Universidade do Minho
Escola de Engenharia

Rafael Teodósio Pereira

Integrating Knowledge from Data and Literature for Building Transcriptional Regulatory Networks

março de 2016

Universidade do Minho
Escola de Engenharia

Rafael Teodósio Pereira

Integrating Knowledge from Data and Literature for Building Transcriptional Regulatory Networks

Tese de Doutoramento em Informática

Trabalho realizado sob a orientação do
Professor Doutor Rui Manuel Ribeiro Castro Mendes
e do
Professor Doutor Orlando Belo

março de 2016

DECLARAÇÃO DE INTEGRIDADE

Declaro ter atuado com integridade na elaboração da presente Tese.

Confirmo que em todo o trabalho conducente à sua elaboração não recorri à prática de plágio ou qualquer forma de falsificação de resultados.

Mais declaro que tomei conhecimento integral do Código de Conduta Ética da Universidade do Minho.

Universidade do Minho, 31 Março de 2016.

Nome Completo: Rafael Teodósio Pereira

Assinatura:________________

Acknowledgements

Firstly, I am grateful to the God for the good health and wellbeing that were necessary to complete this thesis. I would like to express my sincere gratitude to my advisor Prof. Rui Mendes for the continuous support of my Ph.D study, for his patience, motivation, and knowledge.

This work will not have been possible without the help of so many people, in so many ways. I would like to thank to Prof. Miguel Rocha, Hugo Costa and Sónia Carneiro for being untiring and be available whenever than I needed for discussing and direct my work. To Prof. Eugénio, director of the Centre of Biological Engineering by providing me financial support for conferences and congresses. I would also thank to my research group (BIOSYSTEMS), especially Sara who's shared with me for four years the same office, thanks for the good humor and provide a great work environment.

I am feel blessed in this moment for having a lot of good friends both from Portugal as well as from Brazil for giving me the support to achieve this important step on my academic career. Specially to João Marco, Alice Balbé, David Henriques, Hermes Pimentel, Jorge Guedes, Humberto Longo, Ricardo Ravanello, Tatiana Rehbein and Ricardo Martini for all funny moments that make this easier to achieve.

I must express my gratitude to my girlfriend Andriza Saraiva by continuous encouragement, support, patience and love during all these years.

Last but not the least, I would like to thank my family: my parents Renato Pereira and Neuza Pereira also my brother Lucas, for giving me supporting and encouraging throughout my life.

The work presented in this thesis was financially supported by National Counsel of Technological and Scientific Development - CNPq, Brasil. And also developed in partnership with SilicoLife company.

Abstract

Research in the Systems Biology field has been steadily increasing and one of the most addressed topics is the modeling and simulation of biological systems, whose aim is to recapitulate, *in silico* and *in vivo*, all processes that occur within the cell.

Several studies show that biological knowledge is steadily growing and is distributed across several databases, complicating the process of data integration since these databases often adopt different standards including structure, storage, identifiers and ways of exporting information. Furthermore, these databases often concern themselves with a specific organism or a given biological aspect.

Due to the large amount of biological data, the process of data integration has been one of the major challenges in the field of bioinformatics as well as discovering information about cellular process models, such as Transcriptional Regulatory Networks (TRNs). They are useful models for understanding the global organization of regulatory networks, its functional properties and their behavior.

This work developed a new approach for building regulatory networks, retrieving the required information from several databases and integrating it into a repository. A new pipeline was designed for extracting regulatory events from scientific papers stored on the PubMed database. Furthermore, these tasks were integrated into @Note, a software system that provides methods from the Biomedical Text Mining field, such as: Information Retrieval (IR) and Information Extraction (IE).

Resumo

A investigação no domínio da Biologia de Sistemas tem aumentado cada vez mais nos últimos anos e um dos temas mais abordados é a modelagem e a simulação de sistemas biológicos, cujo o objetivo é reproduzir, *in silico* e *in vivo*, todos os processos que ocorrem dentro de uma célula.

Vários estudos mostram que o conhecimento biológico está em constante crescimento e distribuído por diversas base de dados, o que dificulta o processo de integração de dados uma vez que estas bases de dados muitas vezes adotam padrões diferentes de estrutura, armazenamento, identificadores e também na maneira de exportar essas informações. Além disso, essas bases de dados, muitas vezes estão relacionadas com um organismo específico ou então um determinado aspecto biológico.

Devido à grande quantidade de dados biológicos, o processo de integração de dados tem sido um dos principais desafios no domínio da bioinformática, bem como a descoberta de informações sobre modelos de processos celulares, tal como as Redes Reguladoras da Transcrição. Elas são modelos úteis para a compreensão da organização global das redes reguladoras, suas propriedades funcionais e seu comportamento.

Neste trabalho foi desenvolvido uma nova abordagem para a construção de redes reguladoras, recuperando as informações necessárias a partir de diferentes bases de dados e integrando-as em um repositório. Uma nova sequência de tarefas foi desenvolvida para a extracção de eventos regulatórios a partir de artigos científicos armazenados na base de dados do PubMed. Além disso, estas tarefas foram integradas no @Note, um sistema de software que fornece métodos na área da Mineração de Textos Biomédicos, tais como: Recuperação da Informação (RI) e a Extracção de Informações (EI).

Contents

List of Figures

List of Tables

Listings

List of acronyms

BioTM Biomedical Text Mining

CPL Collection Programming Language

CRF Conditional Random Fields

DAS Distributed System Annotation

DNA Deoxyribonucleic acid

FBA Flux Balance Analysis

FRS Frame Representation System

GO Gene Ontology

GRN Gene Regulatory Networks

HMM Hidden Markov Models

IE Information Extraction

INSDC International Nucleotide Sequence Database

INDUS Intelligent Data Understanding System

IR Information Retrieval

ME Metabolic Engineering

MOMA Minimization Of Metabolic Adjustment

NCBI National Center Of Biological Information

NER Named Entity Recognition

NLP Natural Language Processing

OPM Object Protocol Model

OQL Object Query Language

PANTHER Protein ANnotation THrough Evolutionary Relationship

PPI Protein-Protein Interactions

POS Part-Of-Speech

RE Relation Extraction

REST REpresentational State Transfer

RNA Ribonucleic acid

ROOM Regulatory On/Off Minimization of metabolic fluxes

SBML Systems Biology Markup Language

SOAP Simple Object Access Protocol

SQL Structured Query Language

SRS Sequence Retrieval System

SVM Support Vector Machines

TF Transcription Factor

TRN Transcriptional Regulatory Network

VG Verbal Grouping

XGMML eXtensilbe Graph Markup Modelling Language

XML eXtensible Markup Language

Chapter 1

Introduction

Currently, one of the main issues addressed in the bioinformatics field is understanding the structure and behaviour of complex molecular interaction networks.

Several studies show that biological knowledge is steadily growing and is distributed across several databases. Integrating information from these databases is complicated since they often adopt different standards including structure, storage, identifiers and ways of exporting information. Furthermore, these databases often concern themselves with a specific organism or biological aspect.

Due to the large amount of biological data, the process of data integration has been one of the major challenges in the field of bioinformatics as well as discovering information about cellular process models that involve many different molecules, such as Transcriptional Regulatory Networks (TRNs).

1.1 Motivation

Transcriptional regulation is one of the most fundamental mechanisms for controlling the amount of protein produced by cells under different environmental conditions and developmental stages [1].

TRNs are useful models for understanding the global organization of regulatory networks, its functional properties and their behavior.

Actually, for building a TRN, researchers need to collect information across several different data sources such as biological databases, scientific literature, experimental data. It is important to integrate several sources in order to facilitate searches for genes and Transcription Factors (TFs) that are the main biological entities involved in the process of protein regulation.

Model reconstruction entails performing biological data integration. Currently, it is a large part of the work of scientists and involves querying multiple heterogeneous data sources, manually retrieving, integrating and also manipulating data using advanced data analysis and visualization tools.

Nevertheless, the vast amount of scientific papers makes this process quite difficult to accomplish. Furthermore, there is a large number of databases that store this kind of information and this number has been steadily increasing. These databases do not follow a standard for representing biological information and thus there are several inconsistencies regarding the name of genes, proteins and identifiers. The considerable amount of information implies that manual curation is a very time consuming, tedious and error prone task. Text Mining approaches can help automate this task.

1.2 Objectives

The main goal of this work is to develop a novel integrative approach to aid in the process of building TRNs by retrieving relevant information from biological databases and scientific literature.

Since most of the information available belongs to biomedical literature, a large part of this task entails selecting the relevant articles from a large body of papers. However, due to the rapidly increasing number of scientific papers, it is quite difficult to read all the papers that have been published about this subject.

In order to accomplish this, this work is focused on developing methods for retrieving information from biological databases, gathering as much information as possible; to create an integrated repository, that is able to store and load this data and also to design a pipeline to allow the reconstruction of TRNs through using Biomedical Text Mining techniques.

Concerning the data integration process, the information necessary for building TRNs will be harvested for two gram-positive bacteria (*Escherichia coli*, substrain *K-12 MG1665* and *Bacillus subtilis* substrain *168*). After that, Text Mining techniques will be applied to perform the reconstruction of regulatory networks.

As a proof of concept, these methods will be applied to *Escherichia coli substrain K-12 MG1665* because it is considered a well-known and widely studied organism and have a large amount information available among several databases.

The necessary information for building TRNs will include:

- Information about genes, proteins and transcription factors;
- Information collected from available databases containing relevant data;
- Information from the available regulatory models and their components;
- Information gathered from scientific literature.

In order to achieve this goal, several tasks need to be performed:

1. Development of methods for retrieving relevant biological information from the chosen organism;
2. Creation of an integrated repository;
3. Information retrieval from Biomedical literature;
4. Creation of a dictionary;
5. Information extraction from Biomedical literature;
6. Relation extraction;
7. Building the networks;
8. Extraction of Boolean rules from networks;

The first outcome is the development of the data repository structure, methods for the automatic retrieval of information from the databases and the tools to load the available information for the concerning organisms to repository.

Since most of the information necessary to build TRNs comes from Biomedical literature, the information gathered therein is of paramount importance.

This fact has heightened the importance of Biomedical Text Mining approaches in this task. In order to address this issue, a Biomedical Text Mining pipeline will be designed, using the @Note2 [1] framework, for recovering and extracting information from literature concerning regulatory events needed for the creation of TRNs.

The pipeline consists of five main tasks: Literature recovery; Corpus creation; Name entity recognition, Relation extraction and Network generation. The repository, developed in the previous task, is used to integrate information from several databases and create dictionaries which will used by the name entity recognition process for identifying genes, proteins and transcription factors.

The approach proposed in this thesis allows users to build TRNs from scientific papers (extracted from PubMed[2]) and information found on biological databases. In order to assess the quality of this approach, it is necessary to perform a validation process. The validation process chosen was to perform a comparison between the results obtained during the development of this work and the regulatory models already published as well as the findings of Orth [2].

1.3 Research Methodology

The methodology used in this work follows a series of steps that are closely related with the goals established in the previous section.

As a first task, it is important to take into account the relevant aspects related with TRNs as well as identify their main components, to know about

[1] http://www.anote-project.org/

[2] http://www.ncbi.nlm.nih.gov/pubmed

the relationship between biological entities present in these networks, identify and characterize these relationships within a document, how they can be represented in the computational field and their importance in the area of Bioinformatics.

After the literature review, a new biological data integration process is designed, due to the inconsistencies detected between biological databases. This task is considered fundamental for the following work, because it gathers information from different databases and stores in a repository, thus solving the inconsistency issues.

The method proposed in this work for reconstructing TRNs involves the application of several text mining techniques for identifying biological entities which might be involved in this process as well as their interactions. Due to this task, a cooperation was established with the SilicoLife company to develop a new approach for extending the @Note framework to allow this tool to reconstruct these regulatory networks.

In this step, several Biomedical Text Mining techniques were explored based in two methods: the Information Retrieval (IR) and Information Extraction (IE). This task begins by recovering a large amount of documents related to a specific domain, that is a component of IR, after that the IE step is able to convert a set of unstructured documents into structured data in order to apply methods like Named Entity Recognition (NER) and the Relation Extraction (RE) for the discovery of information.

It order to validate this approach, the models obtained will be compared with state of the art models found in the literature. This will be performed by a direct comparison of the models and by measuring their utility in flux balance analysis tools for maximizing the production of chemical compounds with a high interest in the industry.

1.4 Summary of Main Contributions

This work extended an existent framework adding a new useful function for constructing TRNs using a data integrated repository and Text Mining techniques. This section will present a summarized overview of main contributions of this work:

1. a review study that describes the main approaches used to deal with biological data integration and also analyzing an alternative for integrating this data by using biological ontologies [3];

2. a new approach for integrating data related to TRNs from different biological databases, storing it in the KREN repository [4];

3. extending @Note by adding a new functionality for reconstructinf TRNs using the data stored in the KREN repository [5].

Regarding the results of this thesis, a comparison has been performed to evaluate the production of Bio-components such as succinic acid in the *E. coli* organism using the improved model obtained in this work (*in preparation*).

1.5 Dissertation Layout

In the present Chapter, *Introduction*, described a brief summary about the growth of biological data, the difficulties of performing a data integration process and the importance of biological systems models, like the TRNs to help understand cellular behavior. Consequently, it describes the motivation for performing this work and the challenges therein. The main objectives are described as well as the research methodology and the main contributions.

Chapter 2, *State of the Art*, brings together the whole biological context needed to understand the main aspects of the Systems Biology field. Then, an encompassing overview of the main biological networks used to represent several biological processes is presented, focusing on TRN, describing their organization, components and properties. The chapter also summarizes the capacity of these networks to be represented by Boolean networks and this transformation is performed. Furthermore, several Biological databases are explored for searching the information concerning to TRNs and whether they provide ways of retrieving the data as well as a review of the existent approaches in the biological data integration field. Next, the main techniques used in Biomedical Text Mining area for identifying biological events are described. Moreover, the @Note software system is described and how the pipeline for relation extraction by using Text Mining techniques is implemented.

Chapter 3, *KREN: Integrated Repository for Biological Data*, presents KREN, a new approach for integrating biological data retrieved from several databases. The chapter reviews the information inconsistencies found among the databases and the need to use more than one for gathering all the information required, which motivates the existence of the repository. For this purpose, the selection of the databases was considered regarding their biological content related to TRNs and whether they provide a way of retrieving this information automatically. Finally the chapter presents a step-by-step development in order to build the repository that is able to store information from several organisms.

In Chapter 4, *Extending @Note Framework for Building Transcriptional Regulatory Networks*, is proposed an improvement to this software system for creating regulatory networks based on the Biomedical Text Mining techniques already implemented in this tool. First, it is necessary to use the KREN repository to provide the essential information required to extract regulatory events using @Note and also for retrieving the document identifiers associated to the PubMed database. After that, a large set of documents is retrieved from PubMed in order to apply the Relation Extraction (RE) process. This Chapter also describes an approach for visualizing the RE output and how this network might be represented using a Boolean system.

Chapter 5, *Case Study*, introduces several studies in order to prove the relevance of the present research work. The first test scenario is designed to find several specific protein complexes that belong to the whole regulatory network of the *E. coli* organism. Moreover, a comparison between the network created using this approach and the published one is performed in order to find biological entities that are shared or are present in one and not in other, ans also a biological process classification of these entities. As a last test case, an *in silico* simulation was performed in order to improve the existing model with the regulatory events found using the methods implemented in this work.

Chapter 6, *Conclusion*, presents an overview of this work, taking into account the objectives proposed and their achievement, the main contributions of this thesis, providing future work directions based on the broad scope of this approach.

Chapter 2

State of the Art

This chapter sketches the main biological concepts, starting with of the central dogma of biology, the importance of the Systems Biology field, the large amount of information which is produced by these studies and where this sort of information may be found. This chapter will further describe the features of biological networks and how they can be used for representing biological processes. Mainly focused on Transcriptional Regulatory Networks, including their features and how to represent them by using Boolean networks. Moreover, is also described in this chapter several biological databases related to this field as well as different approaches to integrated them.Finally, the concepts of Biomedical Text Mining are reviewed and how these concepts are implemented on @Note framework.

2.1 Central Dogma of Molecular Biology

The Central Dogma of molecular biology is a model used to define the transfer of information between macromolecules (DNA, RNA and protein). It describes the flow of genetic information from the deoxyribonucleic acid (DNA) to produce a given protein [6].

