://www.cloudfoundry.com/.
[‡‡‡] http://www.vmware.com.
[§§§] http://biocontrasts.biopathway.org/.
[¶¶¶] http://string-db.org/.

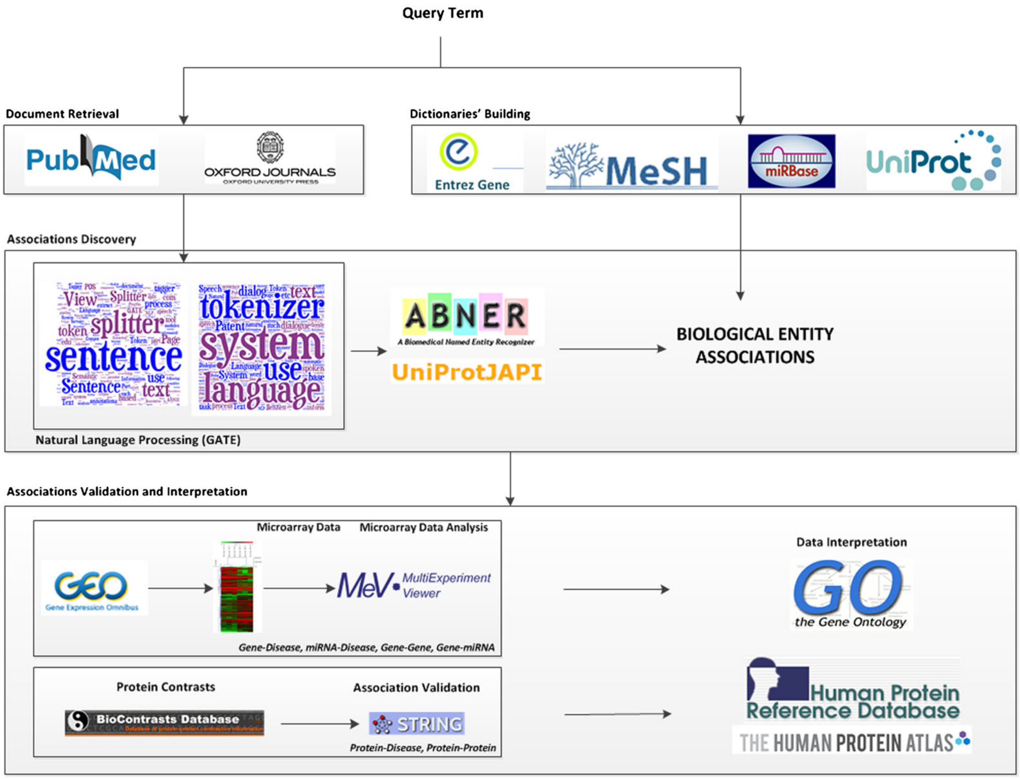


Figure 4. Outline of the modules and resources used.

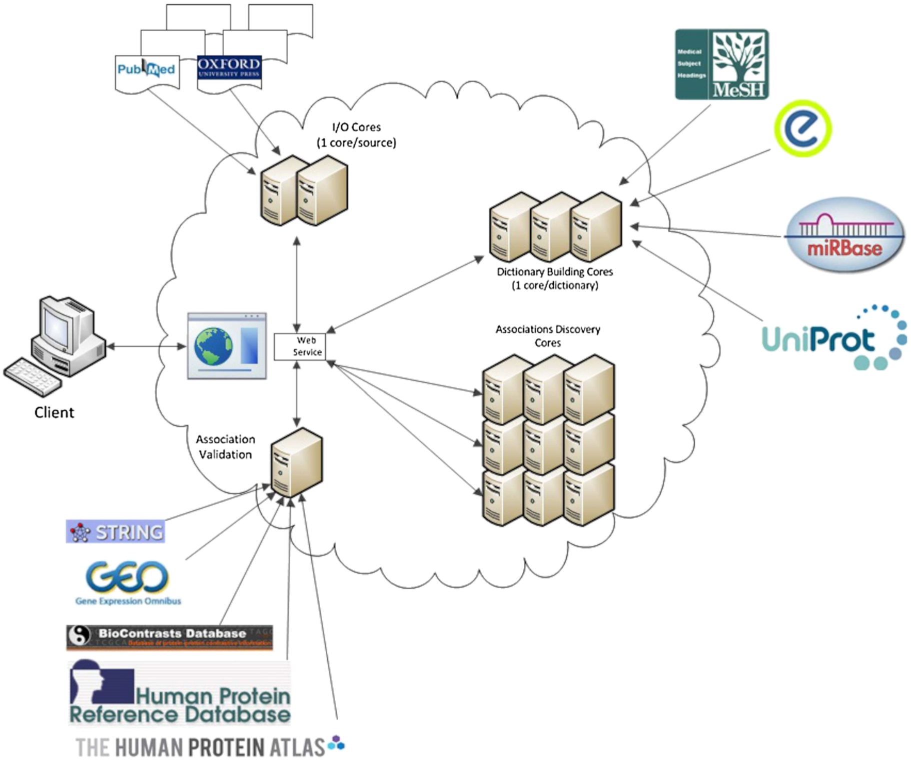
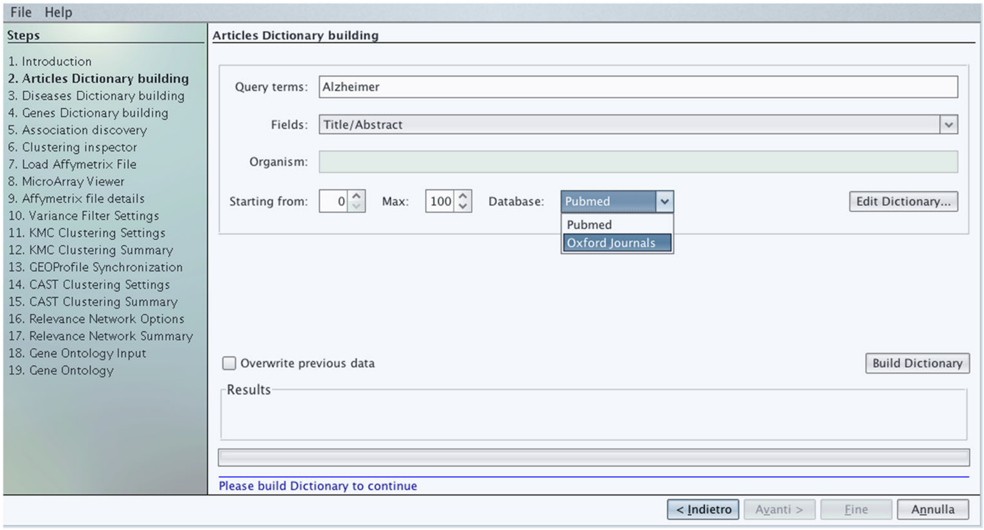