Figure 2.1 shows a representation of this concept where the flux begins with the acid deoxyribonucleic (DNA) being transformed into ribonucleic acid (RNA) which works like a messenger, carrying the information to the ribosomes (small organelles responsible for translating this information into the functional product)[7]. This suggests that the DNA encodes all the information that is necessary to produce proteins.

Since the genome was discovered, the molecular biology field began to produce a large amount of data, resulting in a search for ways of storing, organizing and interpreting this information.

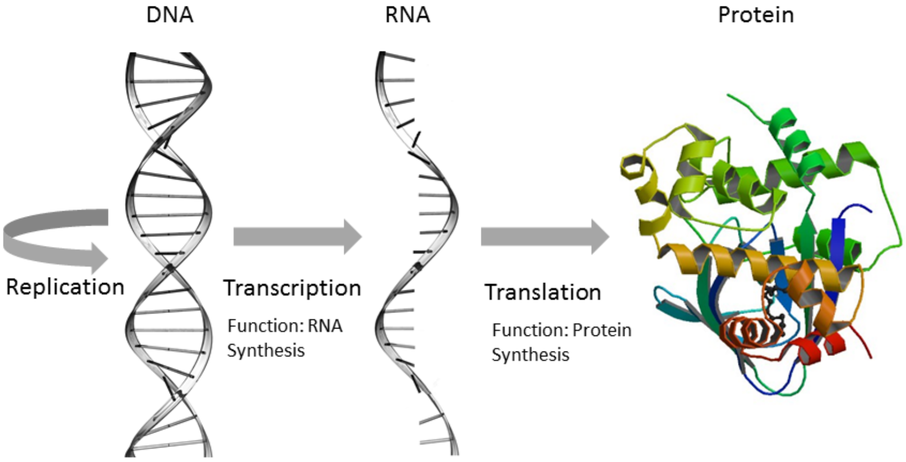


Figure 2.1: The central dogma of Biology

This phenomenon fostered the growth of the interdisciplinary area of Bioinformatics that aims to provide methods and algorithms for organizing, storing and using this new knowledge in order to understand the behaviour of organisms.

2.2 Systems Biology

Until the seventies, biological sciences were known by the study of living beings. However several new discoveries fostered a larger development of genetics and ultimately a new field called Systems Biology emerged.

Biology is one of the natural where it is possible to represent its information by means of digital code. By using skills from areas like Computer Science, Mathematics, Physics and Chemistry it is possible to develop tools that allow data to be stored, visualized, analyzed and shared by the scientific community [8].

Advances in genome sequencing techniques have led to the knowledge of the complete genome of several organisms including the Human Genome[1]. Together with the development of novel methods in Bioinformatics and Systems Biology, this has allowed the reconstruction of genome-scale metabolic and regulatory models. Thus, Systems Biology can be considered an interdisciplinary area, because it involves experimental approaches concerning studies on genomic, proteomic and metabolomic which can be used to analyze the process of gene translation, protein expression and predicting or verifying the flux of certain metabolites when they are under some biological, genetic or chemical conditions.

The Systems Biology field (cf. Figure 2.2) encompasses three main aspects: modeling, analyzes and experimentation. There are currently many laboratories gathering experimental data (e.g., genomics, proteomics, metabolomics) faster than they can be processed and storing them in databases. Furthermore, the analyzes of this data is producing a large body of distributed knowledge worldwide often disregarding the need for standards and protocols for its storage. Finally, the *in silico* simulation of all the processes inside the cell, both metabolic and regulatory, using mathematics models of biological processes is also producing data that has the same caveats.

One of the aims of Systems Biology is to discover ways of finding a cost effective sets of genetic modifications that can be performed in a given strain in order to produce a specific metabolite which might be used for industrial purposes by combining reliable models with simulations methods. In order to achieve this goal, computational tools have been developed to model and simulate several biological processes and store data produced by these systems.

Besides the large amount of heterogeneous information that is being steadily produced worldwide, there is also a vast number of publications about this information appearing faster than researchers are able to read it. The challenge here is to sift through the literature and select the important

[1] National Human Genome Research Institute - http://www.genome.gov

information among all the duplicates and irrelevant data for the studies being conducted by a given researcher.

Finally, the mathematical field closes the interdisciplinarity on Systems Biology. It is used to test and generate new hypotheses and assumptions that have been identified on the literature review process.

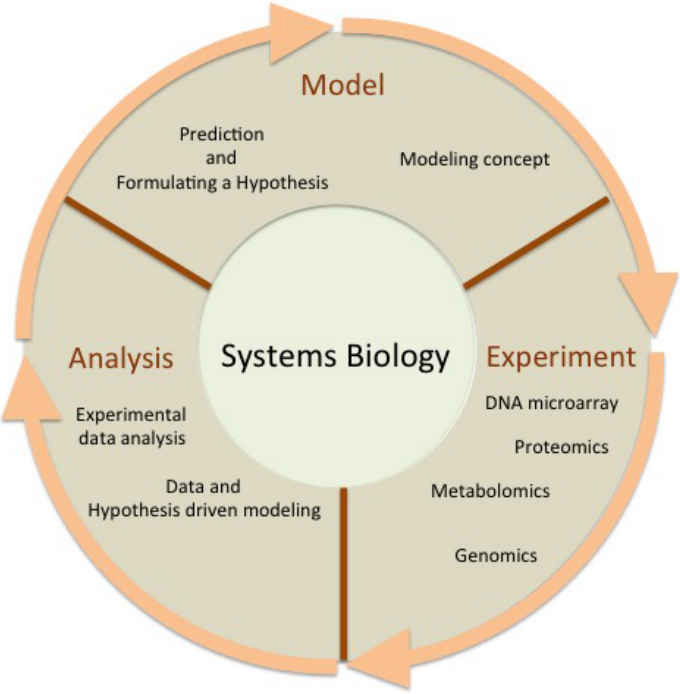

Model
Prediction and Formulating a Hypothesis
Modeling concept
Analysis
Systems Biology
Experiment
Experimental data analysis
Data and Hypothesis driven modeling
DNA microarray
Proteomics
Metabolomics
Genomics

Figure 2.2: Process diagram for representing the application of Systems Biology

2.3 Biological Networks

Biological Networks provide a useful model for analysing and understanding biological systems. They are used to represent several interactions between biological entities (e.g., genes, proteins, chemical compounds) [9]. These biological entities are quite complex and their behavior stems from the strong interconnectivity between them. The use of Biological Networks provides a way to think about these complex systems in terms of graphs and use some of these graph statistics such as the number of neighbours of a certain node, their connectiveness or centrality to understand how important a given node is to the network.

2.3.1 Types of Biological Networks

Nowadays, Bio-Networks have been widely used for several biological processes [10] such as: metabolism, gene regulation, signal transduction and also for the discovery of drug targets [11].

There are several approaches of Bio-Networks, such as Protein-Protein Interactions (PPIs), Metabolic Networks (ME) and Gene Regulatory Networks (GRN). Each of these networks depicts different biological entities (cf. Fig.2.3).

PPI networks (cf. Fig. 2.3A) represent protein-protein interactions. These networks help understand how strongly proteins interact by observing the weight of the edges. They also depict the type of regulation activity of proteins, i.e., whether if a given connection represents an activation or repression. They are essential to predict proteins' functions and cellular processes.

GRN networks, generally are a little more complex than PPI networks, because their structure comprises not only proteins, but also transcription factors and genes. As seen in Figure 2.3B, nodes represent biological entities: the proteins are represented by squares (blue), genes are represented by circles (purple) and the transcription factors by octagons (red). The edges in this type of network represent an action and/or a regulatory interaction. These networks are used to understand the flow of information in a biological system [12].

ME networks (cf. Fig. 2.3C) are more complex than the others described previously because they are composed by a set of genes (represented by purple circles) which codes a set of enzymes (represented by blue squares). The triangles represent the biochemical reactions which are connect to compounds, represented by pentagons (beige), identifying the metabolic reactions. This type of network is generally used to understand the metabolism of an organism, as well as the processes that generate essential compounds such as

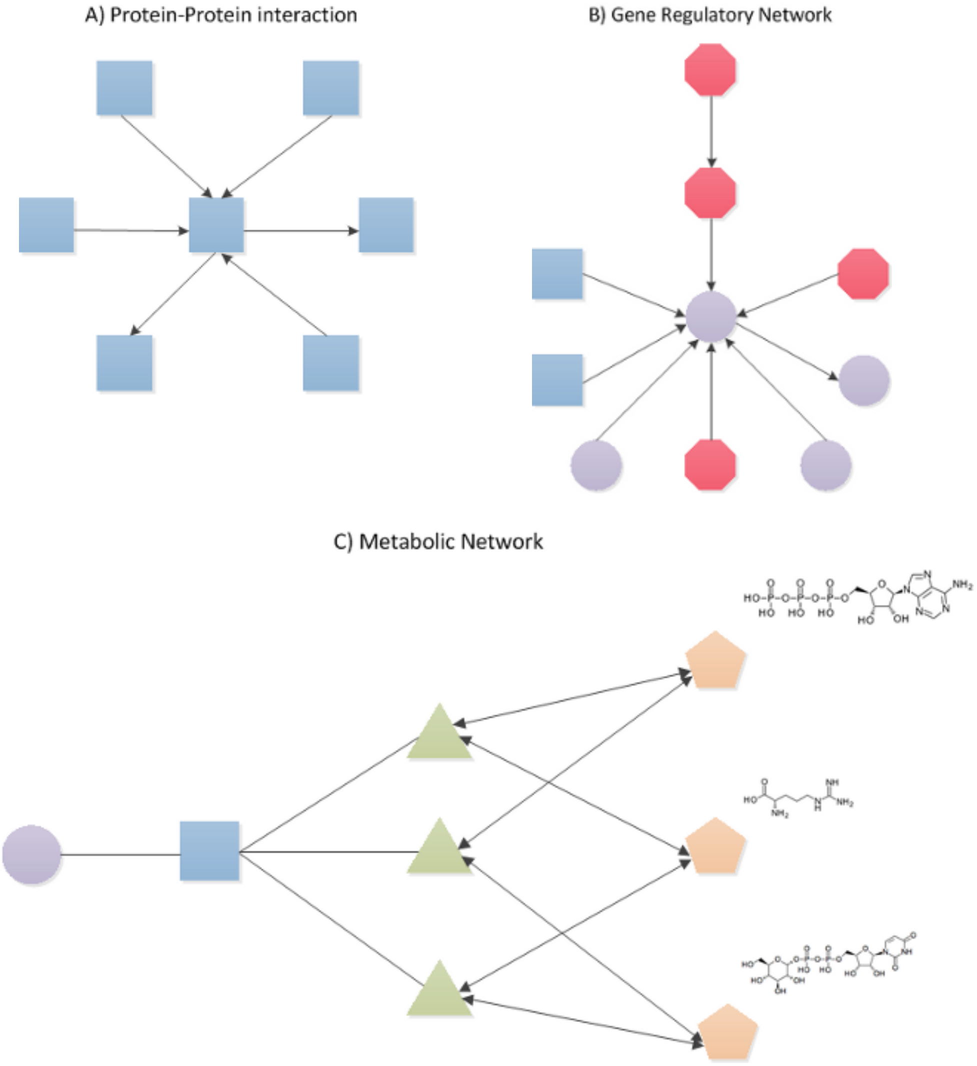


Figure 2.3: A) A representation of interactions which occurs in a PPI networks ; B) Graph that represents the components of a Gene Regulatory Network; C) Illustration of Metabolic Network

lipids, sugar and amino-acids.

2.4 Transcriptional Regulatory Networks

Transcriptional Regulatory Networks (TRN) model the biological process that links regulatory proteins with the genes in a genome. In the 1960's, genetic and biochemical experiments demonstrated the presence of regulatory sequences in the proximity of genes and the existence of proteins that are able to bind those elements and to control the activity of genes by either activation or repression of transcription [13].

Research in the Systems Biology field is steadily increasing in the last years and one of the most addressed topics in this area is the simulation of biological systems, whose aim is to perform a reconstruction, *in silico* and *in vivo*, of all processes that occur within the cell, both metabolic or regulatory. The reconstruction of these networks helps simulations to be paired with experiments, and they are often used by computational scientists in order to understand the quantitative behaviour of many complex biological systems [14].

In recent years, the exploration of life sciences, has been bolstered through the advent of whole genome sequencing. This new information potentiates the reconstruction of genome-scale metabolic networks [15]. But only the reconstruction of metabolic networks is not sufficient to understand the principles about how organisms function. After this step it is necessary to discover how a genetic machinery operates inside an organism, which includes the Transcriptional Regulatory Networks.

2.4.1 Properties of Transcriptional Regulatory Networks

According to *de-Leon* [16], the regulation is composed by two complementary components; The first component concerns the regulatory genes, e.g. transcription factors and signalling molecules.

Transcription factors bind to specific sequences in the DNA and activates or inhibits the transcription of a gene. Signalling molecules carry out the communication between cells and initiate the activation of certain transcription factors in the cells that receive the signal.

The complementary part of these components is the regulatory genome.

Every gene contains regulatory sequences that control when and where it is expressed. The regulatory sequences are arranged in units that are termed *cis*-regulatory modules, that contain a cluster of transcription factor binding sites as shown in Figure 2.4*a*. *Cis*-regulatory modules behave as an information processor that reads the regulatory state of the cell and activates or inhibits the gene that it controls.

Figure 2.4*b* portrays some *cis*-regulatory modules (pink squares), each responsible for controlling gene activity at a given time and in a given organism. The green boxes represent the exons. Activation or inhibition are rules that form a regulatory network that can be seen in Figure 2.4*c*, where gene B is the binding site for gene A (*blue*) and C (*red*).

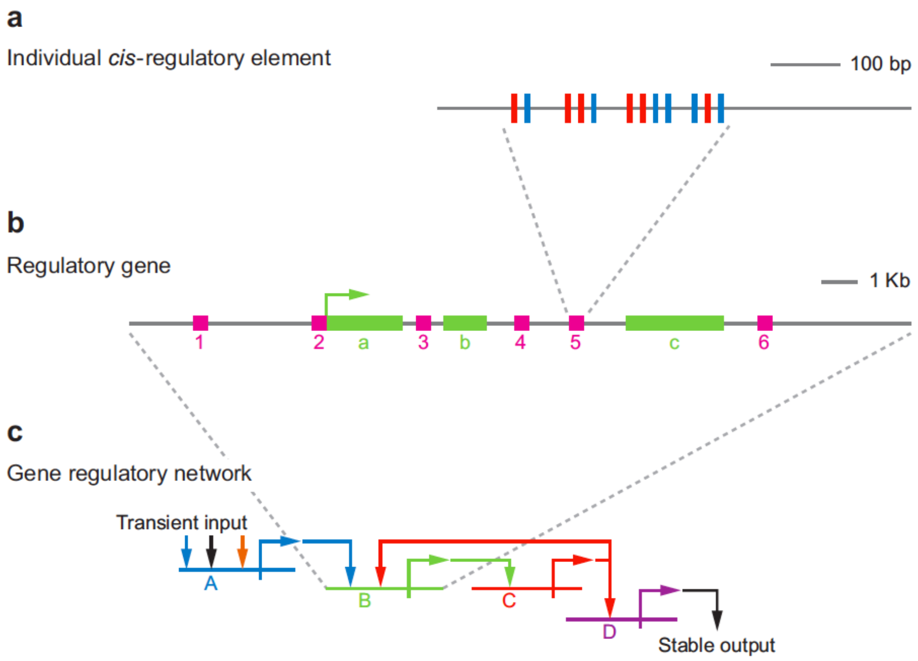


Figure 2.4: The gene regulatory hierarchy [16]. (a) Individuals *cis*-regulatory element; (b) the pink boxes represents the *cis*-regulatory modules; that control it is expressions. (c) Inter-regulating transcription factors and signalling from a network.

The most common form of representing a TRN by using directed graphs, where nodes represent regulators (i.e., transcription factors) and targets, and the edges represent the regulatory interactions. This representation may be divided into three levels, where the first one includes the set of transcription factors, downstream target genes and DNA binding sites (Figure 2.5a). At the second level, these basic units are arranged into common patterns of interconnections called network motifs (Figure 2.5b). At the third level, motifs are grouped into semi-independent transcriptional units named modules (Figure 2.5c)[17]. Thus at the last level, the regulatory network is composed of interconnecting interactions between the modules that comprise the entire network (Figure 2.5d)[18].

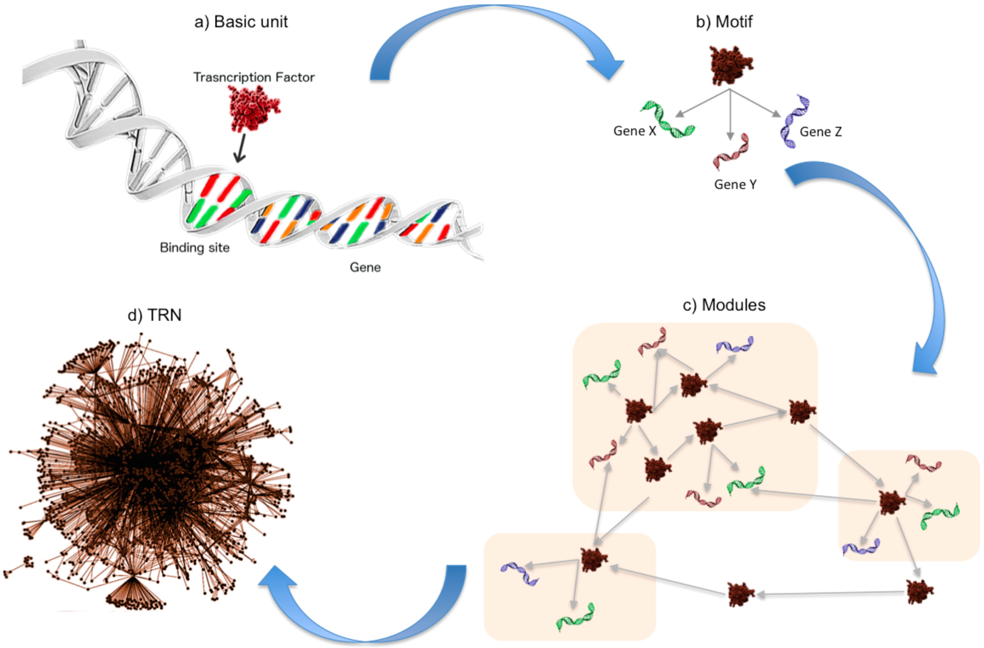


Figure 2.5: Organization of a transcriptional regulatory network: (a) Comprises the transcription factor, its target gene with DNA recognition site and the regulatory interaction between them. (b) The basic units are organized into networks motifs. (c) Network motifs are interconnected to form semi-independent modules. (d) The whole set of interactions that represent a transcription regulatory network

The following describes in better detail the main concepts related with Transcription Factors (TFs), Motifs, Modules, and finally a point of view about the global organization of TRN.

Transcription factors are proteins with special abilities and attributes not found in other classes of proteins. Normally they work in pairs or networks to modulate specific regulatory pathways [19], forming multiple interactions that allow for varying levels of control over rates of transcription. TFs have an important function within transcription regulation, they are responsible for recognizing their targets through specific sequences, called motifs [20].

Motifs represent the simple units of network architecture, in which there are specific patterns of inter-regulation between TFs and target genes [17]. Thus, a TRN may be defined as a network of motifs, where each of these motifs has a specific function in determining gene expression [21].

Modules define an intermediate level. Intuitively, one might expect distinct cellular processes to be conveniently regulated by discrete and separable modules [17]. These modules represent a set of motifs that can be interconnected in semi-independent ways. Thus, regulatory networks may be tightly interconnected and several modules can be separated from the rest of the network.

Global organization of TRN can be described by parameters derived from graph theory [17], as previously mentioned. Other ways of representing a TRN is through a G x R matrix, where G is the number of genes considered and R is the number of transcription factors involved in regulation [20]. When an input in position $(i{,}j)$ receives the value 1, it indicates a regulatory relationship between, the i-th gene and j-th TF. And when an entry receives a value of 0 it indicates there is no regulation between them. Thus, we may conclude that the most basic elements in these networks are the transcription factors and target genes, and the regulatory interactions between them.

2.5 Boolean Networks

With the growth of data in the field of Molecular Biology, there has been an increasing number of approaches that use Boolean networks for studying cell behaviour. In 1969, Kauffman [22] suggested that living organisms are composed by a complex of net interactions between millions of chemical components. One of the contributions of this work was the use of a model where the state of each gene at any given time is one of the Boolean values (*True/-False* or *On/Off*) and where the relationships among genes in a regulatory network can be expressed by Boolean equations. In a Boolean network, genes (both regulatory and regulated) are represented as vertexes and the actions between them, like activation or inhibition are represented by edges [23].

Nowadays, several approaches to model gene regulatory networks can be found, such as linear models [24], Bayesian networks [25], neural networks [26]. But, recent studies reinforce the idea that several biological process may be modeled through the Boolean formalism. TRNs can be represented by Boolean networks because their components exhibit a switch-like behaviour, changing from one state to another in the process of growth, or when the cells need to respond to external signals [27].

In order to illustrate this point, let us imagine an empirical regulation or a part of a regulatory network (cf. Figure 2.6), where the final product can be the production of a certain protein **K**. In this network, the genes are represented by circles, proteins are represented by squares and the hexagon represents the transcription factor. Two genes (**X** and **Y**) activate the protein **Z**, combining with a negative regulation of the transcription factor **W** they activate the gene **T**. Finally the protein **K** is activated inhibiting the gene **R**.

The network shown previously can be represented by a logic circuit diagram (cf. Figure 2.7), that represents a Boolean function of four input variables: **X**, **Y**, **W** and **R**, which determines whether the activity of **K** will be turned *on* or *off*. It is possible to observe that **Z** is completely dependant on the values of **X** and **Y** using the **AND** operation.

In the Boolean Network (Figure 2.6), **W** is shown as an inhibition effect over the **T** gene which in turn regulates **K**, on the other hand, in the logic circuit (Figure 2.7) the same information is illustrated using logic circuits, where the value of **W** and **Z** determine the value of **T**.

These concepts are very important to start the reconstruction of regula-

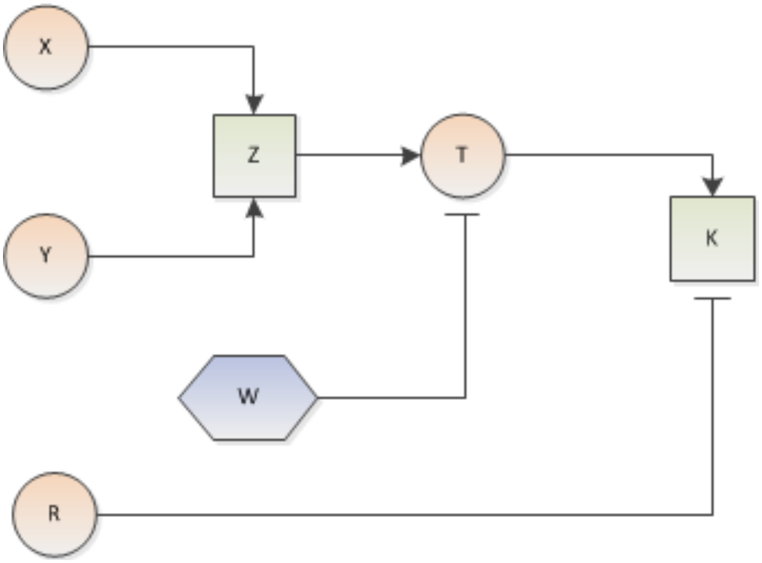


Figure 2.6: Illustration of a diagram that represents the regulation example. Arrowed lines represent activation and lines with bars at the end represent inhibition

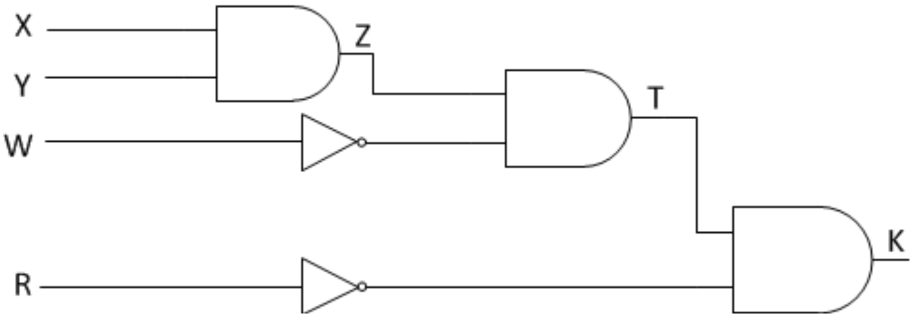


Figure 2.7: Illustration of a Boolean circuit that represents the regulation example. In this case, Boolean gates are used to combine the values of the entries **X**, **Y**, **W** and **R** in order to determine the value of **K**. In this case **NOT** gates represent inhibition while **AND** gates represent co-regulation. The genes **X** and **Y** need both be *on* in order for **Z** to be *on*. **T** will be *on* if both **Z** is *on* and **W** is *off*. Finally, **K** in *on* if **T** is *on* and **R** is *off*.

tory networks that will allow researchers to understand how bacterium such as *Escherichia coli* and *Helicobacter pylori*, can adapt to almost all environmental conditions and how they control the response to environmental changes.

2.6 Biological Databases

Biological databases were initially developed to provide information about genomic sequencing. With the advance of biological techniques and the increase of experimental studies, several databases were created to provide this information to the scientific community. Nowadays, one may find a large number of databases which provide information about several themes like organisms, genetic sequencing, proteins or scientific publications.

However, only a few of these sources have been curated by professionals, while in others the results were automatically generated thus resulting in possible errors and data inconsistencies. Another important issue concerning this proliferation of databases is the lack of standards which results in a heterogeneity of both data types, indexes and structure.

2.6.1 Examples of Biological Databases

Currently, there is a considerable number of biological databases that allow users to search for information concerning Molecular Biology. This work will describe some of these databases, namely NCBI, EcoCyc, PyloriGene, KEGG, Brenda, STRING, GeneRIFs, UniProt, RegulonDB and Swiss-Prot. The information presented below is also summarised in Table 2.1.

- **NCBI**: this center was created as a division of the U. S. National Library Medicine (NLM) to provide computational methods to help biomedical researchers. Its main goal was to develop new technologies in order to help solve biomedical problems. More specifically NCBI was created to provide systems that allow storing and analysing knowledge in molecular biology, biochemical and genetics fields, facilitating high level information to the scientific community by means of databases and software. NCBI can be considered one of the largest research centers in Bioinformatics. It integrates literature databases, genomic databases, tools for data mining, tools for sequence analyzes and other resources in several Biology fields [28].

- **EcoCyc**: This database was created to provide biological information about a specific organism (*Escherichia coli*). The information in this database is mainly about enzymes and metabolic pathways. The data organization of EcoCyc uses a frame knowledge representation system (FRS), that provides an objected-oriented data model, and has several advantages over a database approach because it organizes information within classes each representing a set of objects that share similar properties and attributes [29]. The main purpose of EcoCyc is to store information about proteins, pathways and molecular interaction in *E. coli*. But in recent years its focus was expanded to include annotation and literature-based curation of gene and protein functions of enzymatic, transport and binding reactions, as well as transcriptional regulation, covering the entire genome[30].

- **PyloriGene**: this is a database dedicated to the *Helicobacter pylori* bacterium. It stores and integrates genomic information about this organism. PyloriGene provides a complete dataset of DNA and protein sequences linked to the relevant annotations and functional assignments, enabling users to easily browse and retrieve information [31]. The main goal of PyloriGene, is to store genomes and new literature information in order to improve the functional annotation and classification of published coding sequences. PyloriGene was developed using a generic database model that was initially dedicated to other genomes. The data structure was designed for the representation, manipulation and maintenance of complete microbial genomes and the conceptual model was implemented as a relational database [32].

- **KEGG**: is a bioinformatics resource for understanding the functions and utilities of cells and organisms from both high-level and genomic perspectives [33]. It provides an integrated resource containing genomic, chemical, and network information while still allowing links to foreign databases. KEGG consists of fifteen main databases, that describe several characteristics like pathway maps, human diseases, organisms and biochemical reactions. The KEGG database can be divided into three levels: the first one, called PATHWAY, represents the higher order functions in terms of a network of interacting molecules; the second one, called GENES, is responsible for cataloguing the genes of all the known sequenced genomes and others that are only partially sequenced; and the third one, called LIGAND, stores chemical compounds, enzyme molecules and enzymatic reactions [34]. KEGG can be considered a computer representation of a biological system, consisting of molecular building blocks of genes and proteins (genomic information) and chemical substances (chemical information) that are integrated within a central repository.

- **BRENDA**: is a relational database that provides functional and molecular information about enzymes, based mainly on literature. The enzymes that are stored in BRENDA, are classified according to their EC (Enzyme Commission) numbers that are unique for each enzyme.

  Nowadays, more than 5800 different enzymes are covered by BRENDA [35]. It is an important tool for biochemical and medical fields because it provides information on several properties about all the classified enzymes, including some features like data occurrence, catalyzed reaction, stoichiometry, substrates/products, inhibitors, cofactors and activators [36]. BRENDA has an important feature which is its ability to integrate other sources in order to create a complete picture about enzyme properties.

- **STRING**: the main goal of STRING is to provide a database and web resources for protein-protein interactions, including both physical and functional interactions. It stores and integrates information from several sources, including experimental repositories, computational prediction methods and public text collections [37]. Protein-protein interactions have been proven to be a useful tool for clues on how to organize all protein-coding genes in a genome.

  The complete set of protein associations can be seen as a large network, which captures the current knowledge on the functions and interconnections of the cell [38].

STRING is the main site to search hundreds of organisms (ranging from Bacteria and Archaea to humans). This large number of organisms, represented by fully sequenced genomes, also enables STRING to periodically execute prediction algorithms about the interaction that depends on genome sequence information [39].

This database can be characterized by three aspects: it provides a uniquely comprehensive coverage, with more than 1000 organisms, 5 million proteins and more than 200 million interactions.

- **UniProt**: provides a stable, comprehensive and freely accessible central resource on protein sequences and functional annotation. The main activities performed by UniProt are the manual curation of protein sequences assisted by computational analyzes, sequence storage, development of a user-friendly web site and cross-referencing information with other databases. It is the central resource for storing and interconnecting information from large and heterogeneous sources and the most comprehensive catalog about protein sequence and functional annotation [40]. The structure of UniProt is formed by four components: UniProtKB (UniProt Knowledgebase) is a curated database, that provides functional information about proteins with cross-references to other sources, the second one, UniProt Archive (UniParc), is a repository that contains only protein sequences, the third one, UniProt Reference Clusters (UniRef), whose databases are generated based on UniProtKB and UniParc to provide an updated set of sequences. Finally, the last one is UniProt Metagenomic and Environmental Sequence Database (UniMES) was developed to address the expanding area of metagenomic data [41].

- **RegulonDB**: is currently one of the largest databases offering curated knowledge about transcriptional regulatory networks for several organisms [42]. It was initially developed to be the main reference database offering curated information about the transcriptional regulatory network of *Escherichia coli.*

Nowadays, RegulonDB is a relational database that provides, in a structured manner, manually curated knowledge about transcriptional regulation to the scientific community interested in bacteria. This includes curated information about transcription initiation through the activation or repression of transcription factors (TFs), which bind to individual sites around promoters [43].

- **Swiss-Prot**: is a curated protein database, which strives to provide a high level of annotation (such as the description of the function of a protein, its domain structure, post-translational modifications, etc.), a minimal level of redundancy and a high level of integration with other databases [44]. It consists in two classes of data: both sequence data and its annotation. For each sequence entry the core consists of the sequence data, the citation information (bibliographical references) and the taxonomic data while the annotation class provides a description of protein functions, post-translational modifications, domains and sites, secondary structure, quaternary structure, diseases associated with deficiencies in the protein and sequence conflicts [45]. Since 2002, Swiss-Prot is maintained by the UniProt consortium and is accessible via the UniProt web site [46].

2.7 Biological Data Integration

In order to perform studies in the field of Systems Biology, it is important to gather a large amount of reliable information. However, the previous section indicates that there is a considerable number of databases available on the Internet. The large quantities of data spread over many databases, each adhering to their own standards, nomenclature and indexes and the ever growing body of academic articles render the task of collecting information for conducting a given study quite difficult. It often forces researchers to manually query each database and understand how to conciliate the information, find synonyms and remove redundant information. These are the

Table 2.1: Differences between biological databases

Database	Database model	Knowledge domain	Web services	External sources
NCBI	Relational database	Literature, genomic and sequence analyzes.	Yes	Yes
EcoCyc	Object-oriented data model	Genes, proteins, pathways and molecular interactions.	Yes	Yes
PyloriGene	Relational database	DNA and protein sequences.	No	No
KEGG	Relational database	Pathways maps, human diseases, organisms, biochemical reactions, etc.	Yes	Yes
BRENDA	Relational database	Functional and molecular information about enzymes.	Yes	Yes
STRING	Relational database	Physical and functional protein-protein interactions.	No	Yes
UniProt	Relational database	Protein knowledgebase.	Yes	No
RegulonDB	Relational database	knowledge on transcriptional regulation.	Yes	No
Swiss-Prot	Relational database	Proteins.	No	Yes

reasons why Data Integration is both a critical and challenging task in the studies of biological systems.