Figure 5. Top-level architecture. The entry point of the platform is the Web-Service, which also has the role of controlling the workflow and distributing the workload of the whole process.


File Help
Steps
1. Introduction
2. Articles Dictionary building
3. Diseases Dictionary building
4. Genes Dictionary building
5. Association discovery
6. Clustering inspector
7. Load Affymetrix File
8. MicroArray Viewer
9. Affymetrix file details
10. Variance Filter Settings
11. KMC Clustering Settings
12. KMC Clustering Summary
13. GEOProfile Synchronization
14. CAST Clustering Settings
15. CAST Clustering Summary
16. Relevance Network Options
17. Relevance Network Summary
18. Gene Ontology Input
19. Gene Ontology
Pubmed
Oxford Journals
Edit Dictionary...
Build Dictionary
< Indietro
Avanti >
Fine
Annulla

Articles Dictionary building

Query terms: Alzheimer

Fields: Title/Abstract

Organism:

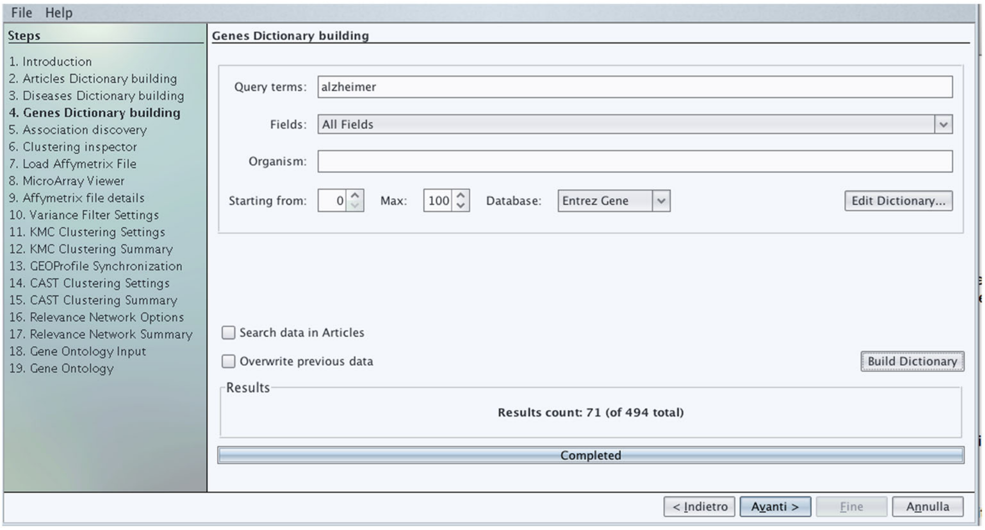
Starting from: 0 Max: 100 Database: Pubmed

□ Overwrite previous data

Results

Please build Dictionary to continue

Figure 6. GUI for document retrieval.