One way to solve the data integration issue is to create a database that incorporates several types of biological information. The best example of this is

NCBI (National Center for Biotechnology Information) [28]. In this database it is possible to find a large number of tools and sources to explore the bioinformatics field, as well as search engines to find papers on PubMed (literature database), DNA sequences on GenBank (genetic sequence database), tools to analyze sequence alignment, data mining and many others.

2.7.1 Different approaches on Data Integration

Approaches besides NCBI aim to solve the integration data problem such as INDUS (Intelligent Data Understanding System) a system for information integration and knowledge acquisition from semantically heterogeneous distributed data [47] and DAS (Distributed System Annotation) a widely adopted protocol for the integration of biological data types in user-driven contexts [48].

Some systems have been developed specifically for the integration of biological data sources, such as SRS (Sequence Retrieval System) [49], K2 [50], Kleisli [51], IBM's Discovery Link [52], TAMBIS [53], OPM [54], BioMediator [55], among others.

SRS was originally designed to facilitate the access to biological sequence databases, but since a few years ago, researchers developed new capabilities in order to extract data from text files. The key feature of SRS is its unique object oriented design. It uses meta-data to define classes for database entry objects and rules for text-parsing methods, coupled with the entry attributes [56].

K2 and Kleisli are very similar approaches since they use a complex data model. The Kleisli model uses a specific language for querying and transforming complex value data, called Collection Programming Language (CPL) [57] that has the same power of SQL (Structured Query Language). Kleisli is a query system based on the nested relational data model, which is a purely "value-based" model and, as such, does not support any concept of schemas or integrity constraints [54].

The Kleisli system can also support many different types of external data sources by adding new wrappers, which forwards Kleisli's requests to these sources and translates their replies into Kleisli's exchange format [58]. Consequently, K2 was developed because CPL is too different from frequently used querying languages like SQL and it implements the Object Query Language (OQL) that is quite similar to SQL and also incorporates a new data type: *dictionaries*. A dictionary is a function with an explicit finite definition domain [51].

Object Protocol Model (OPM) provides constructs for modelling scientific database applications in terms of objects and protocols. OPM allows users to have different views of the same database application and provides them with high-level querying capabilities [59].

IBM's Discovery Link was developed to integrate and analyze large quantities of diverse data using database middleware technology to provide integrated access to data sources used in the life sciences industry. DiscoveryLink is unique among existing systems because it enables easy creation of wrappers for non-relational sources and provides the capability to add new sources dynamically. It also includes query optimization technology that automatically searches for the most efficient means of executing a query and assembling its results [60].

TAMBIS (Transparent Access to Multiple Bioinformatics Information Sources) is a complex architecture system that provides data integration through five biological sources thus enabling biologists to ask complex questions over a range of bioinformatics resources. It is based on a model of the knowledge of the concepts and their relationships in molecular biology and bioinformatics [53].

Another system that provides biological data integration is BioMediator, that was cited above. This system was designed as a tool for posing queries across semantically and syntactically heterogeneous data particularly in the biological area [55]. It was developed to address the problems presented by traditional databases through three main concepts. First, an annotated schema is used to provide a generic description of the biological data; secondly, it provides a simple interface for choosing or creating new relationships through existing knowledge; and finally, the data source which is present in the BioMediator system is generalized in such a way that it exposes all relevant data without using a specific schema.

Most of the systems mentioned above assume a predefined global schema (e.g., Discovery Link, OPM) or ontology (e.g., TAMBIS), with the notable exception of BioMediator, where users can easily tailor the integrating ontology to their own needs. This is highly desirable in a scientific discovery setting where users need the flexibility to specify their own ontologies [47]. Table 2.2 provides an overview of these systems, showing their different aspects such as data model, query language and whether they present bioinformatics tools.

Table 2.2: Data integration systems in bioinformatics

Integration system	Data Model	Query Language	Bioinformatics tools
INDUS	Ontologies	OWL	No
SRS	Icarus	SRS-QL	Yes
K2	No	OQL	Yes
KLEISLI	No	CPL	Yes
IBM's DiscoveryLink	IBM DB2	SQL	Yes
TAMBIS	Ontologies	GRAIL	No
OPM	Semantic model	OPM	No
BioMediator	Semantic model	PQL	Yes
DAS	DAS-protocol	XML	No

2.8 Biomedical Text Mining

Recently, Text Mining has been a helpful technique for extracting information from unstructured documents and also for discovering new meaningful knowledge from it. Scientific literature holds a vast amount of knowledge to be explored. The number of works published is growing quite rapidly and presents a challenge for researchers due to the sheer amount of papers being published on any given subject. This increase in the number of papers is both a boon and a curse because this increase entails a necessity for reading all of these papers to sift through the information in order to find the relevant one among all the replicated and unnecessary information. In order to illustrate this point, Figure 2.8 shows the growth in the number of publications for the query "*Escherichia coli*" performed on the PubMed database.

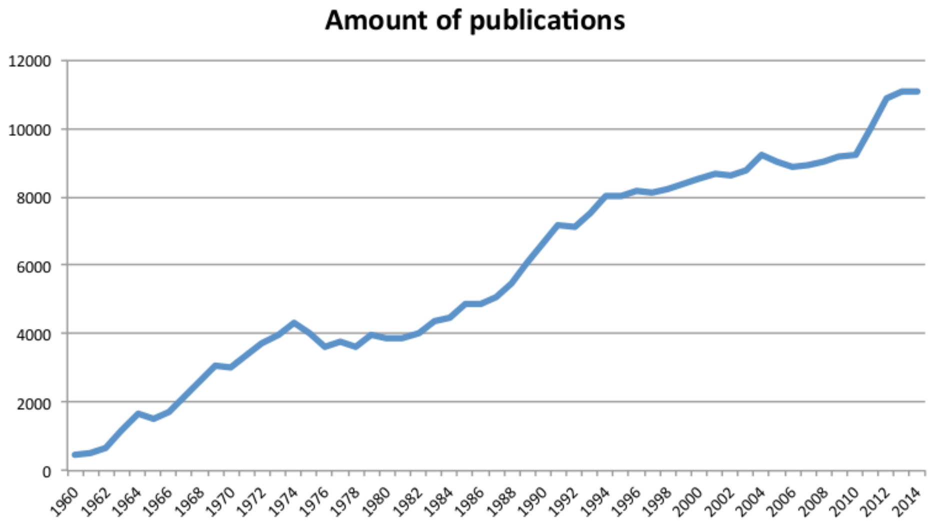


Figure 2.8: Query performed on PubMed database related with subject "*Escherichia coli*"

Text Mining techniques have been used for identifying useful biological information and extracting it automatically from texts. It is used in the Systems Biology field in order to uncover information for simulating, describing or modeling biological systems [61].

Recently, Text Mining is considered an important area of studies in Systems Biology, its importance fostered the creation of a new sub-field called Biomedical Text Mining.

The Biomedical Text Mining field can be divided in two main areas: Information Retrieval and Information Extraction. Whereas IR allows the extraction of all the parts from a document (e.g. abstract) that is stored in a given repository, IE can be divided into Named Entity Recognition (NER), where biological entities can be recognized in the text and Relation Extraction that finds relations between these entities. The next subsections will describe both IR as IE.

2.8.1 Information Retrieval

The aim of this process is to recover as many documents as possible from a given bibliographic source, according to a query performed by the user. The most common methods used for Information Retrieval (IR) are the search for specific terms on specific parts of documents or the search based on keywords. In the first approach, the IR systems will search the specific set of words which are identified in the title, abstract or in the entire document. On the other hand, the search based on a set of keywords, combines predefined words and applies Boolean operators like AND, OR and NOT to retrieve a set of documents related to the query [62].

A recognized and well-used IR system in biomedical field is the one used by PubMed. It uses two methods for the IR process, PubMed applies Boolean operators for combining words and retrieves a set of documents that matches the keyword given. It also implements a vectorial model, transforming each document into a vector of terms and, for each term identified in the text, a value proportional to the frequency that it appears, thus allowing the establishment of a comparison between the terms that compose the query and the terms found in the article [63].

In conclusion, IR systems allow users to perform directed searches and help them find and retrieve relevant documents for their studies.

2.8.2 Information Extraction

Information Extraction (IE) is an important component of a Text Mining approach. Usually, IE aims to find structured information inside natural language documents by either performing syntactic parsing or classification. The simplest and most common approach is to use methods that perform classification of the related biological literature [64]. In others words, IE is used to convert a set of unstructured documents, called *corpus*, into structured data thus enabling the application of methods for extracting new knowledge, such as Named Entity Recognition and Relation Extraction that will be described next.

2.8.2.1 Named Entity Recognition

In an overview of biomedical literature it is possible to find a large number of scientific terms which can be represent proteins, genes, chemical compounds or other biological entities. A Named Entity Recognition (NER) process is responsible for identifying biological entities or specific terms which might be present within the text.

Several issues may arise in this step since it is common to find the the same biological entity written in different ways (e.g., synonyms, abbreviations or ambiguous information) in scientific literature. For these reasons, this is an extremely challenging task in the Text Mining field, with many recent contributions that aim to develop new approaches to circumvent the problems described above.

Nowadays, it is possible to find several approaches for performing this task. Generally they implement one of the following types of methods: tagging based on a lexicon vocabulary; rule-based approaches and Machine Learning techniques.

The first method uses a dictionary curated by specialists in the subject of research in order to find a match between the terms stored in the dictionary and the terms in the document. Every term identified by the dictionary is annotated into a predefined biological class, such as gene, protein, chemical compound and so on. However this method has several limitations that can compromise the process, such as due to incomplete dictionaries or the ambiguity of terms [65].

The rule-based method recognizes entities by search for words or regular expressions in the text. However, this method has some drawbacks since the rules need to be adapted for other biological events since it is not possible to use the same rules among different events.

Methods that use Machine Learning approaches require a set of document models for training in order to classify and identify the terms in the remaining documents. The documents used in the training task must be manually created by an expert in the field. The Machine Learning approaches most commonly used in NER process are Hidden Markov Models (HMM) [66], Support Vector Machines (SVM) [67] and Conditional Random Fields (CRF) [68].

Figure 2.9 shows some approaches that implement methods based on rules, Machine Learning and lexicon resources and it is possible to conclude that the major approaches (ABNER [69], BANNER [70], CheNER [71], Moara [72], OpenDMAP [73]) were developed to use Machine Learning as a technique for recognizing entities. On the other hand, Oscar4 [74] and Linneaus [75] were based only on lexical methods. Among the approaches referred in this section, the GATE [76] architecture, implements both lexicon resource and rule-based ones. There are three other approaches that allow to use both lexicon and Machine Learning methods for NER processing: Neji [77], GeneTUKit [78] and GNAT [79]. Only the @Note2 [80] framework allows the use of Machine Learning, rules and lexicon approaches in order to perform the recognition of biological entities.

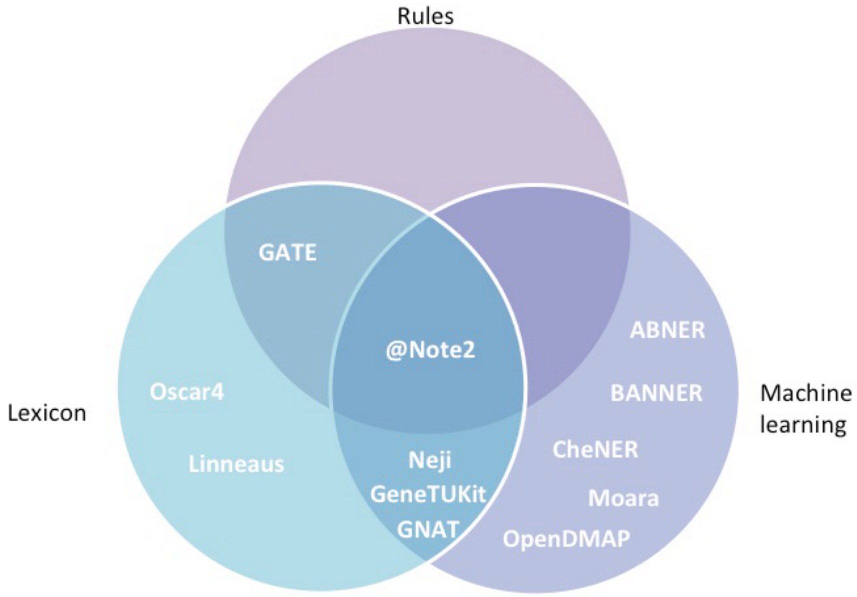


Figure 2.9: In this diagram is possible to see the relationship among of use of different methods developed to perform NER processing.

2.8.2.2 Relation Extraction

The aim of Relation Extraction (RE) is to identify and characterize possible semantic relations between specific terms or entities in the text. Identifying these relationships is a complex task since these texts are written in natural language and are thus quite difficult to understand by machines.

The most common techniques used for performing this task are borrowed from the Natural Language Processing (NLP) field, which emerged around 1950s as the intersection of Artificial Intelligence and Linguistics [81]. NLP aims to extract semantic meaning from text. It is can be represented within text by formal grammars that specify relationships between text units [82].

NLP can be divided into two main layers: lexicon and syntactic. The lexicon layer looks at the words of a given language, and is responsible for collecting the information about the words according to their grammatical class, while the syntactic level groups these words into sets of sentences appearing in the text.

2.8.2.2.1 Lexicon layer is divided into several sub-tasks, such as breaking the text into tokens (tokenization), splitting in into phrases, categorizing and tagging of words (part-of-speech tagging) and morphological decomposition. These steps are relevant to analyze the base form of each word present in the text and will be described below.

- Tokenization: this step breaks the text into tokens, which are usually words separated by spaces or punctuation in the text. It is considered the first operation to be performed when processing documents because it is responsible for splitting a stream of characters into words [83]. The entity *word* is considered the most basic kind of token in NLP [84]. The most simple methods used to perform this task are based on splitting the text according to spaces and punctuation. However, several issues might hinder the tokenization efforts like abbreviations, different formats for representing the same information (date, time, address, etc) and words that contain hyphens because it is not clear if it is considered a single or compound word. If this task is not performed correctly, the errors will be propagated through the rest of the NLP analyzes, thus complicating the RE task [85].

- Split of phrases: in order to decompose a text into sentences, it is necessary to identify the boundary of each phrase, which are generally generally delimited by punctuation like the period, question mark and exclamation mark or colon, semi-colon, ellipsis and the hyphen [85]. However this task is not straightforward since it is possible to find these special characters in names or abbreviations in more technical science documents and all those aspects might contribute for an erroneous NLP processing.

- Categorization and tagging of words: this task is also called part-of-speech (POS) tagging, because it is able to assign a grammatical class (e.g., verbs, adverbs, adjectives, nouns) to each word present in the text. These classes are essential for building elementary blocks of categories that play an important role in NLP tasks like machine translation or information extraction [86].

- Morphological decomposition: the goal of this step is to deal with a large number of words which are present in the text in the inflected form. It is necessary to transform these words into a canonical form. Only a restricted class of words (preposition, adjectives or conjunctions) do not suffer inflection. A very useful sub-task in this step is *lemmatization*, which is a conversion process to the root of word, by removing prefixes and suffixes [87]. For instance, the regulate verb, might assume other forms like: *(un-)regulat(-ed)*,*regulat(-ing)*, *regulat(-ed)*, regulat*(-es)* but their root form is *regulat*.

2.8.2.2.2 Syntactic layer This step is able to uncover the possible meanings of the text sentences and also identify how the sentences are structured grammatically [88]. The output of this task is a possible representation of the sentence that identifies the structure of relationships between words. The syntactic layer may be divided in verbal grouping, Shallow parsing and Deep parsing.

- Verbal grouping: aims to identify possible verbal chains in sentences. Verbal chains can be defined by the verbal tenses, verbs followed by infinitive or gerund, simple verb forms or also more complex ones. This step is used mainly to identify the verbal tense in the sentences, to detect if the sentence is on active or passive voice and also the negative verb forms.

  Figure 2.10 shows an example of a sentence where the main preliminary steps on the syntactic layer were performed: split the sentence in tokens, part-of-speech tagging, morphological decomposition and verb chunking.