File Help
Steps
1. Introduction
2. Articles Dictionary building
3. Diseases Dictionary building
4. Genes Dictionary building
5. Association discovery
6. Clustering inspector
7. Load Affymetrix File
8. MicroArray Viewer
9. Affymetrix file details
10. Variance Filter Settings
11. KMC Clustering Settings
12. KMC Clustering Summary
13. GEOProfile Synchronization
14. CAST Clustering Settings
15. CAST Clustering Summary
16. Relevance Network Options
17. Relevance Network Summary
18. Gene Ontology Input
19. Gene Ontology
Edit Dictionary...
Build Dictionary
< Indietro
Avanti >
Fine
Annulla

Genes Dictionary building

Query terms: alzheimer

Fields: All Fields

Organism:

Starting from: 0 Max: 100 Database: Entrez Gene

□ Search data in Articles

□ Overwrite previous data

Results

Results count: 71 (of 494 total)

Completed

Figure 7. GUI for gene dictionary building.

The proposed approach consists of four main steps:

1. *Document Retrieval and Dictionaries Building*. As biological entities identified by mining only abstracts are underestimated because of abstracts' concise nature, the proposed approach uses a set of full text articles retrieved from the Oxford Journal system using a biological entity name as query term. At the same time, we use Entrez Gene [51], Mesh [52], miRBase [53] and UniProt* for building the dictionaries, respectively, for genes, diseases, miRNA and proteins. Such dictionaries are the basis of our text mining approach. Figures 6 and 7 show, respectively, the GUIs for document retrieval either from Pubmed or from Oxford Journal and for gene dictionary building.
2. *Natural Language Processing for Parsing Full Text*. In parallel to dictionaries building, the retrieved papers are parsed by using the A Nearly New Information Extraction System module included in GATE [54]. The text parsing consists of the following modules: (i) *Text Tokenizing*

*www.uniprot.org.

DOI: 10.1002/cpe

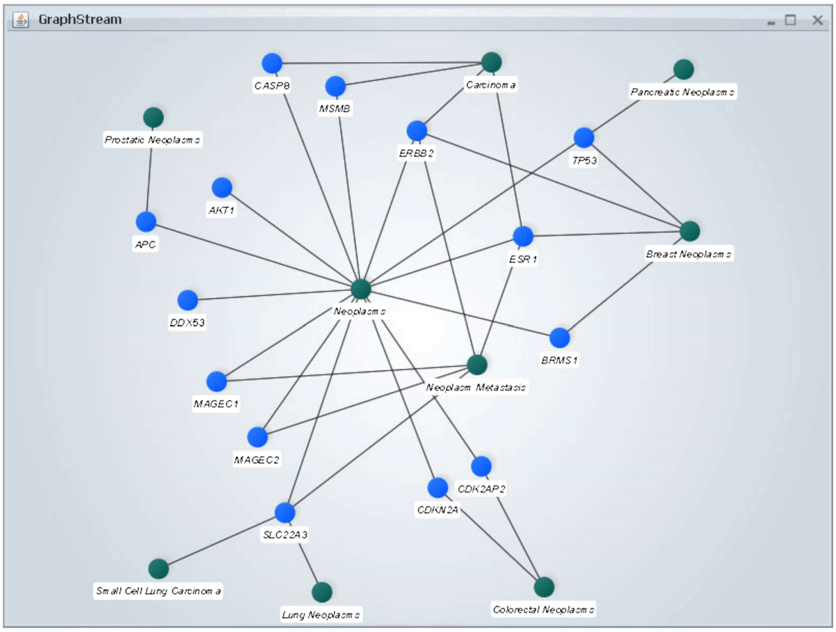


Figure 8. Multiple disease-gene associations. Diseases are green-coded, while genes are blue-coded.

to break the text into tokens, which provides useful information such as token category (proper noun, verb, adjective), token length and orthography (hyphenation, capitalization, word breaks) and (ii) *Sentence Splitter* to split tokens into sentences.

3. A *Named Entity Recognition* (NER) module, which improves the dictionaries' creation. In fact by only using the terms of standard vocabularies, it may happen that no association is derived because of the dissimilarities between the vocabularies' terms and the terms extracted from parsing full papers and abstracts. We used ABNER [55] to tag genes and diseases within a sentence, while UniProtJAPI[†] for proteins.
4. *Associations discovery between meaningful terms*. For each sentence, all the triples (*noun*, *verb*, *adjective*) are detected, and then only those containing valid biological entity names (the ones in the dictionaries and validated by the NER) and consistent verbs (previously defined) are considered as hypotheses, which are subsequently validated with experimental data.

The tool, moreover, allows the users to re-use the inferred associations in different mining processes in order to achieve multiple first-order associations: that is, if, for example, in a mining process, we obtain an association between the entity E_1 and the entity E_2, and in another mining process, we infer an association between the entity E_1 and the entity E_3, then a graph is created with a connection between E_2 and E_3 through the entity E_1. Figure 8 shows the case of multiple associations between diseases and genes, whereas Figure 9 shows multiple associations between proteins.

3.2. Validation of hypothesis generation against experimental data

For the validation of the generated hypotheses our system uses (i) microarray data retrieved from the GEO database for validating associations involving genes or miRNA and (ii) BioContrasts and STRING 9.0 for the associations involving proteins.

In the former case, once a microarray is selected, the tool starts data analysis in order to build the relative gene relevance network (GRN) (i.e. a list of relevant genes for the given disease) containing the gene of the gene–disease association to be validated. The genes of the GRN are then re-codified using DAVID [56]. The microarray analysis modules are based on the Java classes

[†] www.ebi.ac.uk/uniprot/remotingAPI/.

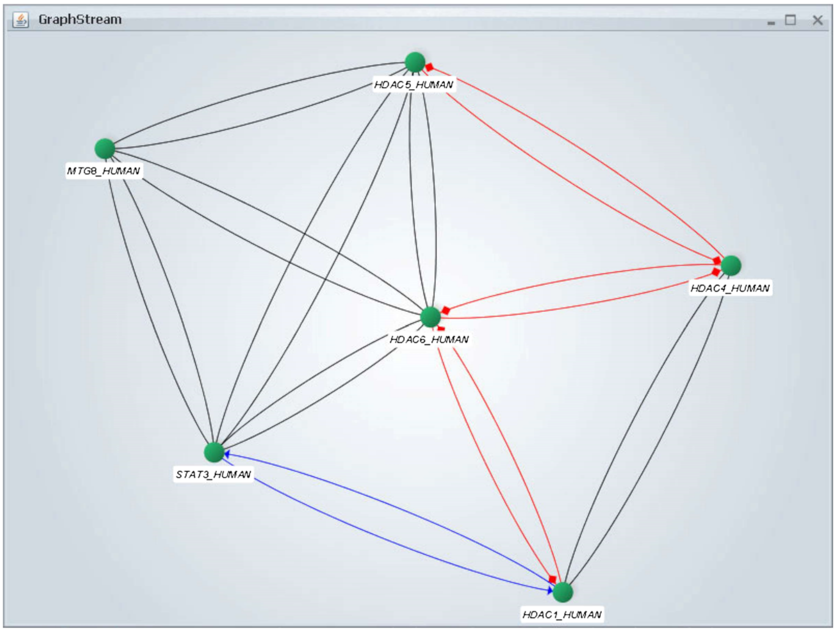


Figure 9. Multiple protein–protein associations. The associations are divided into positive ones, and contrastive ones and are depicted with different colors in the graph.

from the MultiExperiment Viewer software [57]. The first step of the data analysis is to apply Hierarchical Clustering to the microarray data to obtain clusters of genes. Then the cluster containing the gene under examination is selected, and a GRN ('Main GRN') is derived from it by applying Cluster Affinity Search [58]. Because, often, the GRN may not contain a sufficient number of genes because of several factors (ranging from the clustering settings to the microarray data quality), we iterate the procedure on building GRN for all the genes that make part of the 'Main GRN'. In detail, for each gene of the GRN, the microarray datasets that contain it are downloaded from the GEO database, and then, according to the described procedure, another GRN that will be connected to the main GRN (i.e. the one containing the disease under investigation) is built. Finally, the genes of this extended GRN are used to query the Gene Ontology database in order to investigate the biological meaning of such genes with respect to the given disease.

It is understandable that this validation process may not be executed on a single machine as it involves, first, a computational intensive text mining procedure and, second, a recursive data mining phase (several clustering steps executed on matrices with thousands of elements) for building the extended GRN.

The validation of protein–protein associations is, instead, performed by using BioContrasts and STRING 9.0, which is a database of known and predicted protein interactions that makes use of genomic and high-throughput experiments data. In detail, each identified protein–protein association is first passed for validation to BioContrasts which identifies contrasts between proteins by identifying patterns in the form of 'A but not B' in MEDLINE abstracts. If there is not a contrast between the two proteins, the final validation step is to check if the association exists in STRING 9.0, which also provides a set of further proteins involved in the association. Similar to the case of gene–disease association, we use the Gene Ontology to investigate the biological meaning of the proteins previously identified.

Protein–disease associations are instead automatically validated against the Human Protein Atlas[‡] and the Human Protein Reference Database[§].

[‡] www.proteinatlas.org.
[§] www.hprd.org.

3.3. Data analysis on the cloud

The proposed tool is designed as a standalone desktop application that interfaces with a web service located in the cloud. In this work, we exploited the PaaS and IaaS cloud service models offered by CloudFoundry.

The web service is not only the entry point to the platform but also plays a key role in the coordination of the workflow of the whole process. In fact, as shown in Figure 5, the web service is interconnected to all the other modules of the system (I/O, Dictionary building, NLP processing, Association Discovery and Microarray processing) interpreting the user's requests, sychronizing the underlying processes and returning the results to the user. For data management, a MySQL database combined with the Hibernate library was chosen. The cloud execution requires to remove the interface part; thus, we only execute the application engine giving the needed parameters/settings in an XML file.

As soon as the platform is set up, the application launches. The execution time is monitored through the Cloud Foundry command-line interface, but the running application's standard output can be checked. When the program's execution ends, all the retrieved and produced data can be found in the database. These data can also be used in order to derive performance parameters (such as recall, efficiency metrics etc.).

4. EXPERIMENTAL RESULTS

Knowledge discovery systems often produce results that are based on a true scientific basis but does not always hold true. This means that performance analysis of knowledge discovery systems is quite approximate as the definition of 'discovery' is still controversial [59].