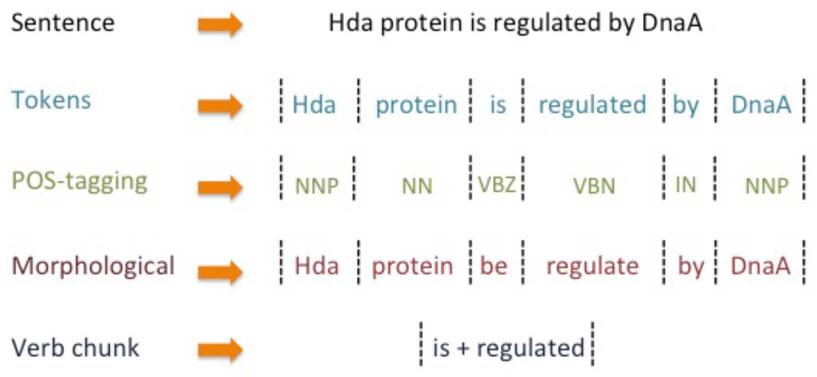


Figure 2.10: Steps needed to perform the Shallow parsing process.

- Shallow parsing: also called *chunking*, this process is responsible for identifying tokens in sentences which were categorized by POS-tagging processing [89][82]. It aims to identify a string of words and their grammatical category in order to build a hierarchical parse tree where the nodes represent a subset of tokens [90]. These subsets of tokens can be classified as sentences (S), nouns or nominal phrases (NP), prepositional phrases (PP), verb phrases (VP) or adjective phrases (AP).

  The main feature of the parsing process is to transform the sentences into a kind of representation that allows the identification of some groups of words and their relations. Figure 2.11, shows an illustration of a parse tree created by a Shallow parsing process.

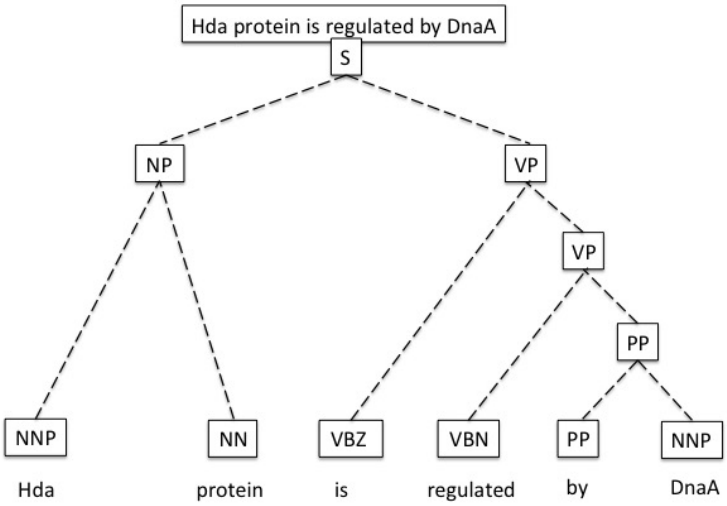


Figure 2.11: This hierarchical tree represents the output of Shallow parsing process, where it is possible to see the classification of tokens which constitute the sentence

- Deep parsing: in some cases, a shallow parser processing is not precise enough. In these cases, a possible solution is to perform a deep analyzes. The reason for the increased complexity over Shallow parsing is because it aims to identify possible sets of words that are grammatically dependent.

  According to Balfourier *et al.* [90], Deep parsing aims to build all possible subsets of elements grouped by juxtaposition that can describe a syntactic category. Each subset represents a segment of the sentence between two categories. The main idea in this approach is to identify the subject and the predicate in the sentences thus uncovering some meaning.

  By performing a deep analyzes of the sentence that was previously used in Shallow parsing, it is possible to see in Figure 2.12 the link between the two segments of the sentence (NP and PP) that represent a probable relation between these two entities.

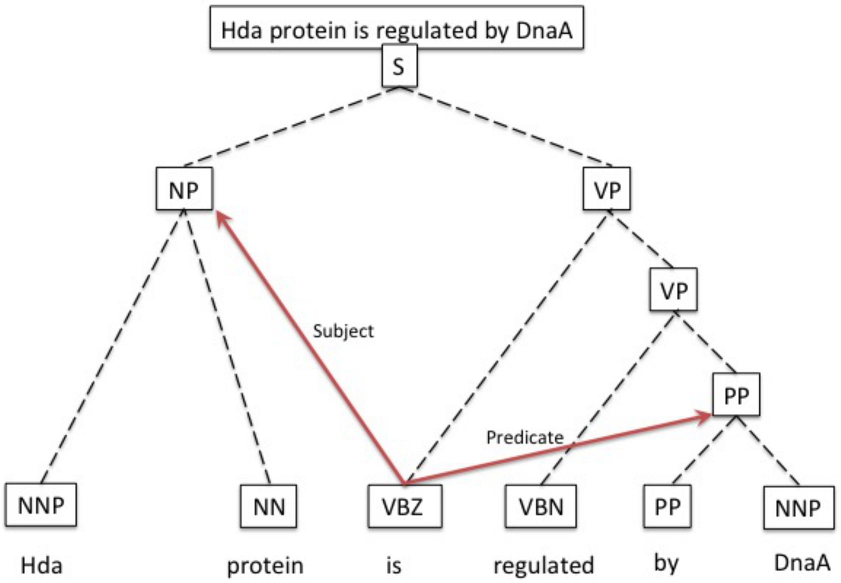


Figure 2.12: Unlike Shallow parsing, in this representation tree, it is possible to identify a relation between the subject and the predicate of the sentence which are bound through the verbal form ”regulated”.

Recently, several techniques have been applied to perform RE, based mainly on co-occurrences, syntactic rules, Machine Learning approaches, statistical methods and ontologies.

The approach based on co-occurrences involves counting for each pair of words in the text, the number of times both of them appear together inside a window of a given size (e.g., composed for instance by 10 words) [91]. Normally this relationship is binary and how many more times the entities are identified along the text, is attributed a superior importance level for it.

For instance, given a part of sentence ”...ArgP, transcription factor, controls the transcription of genes...” where the entities *ArgP* and *transcription factor* are present in the same text unit. It is thus possible that there exists some type of relation between these entities. This type of approach is most commonly used to recognize interactions like: protein-protein, gene-gene, transcription factors-genes and determined proteins related to diseases.

STRING[2] is considered one of the best data sources for studying protein-protein interactions. The method used by STRING in order to perform the relation extraction process is based on a statistical analyzes of co-occurrence in documents and also from other NLP methods. To improve the quality and increase the number of relations that could be extracted, a new scoring method was developed combining the co-occurrences inside the sentences, paragraphs and the entire document [92].

miRTex[3] is a text-mining approach that extracts relationships from the literature based on regulation of miRNA-gene and miRNA-target. One of the methods employed is the use of syntactic rules to identify these relations through using trigger words such as ”regulate”, ”target” or ”suppress”. The words that can indicate a possible regulation are manually selected. Afterwards, a rule is composed by a syntactic pattern and used against the parse tree derived from a sentence [93].

Machine Learning techniques are also commonly used for this task. These techniques can be though as automatic computational procedures based on logical or binary operations, that learn a task from a set of examples [94]. This approach requires a training process in order to improve a specific model that will be used to classify or identify terms in a new set of documents. Normally, it uses a set of documents manually annotated from a specific biological context. A disadvantage of this technique is the need to create a curated corpus, which consumes a large quantity of work time and must be applied to a specific case study.

In the case of relation extraction, it is possible to predefine a set of pattern relations and then this method will be used for classifying new examples [95]. In order to use this technique, it is necessary to have a high quality annotated *corpora* with the specific relationships that will be identified in the text.

One of the solution used to classify and detect these relations are Support Vector Machines (SVM), which is a Machine Learning method that is currently considered a gold-standard approach to perform classifications tasks [96].

[2] http://www.string-db.org/

[3] http://research.bioinformatics.udel.edu/miRTex/

One of the first studies on relation extraction using Machine Learning approaches, was used for extracting relations between genes and their functions. The general idea was to train the system by using a collection of abstracts concerning a given protein and a set of classifiers responsible for identifying interactions and functions about genes. Thus, it was possible to identify a set of words that were useful for discovering certain aspects of protein function [97].

Ontologies have also been applied to the relation extraction field. An ontology is used to define a vocabulary that may be used for describing concepts.

Gruber [98] defines an ontology as an explicit specification of a conceptualization related to a limited knowledge field that is organized in a set of objects which comprise a domain area. Whilst controlled vocabularies only restrict words that can be used in a particular domain, ontologies extend these characteristics and describe a formal specification of terms and their relationships.

An example of an ontology based approach is Textpresso[4], that aims to provide a tool for searching for model organisms in literature sources, and helps database curators identify and retrieve biological entities as well as extract specific biological facts.

This tool might be considered as a text processing system because it performs the splitting of text into sentences, and the sentences into words or phrases and then labeling it using the eXtensible Markup Language (XML)[5] according to their ontology.

Nowadays, the development of Text Mining tools seem to be increasing and the biological area seems to be one of the main targets for these techniques. Several software systems are being developed by using integrated approaches, not only to perform relation extraction but also to apply information extraction techniques to systematically find information about cellular and molecular aspects.

[4] http://www.textpresso.org/

[5] http://www.w3schools.com/xml/

An example is the PLAN2L[6], a web-based system to help researchers retrieve information in a more efficiently way. This approach integrates both Text Mining and information extraction techniques to explore literature information in several levels of granularity. These range from the retrieval of gene characteristics described in multiple documents to important biological relationships that might be interesting for understanding the cellular behaviour [99].

2.8.3 Identifying Relationships

In order to extract relations, it is important to follow several steps which allow the recognition/identification of these relations. As was previously described, a relation is a link between two entities in a sentence. However, if the two concepts are related, they must be somewhat close to each other in the sentence.

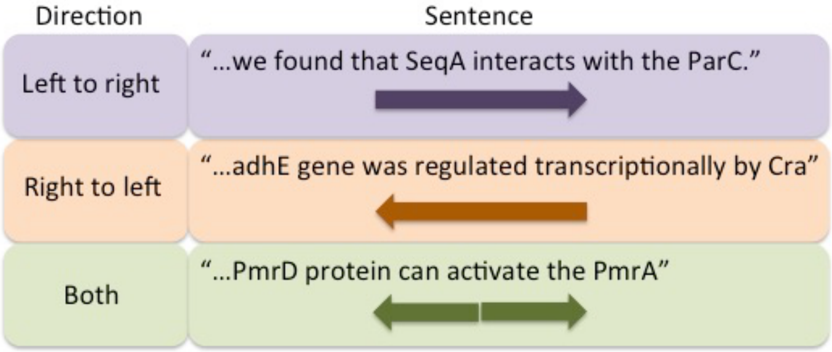
It is necessary to have a clue for identifying a relation. This clue is often a word or a set of words that characterize this relation. When concerning biological events, a clue is identified by certain verbs [100].

It is also important to take into account the polarity of the relation, since it determines if a relationship is negative or positive. In a biological context, the polarity of the sentence is quite important, because it might determine if a given biological entity inhibits/activates the other entity.

Another important detail when characterizing relations is the directionality. It is important to know which entity has been affected by the action. For instance, it is different to say: "protein X inhibits the gene Y" than "protein X is inhibited by gene Y", as it changes the meaning of the sentence (cf. Figure 2.13).

The classification of a sentence is the last step to be performed for characterizing the relationships. It allows the definition of relations which are present in a specific knowledge domain. To perform this task, an ontology can be used to aid in the mapping process between the terms found in a sentence and a specific ontology class.

[6] http://zope.bioinfo.cnio.es/plan2l



Direction	Sentence
Left to right	"...we found that SeqA interacts with the ParC." →
Right to left	"...adhE gene was regulated transcriptionally by Cra" ←
Both	"...PmrD protein can activate the PmrA" ↔

Figure 2.13: In these examples is possible to see the different means which a sentence can obtain according to their direction.

One of the widely used ontologies to describe biological process is the Gene Ontology (GO) [7]. It is a collaborative project whose aim is to provide a consistent description about gene products across several databases.

The main goal of this project is to provide an accurate vocabulary that is curated and structured in order to describe genes and their products in any organism [101].

2.9 The @Note Framework

Over the last few years, the BioSystems research group at the University of Minho and the SilicoLife company have worked together in the biomedical Text Mining field. In this period, a software platform for BioTM called @Note[8], was developed.

A major reformulation, including the development of several novel features has been implemented leading to its current upgraded version (2.0). @Note was developed in Java and uses a MySQL database, which copes with the most important IR and IE tasks and promotes multi-disciplinary research.

@Note integrates three main modules (Publication Manager, Resources

[7] http://geneontology.org

[8] available on http://www.anote-project.org

and Corpora) (cf. in Figure 2.14). The publication manager module handles IR tasks while the resource module and the corpora module handle IE tasks [80].

The main goals of this framework are:

- to facilitate the processes of curation and literature annotation;
- to use already developed models in order to automate tasks like text annotation and document retrieval;
- to configure and use models without any need for programming;
- to translate and validate models and finally to allow developers to extend the application to provide new functionalities

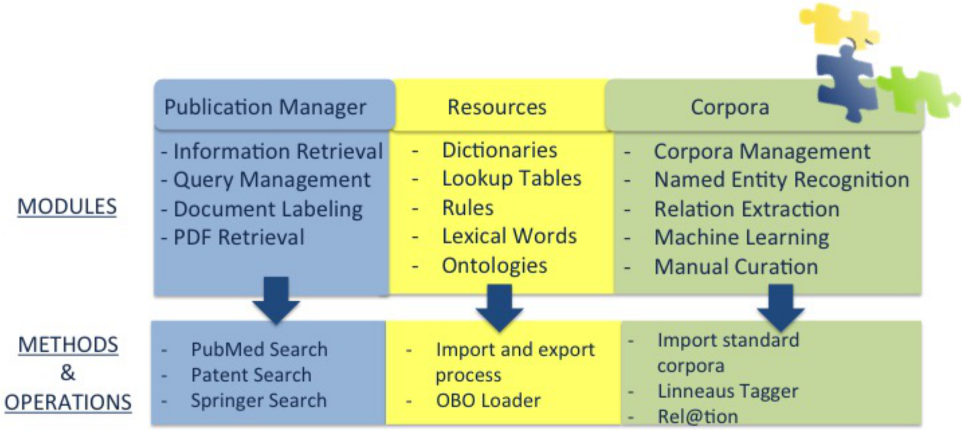


Figure 2.14: Modules that compose the @Note framework.

2.9.1 @Note Information Retrieval

The main goal of the IR systems is to search for documents (full texts, abstracts or sentences) about a given topic of interest. IR is used to perform a query against bibliographical repositories and then retrieve all related documents. Nowadays PubMed is the best-known IR system and is used with this purpose in mind [63]. It is developed to use both Boolean search as well as vectorial models like is described in 2.8.1. The @Note framework uses this approach to retrieve documents from PubMed.

2.9.2 @Note Information Extraction

There is a continuously growing number of scientific documents available in electronic form. Due to this fact but also to the intricacies of human languages, to be able to automatically analyze the content of these texts is still a major challenge in the field of Natural Language Processing. Several efforts have performed in order to cope with this challenge, mainly for developing methods to create systems that extract structured data, identify entities and which allow the extraction of relations.

In order to achieve this goal, the @Note framework takes advantage of a development environment for Text Mining called by Generally Architecture for Text Mining (GATE)[9], which supports several methods for performing Text Mining tasks, like tokenization, sentence splitting, part-of-speech tagging, information retrieval and also to provide access to a linguistic resources like lexicons and ontologies [102].

In order to cope with the Named Entity Recognition (NER) task, @Note implements a dictionary matching software that is able to process multiple documents, by using built-in dictionaries for document tagging.

Figure 2.15 presents the methods that are implemented in @Note to achieve the IE tasks. Regarding the tasks described in subsection 2.8.2, the @Note framework has been developed for handling the tasks of Named Entity Recognition (NER) and Relation Extraction (RE) that will be described bellow.