The only reliable way, adopted so far, to examine accurately the performance of a knowledge discovery system is to use gold standard annotations and then compare the obtained associations against them.The tool described in this paper allows us to extract different types of associations, but we tested only the case of gene-disease associations as the main goal is to see how the system carried out large scale analysis on the cloud. As a gold standard for gene-disease associations (Table I), we used the list of 110 diseases with 1318 associations to genes described in [60]. For consistency, only the diseases with a minimum count of 100 retrieved documents were considered. For the totality of the diseases in the gold standard dataset, the application retrieved 220782 papers.

The performance evaluation of the system was divided in two main sections: (i) comparison, in terms of efficiency and of valid (i.e. that have evidence in the used gold standard associations) retrieved associations, between the NLP method and the co-occurrence-based approach [6] using an optimized sequential (1 core) version of the application [12] both in the cloud and on a local computer; and (ii) assessment of how the cloud implementation scales when the number of processor cores available for processing increases.

For the former, the systems used were an Intel Core 2 Duo processor, running at 2 GHz, with 2 GB of RAM for the local configuration and an equivalent 2-core at 2 GHz with 2 GB of RAM for the cloud one. Being a sequential implementation, only one core of the two available was used by the tool (and the application was set to run at real-time priority), while the one remaining was dedicated to the operating system. For the latter, the platform was tested on a 64 core deployment, running at 2 GHz, divided in 16 systems with 4 cores and 4 GB of RAM each, using multiple application instances and following the architecture diagram shown in Figure 5. From the 64 available cores, 3 were assigned for dictionary building, 2 cores for input operations (PubMed, Oxford Journal), 1 core for the web service for each user (8 cores were preallocated) and 1 core for the processing of the microarrays, leaving, 50 processor cores for the documents' processing. The scalability (i.e. how

Table I. Dataset used for our experimental evaluation.

Number of diseases	110
Number of papers examined	220782
Number of relevant associations	1318

DOI: 10.1002/cpe

the system behaves when increasing the workload and how adding more hardware resources affect the performance of it) of the platform in the cloud was assessed by testing the application using 10, 20, 30, 40 and 50 processors. The time needed to complete the processing of the papers in each setting is shown in Table II.

It is clear how the cloud boosts the performance of the application. In fact, comparing it against the locally running instance, we observed a gain in efficiency of about 25% when the NLP module was used (1715 min against 2281 min) and about 26% when the co-occurrence module was used (3207 min against 4307 min), when the sequential implementation was used on the cloud. This increase in performance was achieved because the file transfers and I/O operations were carried out between entities in the cloud (i.e. a very much broader bandwidth in the file transfer operations both among the data sources and less memory and processor overhead for the operating system).

When more processors were available, the platform demonstrated that it can scale efficiently. In fact, Table II shows that the performance of the platform scales well when the number of cores increases, achieving an efficiency enhancement of about $10X$ for both the NLP and the co-occurrence methods when 50 cores were used w.r.t. when only 2 cores were used. While such amount of performance gain seems relatively small compared to the ideal one ($25X$), it does not consider the case when multiple users interact with the system simultaneously. Additionally, the

Table II. Performance in terms of time needed for each setting to process the whole stack of documents.

	Time in Minutes	
	NLP	Co-occurrence
Sequential (local)	2281	4307
Sequential (cloud)	1715	3207
Parallel 2 cores	1321	2531
Parallel 10 cores	539	1033
Parallel 20 cores	287	566
Parallel 30 cores	202	391
Parallel 40 cores	158	305
Parallel 50 cores	132	255

NLP, natural language processing.

Table III. Experimental results of a subset of the dataset in terms of valid associations/retrieved associations per disease and recall.

			Co-occurrence					NLP				
Disease	RA	DA	TP	PA	RC	TL	TC	TP	PA	RC	TL	TC
Anemia	5810	33	20	18	0.61	10	7	26	21	0.79	4	3
Breast Cancer	10000	24	15	13	0.63	219	165	21	16	0.88	62	49
Diabetes Melitus	10000	38	13	19	0.34	190	143	28	28	0.74	123	91
Hypertension	10000	13	7	2	0.54	202	142	9	3	0.69	98	73
Leukemia	10000	39	20	16	0.51	233	178	24	26	0.62	146	113
Liver Cancer	8175	10	4	4	0.40	177	125	6	7	0.60	114	82
Lymphoma	10000	10	6	7	0.60	225	159	6	9	0.60	133	103
Melanoma	7931	6	3	7	0.50	181	143	4	4	0.67	113	84
Obesity	10000	24	15	15	0.63	145	114	17	21	0.71	110	86
Prostate Cancer	7652	14	5	8	0.36	120	85	11	8	0.79	76	58

RA is the number of the retrieved papers from Oxford Journal when the corresponding diseases name was queried, DA denotes the number of the existing associations in the gold standard dataset. TP is true positives, that is, the number of gene–disease associations that were both in the gold standard dataset and the applications output, and PA denotes the number of associations that did not exist in the gold standard dataset, but the tool marked them as valid (possible associations). RC represents the recall for the corresponding disease. TL and TC are the time, in minutes, needed by the sequential version to complete the processing running, respectively, on the local computer and the cloud one.
NLP, natural language processing.

platform scales approximately linearly, meaning that it can handle very large amounts of data, given the necessary resources, in a predictable manner.

Table III shows the achieved results of the sequential implementation in terms of valid associations/retrieved associations and recall (defined as true positives over the whole association in the gold standard dataset for a given disease).

Under this aspect, both the solutions performed identically because the algorithm's logic was identical. Observing the results, it can be deduced that the NLP based approach achieved consistently better recall values than the co-occurrence based method. While the PA values (Table III) could be considered as false positive values (i.e. associations not in the gold standard dataset but identified by the tool), it represents possible hidden 'knowledge' that must be further investigated.

From the same table, we can also observe that the sequential version deployed on the cloud offers a net performance boost, when the NLP implementation was used instead of the co-occurrence method, with an average value of 89%.

5. CONCLUSIONS

In this paper, we have presented an open source, Cloud-based platform that assists life science researchers in knowledge discovery. In particular, by integrating text mining methods on scientific documents found in PubMed and Oxford Journal with high-throughput data, the proposed tool is able to identify and validate possible associations between genes, diseases, proteins and miRNA that may be involved in biological processes.

Furthermore, this work shows how parsing full text papers and porting an application on the cloud increases its efficiency and effectiveness. In addition, to accomplish comparable performance with other parallel computing architectures (GRID, MPI etc.), a more in-depth modification of the program code would be necessary. As a consequence, applications in many scientific fields that make use of large volumes of data (bioinformatics, medicine, astronomy etc.) can now expand their datasets, obtaining better results both in terms of efficiency and accuracy.

In the near future, we aim to publish the tool herein presented as a free SaaS service to make it available for other users who may want to integrate it in their platform, although the core web-service (hence not the GUI) it is based on is available and can be obtained by contacting the corresponding author. Future development will be focused on integrating multimedia retrieval methods [61] that could be used for extracting semantic information from images contained in the scientific papers under examination of the application in order to increase even more the number of the discovered associations.

ACKNOWLEDGEMENT

We would like to thank Sebastiano Milardo for his contribution in the development of the platform herein presented.

REFERENCES

1. Özgür A, Vu T, Erkan G, Radev D. Identifying gene-disease associations using centrality on a literature mined gene-interaction network. *Bioinformatics* 2008; **24**(13):i277–i285.
2. Ananiadou S, Kell DB, Tsujii Ji. Text mining and its potential applications in systems biology. *Trends in Biotechnology* 2006; **24**(12):571–579. DOI: 10.1016/j.tibtech.2006.10.002.
3. Jensen LJ, Saric J, Bork P. Literature mining for the biologist: from information retrieval to biological discovery. *Nature Reviews Genetics* 2006; **7**(2):119–129.
4. Liu Y, Navathe S, Civera J, Dasigi V, Ram A, Ciliax B, Dingledine R. Text mining biomedical literature for discovering gene-to-gene relationships: a comparative study of algorithms. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 2005; **2**(1):62–76.
5. Von Mering C, Jensen L, Snel B, Hooper S, Krupp M, Foglierini M, Jouffre N, Huynen M, Bork P. String: known and predicted protein–protein associations, integrated and transferred across organisms. *Nucleic Acids Research* 2005; **33**(suppl 1):D433–D437.
6. Faro A, Giordano D, Spampinato C. Combining literature text mining with microarray data: advances for system biology modeling. *Briefings in Bioinformatics* 2012; **13**:61–82.