2.9.2.1 @Note NER process

As was discussed earlier, the aim of this task is to identify and classify several biological entities within a determined class of interest. In order to perform the NER task, the @Note uses a software lexicon-based system, called Lin-

[9] Available at https://gate.ac.uk/

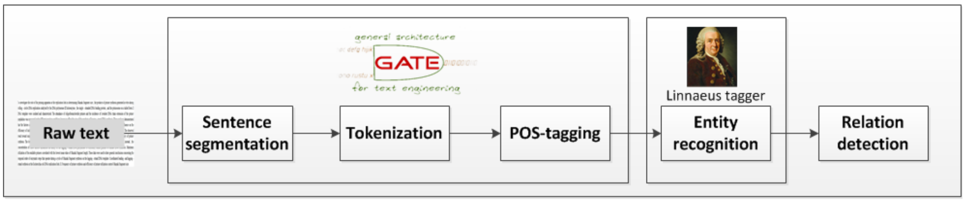


Figure 2.15: Pipeline representation of tasks that compose the methods implemented for IE processing

naeus Tagger[10].

This software was originally developed for species name recognition, and is capable of analysing several types of documents in the biomedical field, and generate several types of output file. It uses a dictionary-based approach which implements an efficient deterministic finite-state automaton for identifying species' names and several heuristic methods for handling ambiguity among terms [75]. This software was added to @Note because it is able to work with both internal or external dictionaries in order to recognize biological entities.

The Linnaeus approach uses a NCBI taxonomy[11] and also a predefined set of species related synonyms in order to create an optimized dictionary for the tagging process. This step will generate a set of regular expressions which will be used to create the automaton.

After that, the automaton created in the previous step will be used together with the documents for the tag processing task. Since it is probable that a large number of ambiguous terms may appear as a result of this task, a set of heuristics rules — detection of acronyms and filters — are used for improving the results.

In the last version of @Note (@Note2) uses a new plug-in that is able to use several lexical resources combined with Linnaeus approach for enhancing the recognition process. It is divided in three main steps: preprocessing, normalization and matching.

The first preprocessing step in @Note allows users to add their own stop-words, which remove the most commonly words used in a given language

[10] http://linnaeus.sourceforge.net
[11] http://www.ncbi.nlm.nih.gov/taxonomy

thus allowing @Note to focus on the relevant words. The second part of the preprocessing step identifies the grammatical class of terms, as well as also is used for the POS-tagging process.

The second step performs a text normalization by removing extra spaces and extra line breaks, inserting missed spaces and inserting spaces between commas and hyphens.

Finally, the last step implemented over this recognition method, performs matches between document terms and the terms stored in the dictionary through the use of regular expressions.

2.9.2.2 @Note Relation Extraction process

In order to successfully characterize a relation, one must perform several steps for accomplishing a relation extraction: determine the boundaries of the phrase, define the clue that will work as a trigger for identifying the relation, normalize the verbal forms by transforming them into a canonical form (the list of entities recognize by the NER process), discover the polarity of the phrases, their direction and classification. Figure 2.16 represents the features needed to characterize a relation.

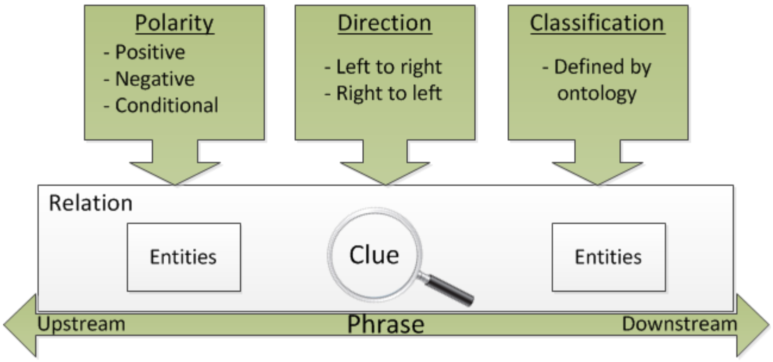


Figure 2.16: The main features needed to represent a relation by using the @Note

In this relation model presented by @Note, the entities are the main component needed for identifying a relation, as well as the clue that plays a role for linking the entities in order to establish a relationship.

Normally, these relationships between entities are often binary (one-clue-one). However, it is possible to characterize relationships that link one entity to many (one-clue-many), many to one (many-clue-one) or many entities to many entities (many-clue-many).

For this step, the model of relation implemented by @Note is only capable of identifying the relationships which have at least two entities present in the phrase.

Concerning the polarity step, @Note implements this task following the same features referred in subsection 2.8.3. However, it adds one more feature for characterizing relations when the verbal form assumes a conditional aspect, for instance using the expressions: *may*, *could* and *should*. In this specific case, the user can either choose to validate these relations or simply discard them.

About the directionality of relations, @Note enables the identification of if it follows the direction of the text (left-to-right) or the opposite one, thus uncovering entities that are event promoters. @Note also includes a new feature for recognizing relations without direction. In order to be able to classify a relation, @Note requires an ontology that can describe a specific domain to be studied, thus all relations identified by @Note will be related to this domain.

The model of relation extraction implemented by @Note is composed by three main steps:

- create a standard schema for annotations that combines the textual and semantic layer;
- relation extraction
- classification.

In order to build the syntactic layer, it is necessary to apply several NLP methods at the textual level: tokenization, phrase splitting, POS-tagging, morphological decomposition and verbal grouping.

In order to retrieve relations, the syntactic and semantic layer from the previous step are finally combined gathering all features needed to characterize a relation (cf. Figure 2.17).

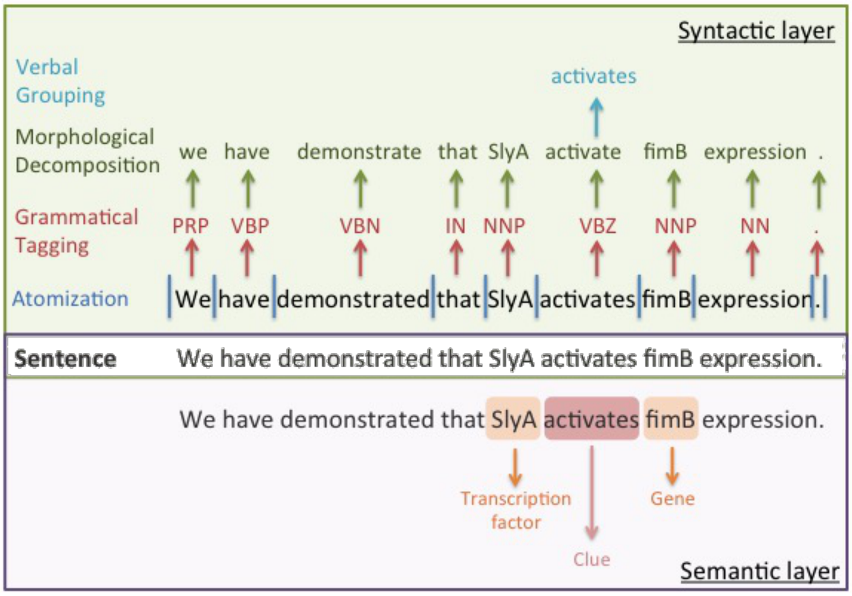

Syntactic layer
Verbal Grouping
activates
Morphological Decomposition
we have demonstrate that SlyA activate fimB expression .
Grammatical Tagging
PRP VBP VBN IN NNP VBZ NNP NN .
Atomization
We have demonstrated that SlyA activates fimB expression .
Sentence
We have demonstrated that SlyA activates fimB expression.
We have demonstrated that SlyA activates fimB expression.
Transcription factor
Gene
Clue
Semantic layer

Figure 2.17: In this figure, it is possible to see the combination of the two layers that compose the main structure used by @Note for characterizing the relations.

After combining these two layers, @Note provides a rule-based approach for extracting relations. The main components used for this objective are: the verbal grouping from syntactic layer and the entities identified in the semantic layer.

Regarding the entities, it is possible to detect if they are positioned upstream or downstream of the verbal grouping. If there are less than two entities associated to the verbal grouping, the relation in question is ignored.

To illustrate this point, an example is given on Figure 2.18 with a phrase containing five biological entities and three verbal groupings. The extraction of each verbal grouping defines the entities that are located upstream and downstream, based on the verbal relative position. The biological entities that comprise a relationship are delimited upstream by the previous verbal grouping (VG) or by the beginning of the phrase and downstream by the verbal grouping immediately next to it or by the ending of the phrase.

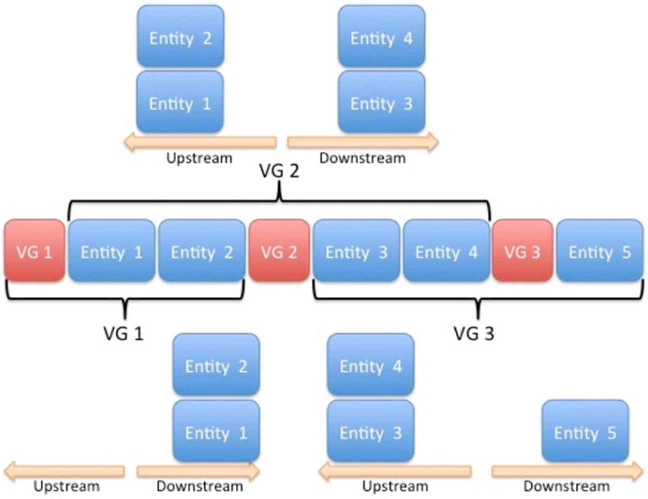


Figure 2.18: An example for identifying relationships in a phrase.

2.10 Summary

This chapter has presented an encompassing overview about the main concepts related to the biological component of this work, addressing the necessary background needed to understand transcriptional regulatory networks and their properties. This chapter also described the concepts of Boolean networks, the main biological databases used nowadays and their features. The literature review has shown that there is no a consensus or standard for sharing biological information with other sources. However it was possible to find and describe some biological data integration approaches. Moreover, the main methods used in the Text Mining field for performing both Information Retrieval and Information Extraction were described, as well as the characterization of relationships. Finally, an overview of the @Note framework was given along with the methods embedded in this software system that allow the definition of a pipeline for relation extraction approaches.

Chapter 3

KREN: Integrated Repository for Biological Data

One of the issues that is frequently addressed in the bioinformatics field is understanding the structure and behaviour of complex molecular interactions that control the cells.

The vast amount and complexity of biological data retrieved in recent years requires an integrated approach, thus forcing scientists to look for novel approaches in order to address this issue. This task is difficult because researchers often need to retrieve information from several databases and work with different data types at the same time.

Nowadays, one of the most relevant topics in the field of bioinformatics is the reconstruction of Transcriptional Regulatory Networks. This activity needs to integrate information from many different sources and is currently one of the most important topics in this field.

The criteria for building this repository was not only the quality of the information found therein but also the ability of retrieving information automatically. Given these criteria, four biological databases were chosen: KEGG, RegulonDB, EcoCyc and NCBI.

This chapter will describe the inconsistencies found in the databases chosen, how the approach for retrieving important information was developed from these databases and finally how the repository was built.

3.1 Biological Database Inconsistencies

The first databases were developed for storing information about genomic sequencing. With the advance of biological techniques and the increase of experimental studies, more databases were created in order to provide this information for the scientific community.

Recently, there is a large number of biological databases that are divided according to their characteristics, the type of information that is stored and also whether they are organism specific. The increasing amount of biological information available in these databases entails a larger complexity in the task of integrating this knowledge.

It is often necessary to retrieve data from several databases because one hardly ever find the information necessary for one's study from a single source. Thus, it is necessary to combine data from these sources into a single repository. Usually, the diversity and complexity of these databases render the task of combining knowledge quite difficult.

The genes are the entities focused in this process, because they are one of the main components of TRNs and also are common in these databases.

Normally, biological databases have their own identifier for genes, proteins, organisms, chemical reactions or compounds. These identifiers are used for accessing several types of information like summary description, nomenclature, gene sequences or protein interactions. Since these identifiers are hardly ever these same in different databases, sometimes even the name of genes may diverge, it is not easy to combine the information.

One way to solve this issue is to find another feature that makes it possible to establish a correspondence. After a careful analysis, one candidate field stood out, the *locus tag*, which is a string that is systematically annotated in every gene in a genome, since it was present in all the databases used for this task.

These tags are being used by the biological researchers to substitute gene names. However this field may sometimes create some confusion since, the same locus tag is genome specific and, therefore references different genes and/or functions depending on the genome. In order to prevent this confusion, the International Nucleotide Sequence Database Collaboration (INSDC)[1] created a register for the locus tag prefix. It is then simply a matter of registering the locus tag prefix before annotating a specific genome.

3.2 Retrieving Information from Biological Databases

Currently, several databases allow users to retrieve information by using Web Services [103]. Most of these applications use some protocols for communication between the client and the Web Service, like SOAP (Simple Object Access Protocol)[2]. One advantage of Web Services is that client-side applications do not need to fully understand the database behind the service itself [104].

Nowadays, scientists are frequently relying on these services for data analysis and several studies have been published using results from these approaches [105].

This task will focus on four databases that offer information that is necessary for this work: NCBI, EcoCyc, KEGG and RegulonDB, briefly described in Table 3.1. These databases were chosen specifically because the information stored therein is important for studies related with TRNs.

Unfortunately, the information provided by these databases cannot be easily integrated since there is no common interface for accessing and extracting information; files retrieved have different structures; and there are inconsistencies in both the number of genes and proteins.

It is essential for this work to gather as much information as possible concerning genes, proteins, transcription units and bibliography references in order to perform the reconstruction of TRNs. Thus, in order to solve the data integration problem, it was necessary to create a repository that compiles all the information gathered from these databases. This repository will be described in the next section.

[1] https://www.insdc.org/

[2] http://www.w3.org/TR/soap

Table 3.1: Biological database chosen to compose the KREN repository

Database	Knowledge domain	Web services
NCBI	Literature, genomic and sequence analysis.	Yes
EcoCyc	Genes, proteins, pathways and molecular interactions.	Yes
KEGG	Pathway maps, human diseases, organisms, biochemical reactions, etc.	Yes
RegulonDB	Knowledge about transcriptional regulation.	Yes

3.3 An approach for Integrating Biological Data

Integrating data from different sources is a very difficult task within the bioinformatics domain. This task is complex partly because of the term ambiguity in the available databases, terminologies and also the way of each database creates their keys to identify information.

This proposal describes a methodology that uses the Web Services available for each database mentioned on the previously section, in order to retrieve information about genes from several organisms. In order to create a pipeline for building the repository, the bacterium *Escherichia coli k-12* was chosen because it is a well studied organism.

3.3.1 Building the Repository

Searching for the information necessary for TRN reconstruction is a hard task because of the difficulties presented in Section 3.1. Besides the main aim of gathering information for TRN reconstruction, the repository also stores other types of information.

Figure 3.1 shows the pipeline used for building the KREN, ranging for information retrieval until data storage.

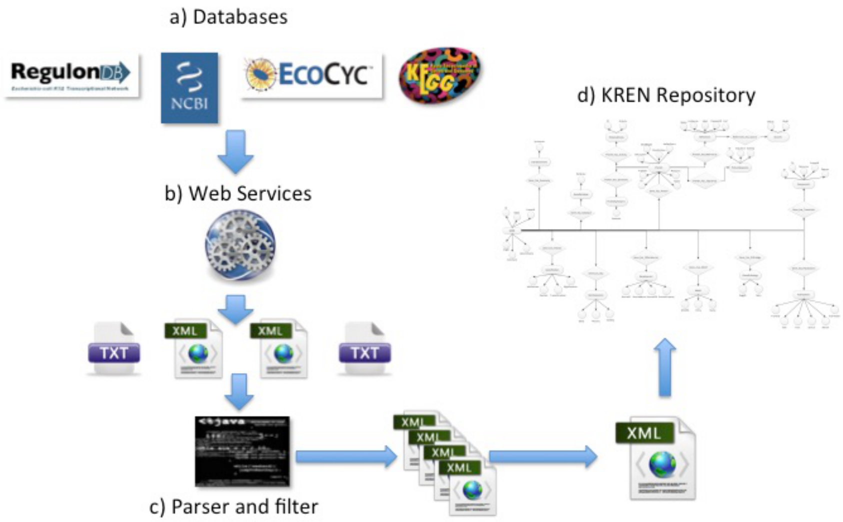


Figure 3.1: Pipeline structure; a) Databases which were chosen; b) Web Services approaches used to retrieve information from the databases; c) A parser and a filter were developed to apply the same structure for all files and select only the necessary information; d) Represents the structure of the repository.

Initially, it is necessary to create a file for each database containing a list of gene identifiers for the organism, since identifiers are database specific. Using the JAVA[3] programming language, a method was implemented to read these files and connect to the Web Services and retrieve all the information available about these genes.