7. Ekins S, Nikolsky Y, Bugrim A, Kirillov E, Nikolskaya T. Pathway mapping tools for analysis of high content data. *Methods in Molecular Biology (Clifton, N.J.)* 2007; **356**:319–350.
8. Wu J, Mao X, Cai T, Luo J, Wei L. KOBAS server: a web-based platform for automated annotation and pathway identification. *Nucleic Acids Research* 2006; **34**:W720–724.
9. Frijters R, Heupers B, van Beek P, Bouwhuis M, van Schaik R, de Vlieg J, Polman J, Alkema W. CoPub: a literature-based keyword enrichment tool for microarray data analysis. *Nucleic Acids Research* 2008; **36**:W406–410.
10. Tranchevent L, Barriot R, Yu S, Van Vooren S, Van Loo P, Coessens B, De Moor B, Aerts S, Moreau Y. Endeavour update: a web resource for gene prioritization in multiple species. *Nucleic Acids Research* 2008; **36** (suppl 2):W377–W384.
11. Perez-Iratxeta C, Wjst M, Bork P, Andrade MA. G2D: a tool for mining genes associated with disease. *BMC Genetics* 2005; **6**(1):1–9.
12. Spampinato C, Giordano D, Kavasidis I, Milardo S. Biowizard: Discovering and validating associations between biological entities by integrated analysis of scientific literature and experimental data. *2012 25th International Symposium on Computer-based medical systems (CBMS),* IEEE, Rome, Italy, 20–22 June 2012; 1–6.
13. Mell P, Grance T. The NIST definition of cloud computing. *NIST Special Publication* 2011; **800**:1–3.
14. Bateman A, Wood M. Cloud computing. *Bioinformatics* 2009; **25**(12):1475–1475.
15. Dai L, Gao X, Guo Y, Xiao J, Zhang Z. Bioinformatics clouds for big data manipulation. *Biology Direct* 2012; **7**(1):1–7.
16. Hey A, Tansley S, Tolle K. *The Fourth Paradigm: Data-Intensive Scientific Discovery*. Microsoft Research Redmond: WA, 2009.
17. Li H, Durbin R. Fast and accurate short read alignment with burrows–wheeler transform. *Bioinformatics* 2009; **25**(14):1754–1760.
18. Schatz M. Cloudburst: highly sensitive read mapping with MapReduce. *Bioinformatics* 2009; **25**(11):1363–1369.
19. Shendure J, Ji H. Next-generation dna sequencing. *Nature Biotechnology* 2008; **26**(10):1135–1145.
20. Matsunaga A, Tsugawa M, Fortes J. Cloudblast: combining MapReduce and virtualization on distributed resources for bioinformatics applications. *IEEE Fourth International Conference on Escience, 2008. eScience'08,* IEEE, Indianapolis, Indiana, USA, 7–12 December 2008; 222–229.
21. Johnson M, Zaretskaya I, Raytselis Y, Merezhuk Y, McGinnis S, Madden T. NCBI blast: a better web interface. *Nucleic Acids Research* 2008; **36**(suppl 2):W5–W9.
22. Stein L. The case for cloud computing in genome informatics. *Genome Biology* 2010; **11**(5):207.
23. Li H, Homer N. A survey of sequence alignment algorithms for next-generation sequencing. *Briefings in Bioinformatics* 2010; **11**(5):473–483.
24. Grossman R, Gu Y. Data mining using high performance data clouds: experimental studies using sector and sphere. *Proceeding of the 14th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining,* ACM, Las Vegas, NV, USA, 24–27 August 2008; 920–927.
25. Aldinucci M, Torquati M, Spampinato C, Drocco M, Misale C, Calcagno C, Coppo M. Parallel stochastic systems biology in the cloud. *Briefings in Bioinformatics* 2013. DOI: 10.1093/bib/bbt040.
26. Sun X, Fan L, Yan L, Kong L, Ding Y, Guo C, Sun W. Deliver bioinformatics services in public cloud: challenges and research framework. *2011 IEEE 8th International Conference on E-Business Engineering (ICEBE)*, Beijing, China, 19–21 October 2011; 352–357.
27. Dean J, Ghemawat S. MapReduce: simplified data processing on large clusters. *Communications of the ACM* 2008; **51**(1):107–113. DOI: 10.1145/1327452.1327492.
28. Shvachko K, Kuang H, Radia S, Chansler R. The hadoop distributed file system. *Proceedings of the 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST) (MSST '10)*, 2010; 1–10.
29. Thusoo A, Sarma J, Jain N, Shao Z, Chakka P, Anthony S, Liu H, Wyckoff P, Murthy R. Hive: a warehousing solution over a map-Reduce framework. *Proceedings of the VLDB Endowment* 2009; **2**(2):1626–1629.
30. Liu H, Orban D. Cloud MapReduce: a MapReduce implementation on top of a cloud operating system. *2011 11th IEEE/ACM International Symposium on Cluster, cloud and Grid Computing (CCGrid),* IEEE, Newport Beach, CA, USA, 2011; 464–474.
31. Dean J, Ghemawat S. MapReduce: simplified data processing on large clusters. *Communications of the ACM* 2008; **51**(1):107–113.
32. Watson P, Leahy D, Hiden H, Woodman S, BerryLiu J. An azure science cloud for drug discovery. *Microsoft External Research Symposium*, Redmond, Washington, USA, 30–31 March 2009.
33. Ekanayake J, Gunarathne T, Qiu J. Cloud technologies for bioinformatics applications. *IEEE Transactions on Parallel and Distributed Systems* 2011; **22**(6):998–1011.
34. Adams MD, Kelley JM, Gocayne JD, Dubnick M, Polymeropoulos MH, Xiao H, Merril CR, Wu A, Olde B, Moreno RF. Complementary DNA sequencing: expressed sequence tags and human genome project. *Science* 1991; **252**(5013):1651–1656.
35. Häsler J, Strub K. Alu elements as regulators of gene expression. *Nucleic Acids Research* 2006; **34**(19):5491–5497.
36. Nazareno F, Lee K, Cho W. Mining molecular interactions from scientific literature using cloud computing. *2010 IEEE International Conference on Bioinformatics and Biomedicine Workshops (BIBMW),* IEEE, Hong Kong, China, 18–21 December 2010; 864–865.
37. Lin J, Dyer C. Data-intensive text processing with MapReduce. *Synthesis Lectures on Human Language Technologies* 2010; **3**(1):1–177.