It is important to remember that each database has a different structure and different types of output files. For instance: RegulonDB and KEGG return a text file, NCBI and EcoCyc a XML[4] (eXtended Markup Language) file, thus increasing the complexity of the process of integration.

After the data retrieval process, it was necessary to perform an analysis over the structure of each file and develop a parser to transform the information into XML since this enables users to perform searches and also navigate across the information thus facilitating the integration process.

Storing information into XML files is currently a valid alternative over

[3] https://www.java.com/

[4] http://www.w3.org/XML/

using relational databases. It is becoming a standard way of representing data changes, a form of web publication and also a flexible syntax which allows the same information to be represented in different ways.

Once all data was in the same format, the structure of each file was analyzed in order to create an unique repository able for storing all the information from these databases. Before the data integration can start, it is necessary to perform a data validation. For accomplishing this task, XQuery[5] - a language used to perform queries in XML documents - was used.

During the process of validation, some issues began to appear, such as: there were discrepancies in the amount of annotated genes in each database, gene identifiers were different, the same gene didn't have the same name on all databases and some genes are found in a database and not in others. Since there is an apparent incompatibility between these databases, it was important to search for common features among these files.

For this specific organism, there is an attribute that is shared to all databases, called *B-number* create by Blattner [106], that is commonly used as a gene identifier.

Figure 3.2 shows a relation between the number of genes and *B-numbers* associated in each database. As can be seen, not all genes are related with a *B-number*. Meanwhile it was possible to find almost all genes associated to this identifier in two of the databases; In NCBI that has 4498 genes, 4496 have a *B-number* while in KEGG all genes are associated with a *B-number*.

In the other two databases this inconsistency is more visible. For instance: EcoCyc has a total of 4501 genes, where 230 of these does not have *B-numbers* associated. In RegulonDB, 4496 out of 4639 genes have *B-number*.

Due to the fact that the same *B-numbers* are used for many genes and that some genes do not have one of these identifiers, the task of integration is complex and there is a need of disambiguating this information.

When contemplating data integration, there are other problems beside this one, as can be seen in Figure 3.3 that shows the difference between the number of genes found in these databases.

The large number of coincidences occurs when comparing KEGG and

[5] http://www.w3.org/XML/Query/

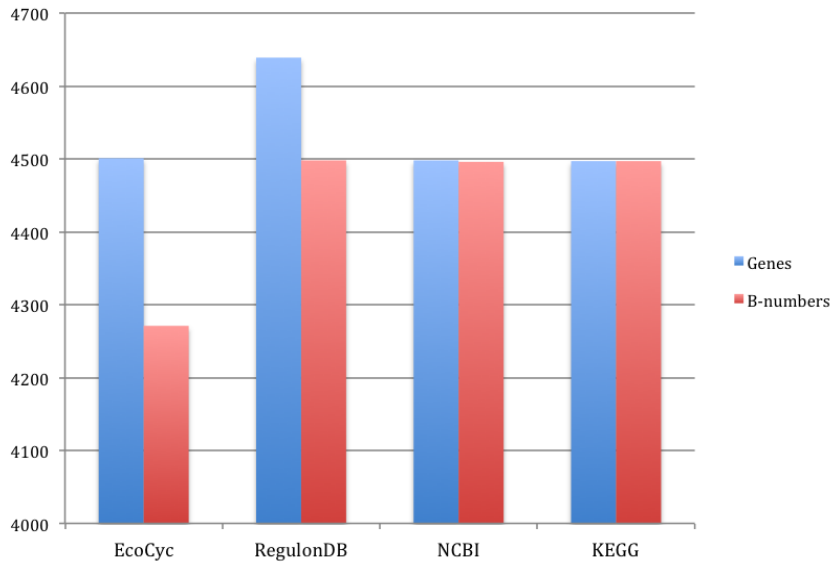


Figure 3.2: Comparison between number of genes and *B-numbers* associated in each database

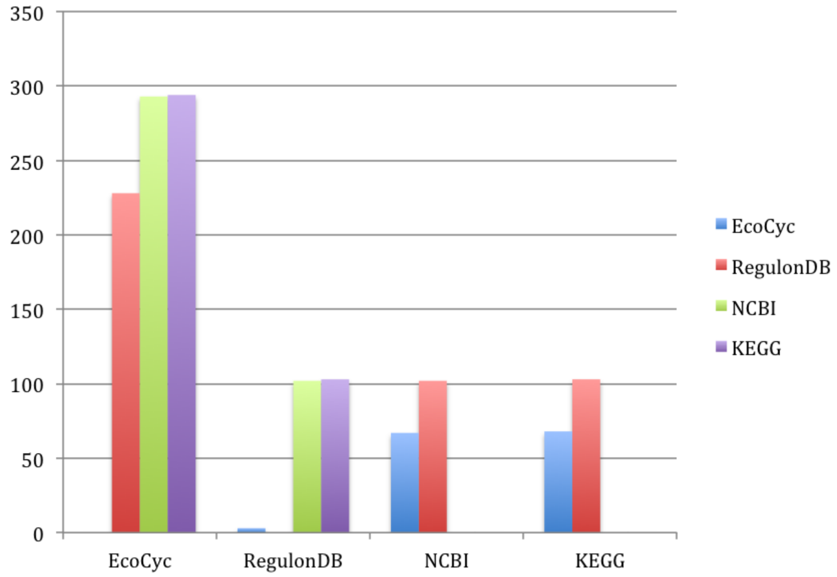


Figure 3.3: Difference between the amount of genes found in each database

NCBI since they both have the same number of genes. The same happens when comparing the EcoCyc database and RegulonDB, since only three genes which are present in EcoCyc do not appear in RegulonDB. The worst case is when comparing KEGG and EcoCyc, since 294 genes are present in KEGG do not seem to be present in EcoCyc.

When contemplating the goal of building the repository, it is necessary to gather the largest amount of information possible, especially about genes, proteins, transcription units and bibliography references, for performing the reconstruction of these networks. Thus, to solve the data integration problem, it was necessary to create a repository, called KREN [4], that compiles all the information gathered from these databases. The main component of this repository is the gene, shown in Figure 3.4.

All information inside this data source is related with this entity and all genes are identified by an unique *B-number* that is common among the databases. For this work, the KREN was used to provide some specific information, such as gene names, protein names, gene synonyms, protein synonyms, transcription factors and also a list of identifiers from the PubMed database, where it is possible to retrieve all scientific publications related with genes from *E. coli*.

This information will be useful to create a Biomedical Text Mining (BioTM) resource like corpus and lexical resources (dictionaries) that will be needed to accomplish the goal of reconstructing regulatory networks.

3.4 Summary

This chapter has presented a development of a new integrated repository for providing information concerning to the genes related to several organisms. The KREN repository retrieves information among four well know biological databases: KEGG, RegulonDB, EcoCyc and NCBI and store it into a XML file. It addresses the process of biological data integration in the field of TRNs, bringing an alternative approach to retrieve, integrate and store information from some of more relevant databases.

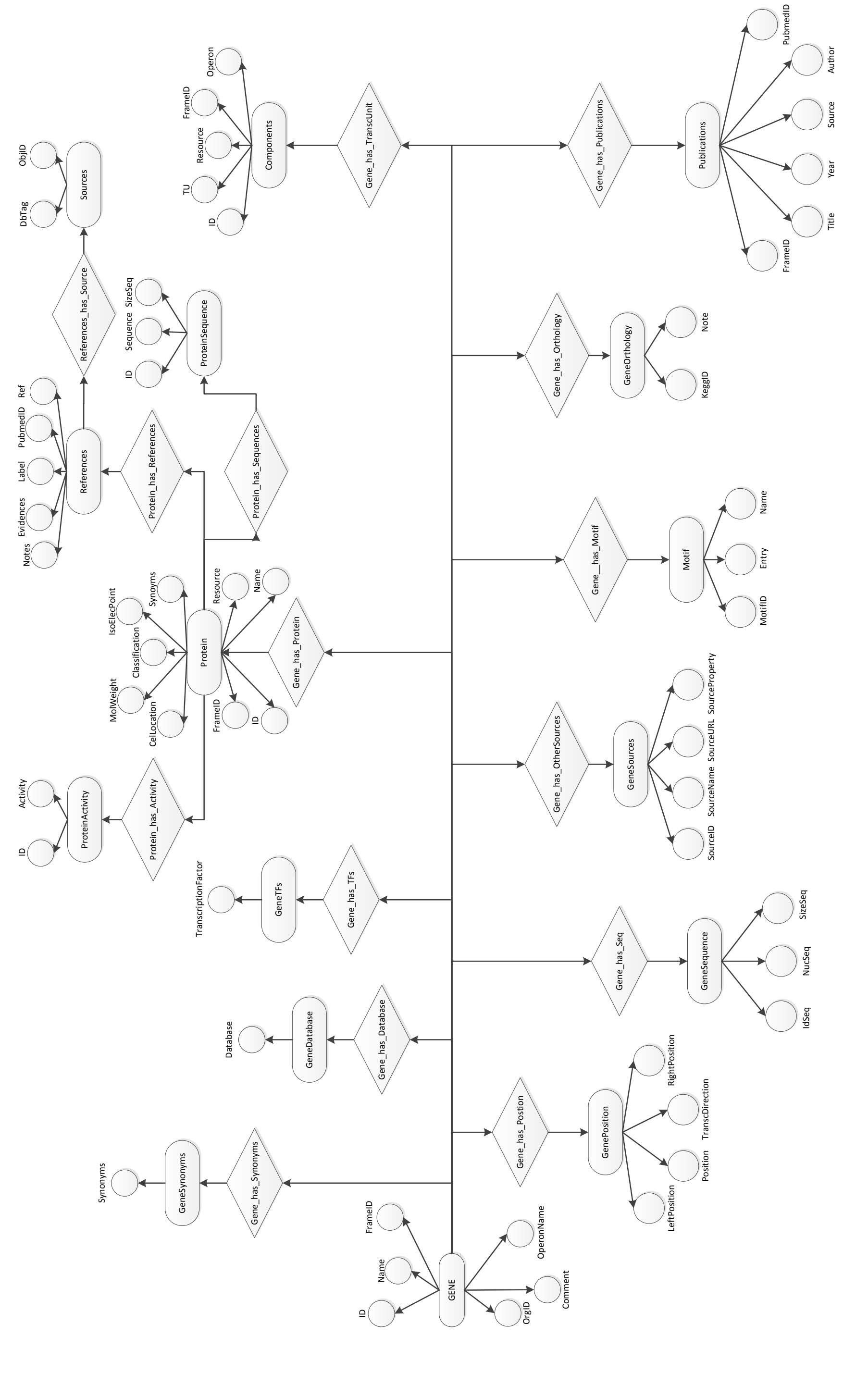


Figure 3.4: Diagram for depicting the structure of KREN repository

Chapter 4

Extending the @Note Framework for Building Transcriptional Regulatory Networks

Over the last few years, the BIOSYSTEMS[1] research group at the University of Minho and the SilicoLife[2] Company have worked together in the Biomedical Text Mining field. In this period a software platform called @Note was developed. It was implemented in JAVA[3] and uses a MySQL[4] database, which copes with the most important functionalities.

The @Note framework aims to help researchers establish Text Mining workflows, to facilitate the curation process and literature annotation and also to use developed models for automating tasks like text annotation and document retrieval. It is an open source project and allows developers to extend it by adding new capabilities.

Despite the many functionalities implemented therein, it was not possible to build regulatory networks using this tool. This chapter covers the

[1] http://www.ceb.uminho.pt/biosystems

[2] http://www.silicolife.com/pt/

[3] https://www.java.com

[4] https://www.mysql.com/

work performed in order to extend the @Note framework for building TRNs, namely how the KREN repository is used for the information retrieval process, building the dictionary that will be used for recognizing the biological entities, the creation of corpora, the process of Named Entity Recognition and finally the extraction of relationships based on regulatory events.

Figure 4.1 depicts the workflow designed in this chapter for building TRNs that starts by using the KREN as the data source and the @Note to implement the Text Mining pipeline.

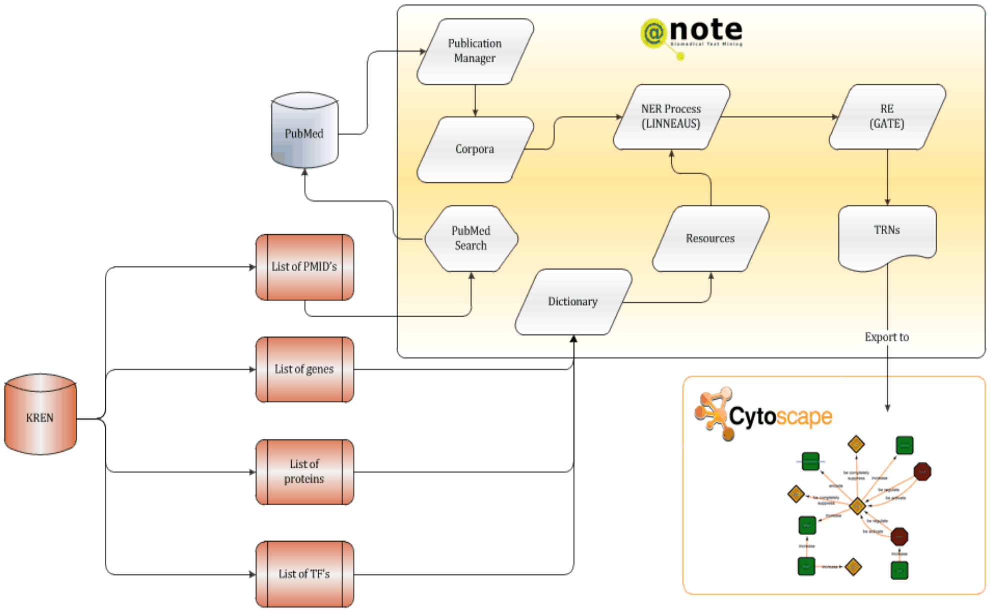


Figure 4.1: Workflow developed for building TRNs

4.1 Retrieving Relevant Information from PubMed

The aim of this task is to provide an efficient information retrieval process which is deemed relevant for a specific query. In this work, the query is related to the genes annotated for a given organism stored on the KREN repository. This section describes the process of retrieving a large amount of scientific papers related to the organisms studied and storing them on the @Note database.

For this propose, the PubMed database was chosen since it is one of the most important sources of available information in the field of the Life Sciences. It provides information concerning several fields in the literature (e.g., titles, authors and abstracts) through the use of Web Services.

The Web Service implemented by PubMed provides a stable interface for information access through using a fixed URL[5] (Uniform Resource Locator) syntax that is able to interpret a set of input parameters into the values necessary for searching and retrieving the requested information [107]. It is implemented using the REpresentational State Transfer (REST) defined by Fielding [108]. It is a type of architecture based on the client-server paradigm, commonly used by Web Services, that provides a uniform interface and makes this communication be as generic as possible [109].

A method developed in this section allows to perform a search for all scientific papers associated to each gene in the target organism through their PubMed identifiers. The first class implemented, *ImportPublicationsFromKREN*, is able to search for all genes in the KREN repository and get the PubMed identifiers associated to these genes. Then, for each PubMed key, a query is performed in the @Note database in order to verify if this publication is already stored; otherwise this paper is downloaded from the source and stored into the @Note database. The classes implemented can seen on Figure 4.2.

The organisms chosen to perform this task were *E. coli K-12* and *Bacillus subtilis 168*. All the corresponding papers are then stored in a corpus on the @Note database, indexed by gene identifier (*locus tag*), thereafter creating the corpora.

33702 papers were retrieved concerning the *E.coli* bacteria and 6715 for the *B. subtilis*. Due to the large amount of papers published about these organisms, only the abstracts of the papers were retrieved from the PubMed database. This decision was taken due to the time-consuming nature of the Text Mining tasks if they were to be applied to the full documents.