38. Langmead B, Schatz M, Lin J, Pop M, Salzberg S. Searching for SNPS with cloud computing. *Genome Biology* 2009; **10**(11):1–10.
39. Contrail project home page (Contrail: Assembly of Large Genomes using Cloud Computing). http://sourceforge.net/apps/mediawiki/contrail-bio/index.php?title=Contrail.
40. Langmead B, Hansen K, Leek J. Cloud-scale RNA-sequencing differential expression analysis with Myrna. *Genome Biology* 2010; **11**(8):1–11.
41. Rezaul K, Golam B, Jeong B, Choi HJ. Cloud technology for mining association rules in microarray gene expression datasets. *International Journal of Database Theory & Application* 2012; **5**(2):61–74.
42. Delmerico J, Byres N, Brunn A, Jones M, Gall S, Chandlery V. Comparing the performance of clusters, Hadoop, and Active Disks on microarray correlation computations. *2009 International Conference on High Performance Computing (HIPS),* IEEE, 2009; 378–387.
43. Nguyen T, Shi W, Ruden D. CloudAligner: A fast and full-featured MapReduce based tool for sequence mapping. *BMC Research Notes* 2011; **4**:171, pp. 1–7.
44. Splanchnomega K, Vera K, Sheath A, Miller J. Adding semantics to web services standards, Proceedings of the International Conference on Web Services, New York: USA, 2003; 395–401.
45. Roman D, Killer U, Clause H, de Bruin J, Larva R, Stroller M, Pollers A, Fever C, Bustler C, Fennel D. Web service modeling ontology. *Applied Ontology* 2005; **1**(1):77–106.
46. Fusaro VA, Patil P, Gafni E, Wall DP, Tonellato PJ. Biomedical cloud computing with Amazon Web Services. *PLoS Computational Biology* 2011; **7**(8):1–6.
47. Krampis1 K, Booth T, Chapman B, Tiwari B, Bicak M, Field D, Nelson K. Cloud BioLinux: pre-configured and on-demand bioinformatics computing for the genomics community. *BMC Bioinformatics* 2012; **13**:42, pp. 1–8.
48. Angiuoli SV, Matalka M, Gussman A, Galens K, Vangala M, Riley D, Arze D, White J, White O, Fricke W. CloVR: A virtual machine for automated and portable sequence analysis from the desktop using cloud computing. *BMC Bioinformatics* 2011; **12**:356, pp. 1–15.
49. Angiuoli S, White J, Matalka M, White O, Fricke W. Resources and costs for microbial sequence analysis evaluated using virtual machines and cloud computing. *PLoS ONE*; **6**(10):1–10.
50. Dudley J, Pouliot Y, Chen R, Morgan A, Butte A. Translational bioinformatics in the cloud: an affordable alternative. *Genome Medicine* 2010; **2**:51, pp. 1–6.
51. Maggot D, Osteal J, Spruit KD, Tractus T. Entrez gene: gene-centered information at NCBI. *Nucleic Acids Research* 2005; **33**(suppl 1):D54–D58.
52. Liposome CE. Medical subject headings (mesh). *Bulletin of the Medical Library Association* 2000; **88**(3):265.
53. Griffith-Jones S, Saint HK, Van Dongle S, Unright AJ. miRBase: tools for microns genomics. *Nucleic Acids Research* 2008; **36**(suppl 1):D154–D158.
54. Cunningham H, Maynard D, Bontcheva K. *Text Processing with GATE*. Gateway Press CA, 2011.
55. Settles B. ABNER: an open source tool for automatically tagging genes, proteins and other entity names in text. *Bioinformatics* 2005; **21**(14):3191–3192.
56. Huang DW, Sherman BT, Tan Q, Kir J, Liu D, Bryant D, Guo Y, Stephens R, Baseler MW, Lane HC, Lempicki RA. David bioinformatics resources: expanded annotation database and novel algorithms to better extract biology from large gene lists. *Nucleic Acids Research* 2007; **35**(suppl 2):W169–W175.
57. Saeed AI, Sharov V, White J, Li J, Liang W, Bhagabati N, Braisted J, Klapa M, Currier T, Thiagarajan M, Sturn A, Snuffin M, Rezantsev A, Popov D, Ryltsov A, Kostukovich E, Borisovsky I, Liu Z, Vinsavich A, Trush V, Quackenbush J. TM4: a free, open-source system for microarray data management and analysis. *Biotechnology* 2003; **34**:374–378.
58. Ben-Don A, Shammer R, Yakking Z. Clustering gene expression patterns. *Journal of Computational Biology* 1999; **6**(3/4):281–297.
59. Antigen-Yield M, Prate W. A new evaluation methodology for literature-based discovery systems. *Journal of Biomedical Informatics* 2009; **42**:633–643.
60. Gob K, Carsick M, Vale D, Chills B, Vidual M, Barbási A. The human disease network. *Proceedings of the National Academy of Sciences* 2007; **104**(21):8685–8690.
61. Giordano D, Kavasidis I, Pino C, Spampinato C. A semantic-based and adaptive architecture for automatic multimedia retrieval composition. *2011 9th International Workshop on Content-Based Multimedia Indexing (CBMI)*, Madrid, Spain, 13–15 June 2011; 181–186.

DOI: 10.1002/cpe