[5] https://www.w3.org/TR/url-1/

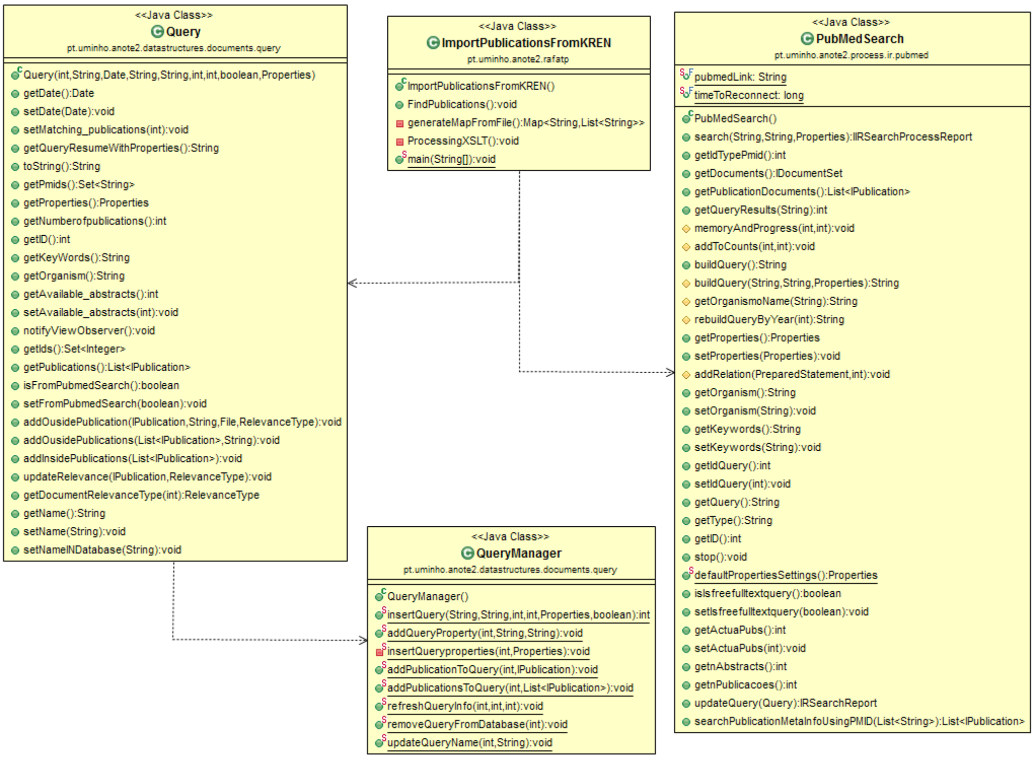

<<Java Class>>
Query
pt.uminho.anote2.datastructures.documents.query
Query(int,String,Date,String,String,int,int,boolean,Properties)
getDate():Date
setDate(Date):void
setMatching_publications(int):void
getQueryResumeWithProperties():String
toString():String
getPmids():Set<String>
getProperties():Properties
getNumberofpublications():int
getID():int
getKeyWords():String
getOrganism():String
getAvailable_abstracts():int
setAvailable_abstracts(int):void
notifyViewObserver():void
getIds():Set<Integer>
getPublications():List<IPublication>
isFromPubmedSearch():boolean
setFromPubmedSearch(boolean):void
addOusidePublication(IPublication,String,File,RelevanceType):void
addOusidePublications(List<IPublication>,String):void
addInsidePublications(List<IPublication>):void
updateRelevance(IPublication,RelevanceType):void
getDocumentRelevanceType(int):RelevanceType
getName():String
setName(String):void
setNameINDatabase(String):void
<<Java Class>>
ImportPublicationsFromKREN
pt.uminho.anote2.rafatp
ImportPublicationsFromKREN()
FindPublications():void
generateMapFromFile():Map<String,List<String>>
ProcessingXSLT():void
main(String[]):void
<<Java Class>>
QueryManager
pt.uminho.anote2.datastructures.documents.query
QueryManager()
insertQuery(String,String,int,int,Properties,boolean):int
addQueryProperty(int,String,String):void
insertQueryproperties(int,Properties):void
addPublicationToQuery(int,IPublication):void
addPublicationsToQuery(int,List<IPublication>):void
refreshQueryInfo(int,int,int):void
removeQueryFromDatabase(int):void
updateQueryName(int,String):void
<<Java Class>>
PubMedSearch
pt.uminho.anote2.process.ir.pubmed
pubmedLink: String
timeToReconnect: long
PubMedSearch()
search(String,String,Properties):IIRSearchProcessReport
getIdTypePmid():int
getDocuments():IDocumentSet
getPublicationDocuments():List<IPublication>
getQueryResults(String):int
memoryAndProgress(int,int):void
addToCounts(int,int):void
buildQuery():String
buildQuery(String,String,Properties):String
getOrganismoName(String):String
rebuildQueryByYear(int):String
getProperties():Properties
setProperties(Properties):void
addRelation(PreparedStatement,int):void
getOrganism():String
setOrganism(String):void
getKeywords():String
setKeywords(String):void
getIdQuery():int
setIdQuery(int):void
getQuery():String
getType():String
getID():int
stop():void
defaultPropertiesSettings():Properties
isIsfreefulltextquery():boolean
setIsfreefulltextquery(boolean):void
getActuaPubs():int
setActuaPubs(int):void
getnAbstracts():int
getnPublicacoes():int
updateQuery(Query):IRSearchReport
searchPublicationMetaInfoUsingPMID(List<String>):List<IPublication>

Figure 4.2: Class diagram compose by main classes for retrieval information from PubMed database.

4.2 Applying the Named Entity Recognition Process

Nowadays, it is possible to find several algorithms for recognizing entities on texts not only in the biological field but also in many others. In this work, is concerned specifically with biological entities. Before applying the NER process, a set of preprocessing steps may be used to facilitate this task, such as removing stop-words or using a part of speech tagging mechanism (Pos-tagging) in order to label the words according to their grammatical features, e.g. verbs, nouns and adjectives, in order to facilitate the process thus avoiding labeling possible biological entity as verbs.

For the NER processing, the @Note framework implements a short version of the Linnaeus [75] algorithm, that was developed mainly to recognize organism names; nowadays it has evolved to an algorithm for general search (e.g., genes and proteins). Listing 4.1 represents a pseudocode for the dictionary matcher method implemented on Linnaeus.

The algorithm starts by creating a sorted list containing all the terms that were retrieved from the dictionary. The next step loops through each document in the corpus (cf. lines 3 through 17). Inside this loop, the document is broken into a list of tokens and their positions.

While there are still tokens to process, the dictionary is searched for the current term increasing it by one token every time until either the term is found or the end of the dictionary is reached.

The loop from lines 9 through 17 handles the search of terms in the dictionary. Initially, the search for the term starts at the beginning of the dictionary but, if the beginning of a term is not found, the search continues with a bigger term (i.e., with more tokens) from the position where the current term would be if it was found in the dictionary. This happens because the `binarySearch` function either returns a positive integer with the position where the term was found or a negative value with the position where the term would have been found if it was in the dictionary.

Listing 4.1: Pseudocode for representing a short version of Linnaeus algorithm

```
1  result = []
2  dic_terms = sorted(dictionary)
3  for text in documents:
4      positions, tokens = enumerate(text.split())
5      pos = 0
6      while pos < len(tokens):
7          lst = pos
8          dic_pos = 0
9          found = False
10         while not found and dic_pos < len(dic_terms):
11             txt_term = tokens[pos : lst]
12             dic = dic_terms[dic_pos:]
13             index = binarySearch(dic, txt_term)
```

```
            if index >= 0:
                result.append(txt_term)
                found = True
                pos = lst + 1
            else:
                dic_pos = - index - 1
                lst += 1
return result
```

An advantage of this approach is the use of a lexicon resource such as the dictionary for entity identification thus providing a convenient way of finding and identifying entities.

Meanwhile, a list of PubMed identifiers for each gene was recovered from the repository, and subsequently, a search on PubMed was performed, getting 33702 papers concerning 4489 genes from *E. coli*, and 6715 papers related with 4421 genes from *B. subtilis*.

This step yielded a corpora, where each gene is associated with a list of publications. Figures 4.3 and 4.4 show the amount of biological entities that were identified for each of these organisms according to their relevance.

Using this information, the following step will be able to create a dictionary that will store the biological entities.

4.3 Using the KREN Repository for building the dictionary in the @Note Framework

Creating a complete dictionary is an essential task because it will allow the @Note framework to recognize every biological entity necessary for building TRNs. However, it is necessary to build a dictionary for each case study, because some of the genes, proteins and transcription factors are specific to each organism. The main idea in this task is to use the KREN repository to provide a way of retrieving all the needed information into a data source and export it onto @Note.

In Figure 4.5 it is possible to see a diagram composed by the main infor-

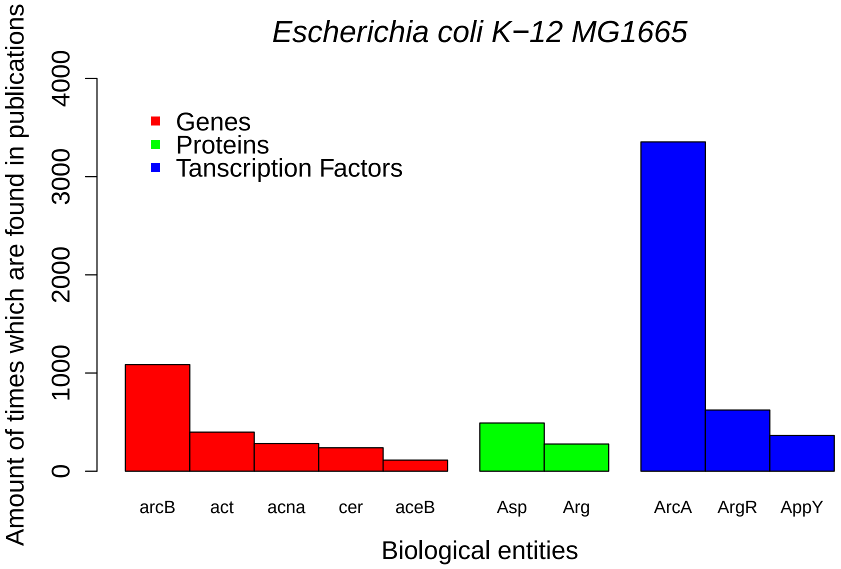


Figure 4.3: Amount of entities identified by NER process for *E. coli* organism according to their relevance

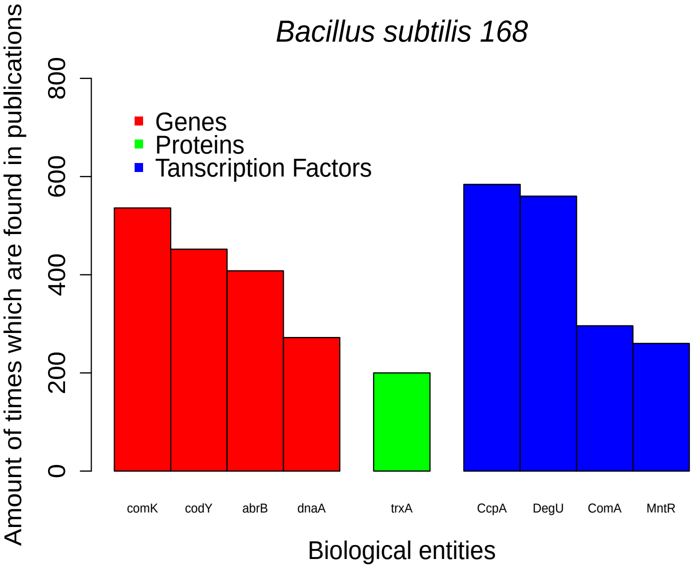


Figure 4.4: Graphic representation for the most amount of entity recognized by NER process for *B. subtilis* organism

mation used from this data source.

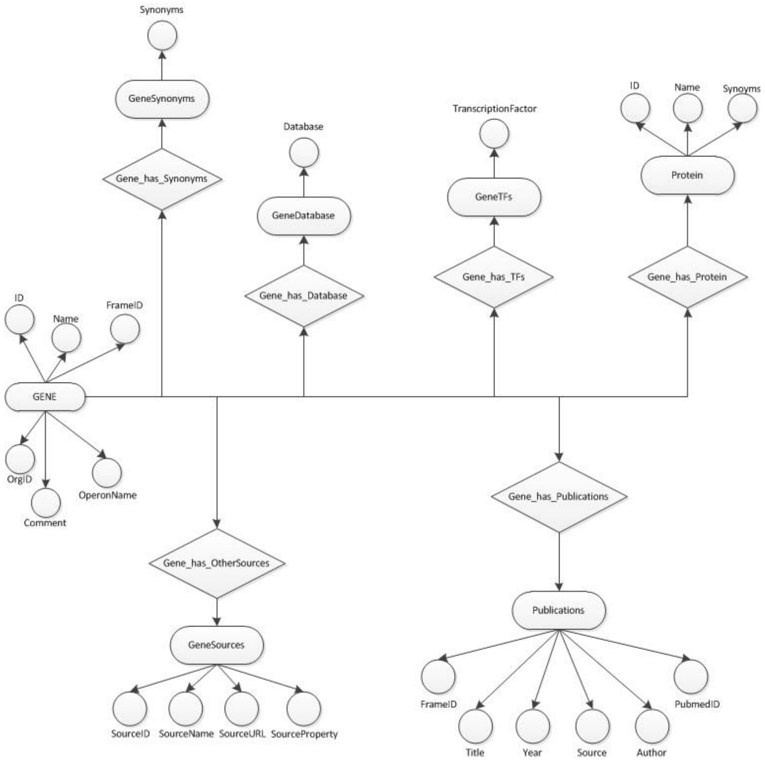


Figure 4.5: It is possible to see in this diagram the information used for building the TRNs: synonyms, database, proteins, transcription factors, other sources of information and publications. It is important to highlight that this information is related directly to each gene.

In order for @Note to recognize the new biological entities, it was necessary to create three new types of resource elements (gene, protein and transcription factor). The main element is the gene, whose unique identifier serves as a key that is associated with the rest of the information.

A method to access the data stored in the KREN repository and load in onto @Note was developed. This method performs a search for terms such as gene names, synonyms for each gene name, protein names, synonyms for each protein name, names of transcription factors and PubMed database identifiers. However, before this information can be loaded on @Note, it is necessary to find all the duplications concerning terms found among the databases that compose the KREN and exclude them.

A class diagram is shown in Figure 4.6, where it is possible to see the methods developed as the KREN class that is able to read the repository and that extends the *DictionaryLoaderHelp* class to create the dictionary. The *IDictionary* interface provides all the operations needed to add, remove and get the biological terms over the dictionary.

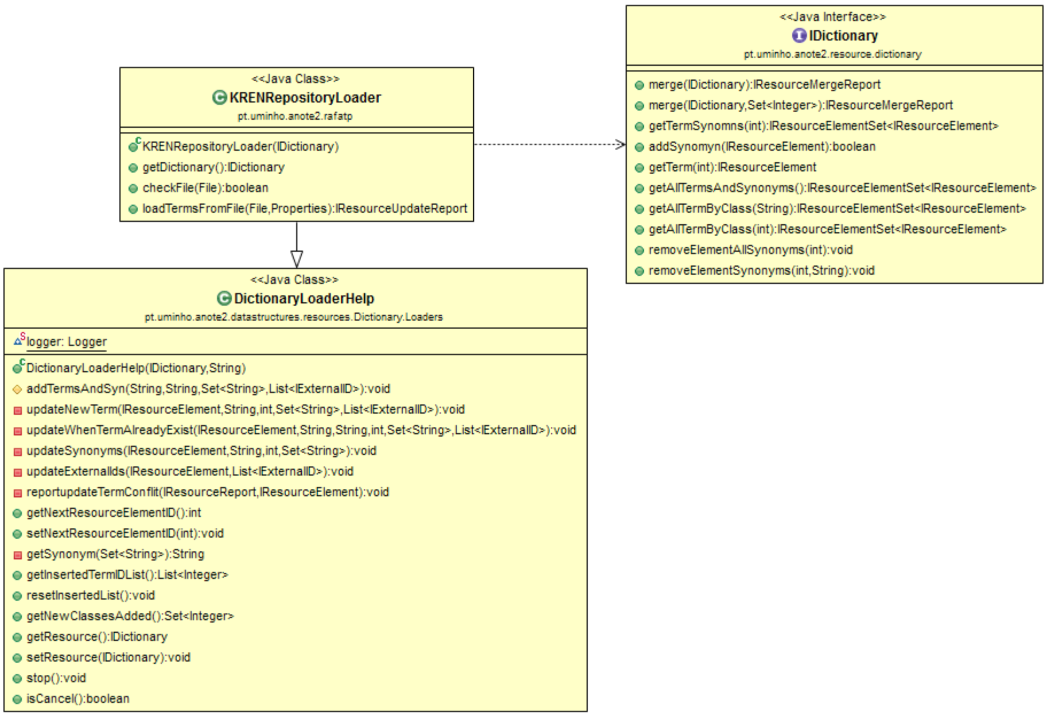

<<Java Class>>
KRENRepositoryLoader
pt.uminho.anote2.rafatp
KRENRepositoryLoader(IDictionary)
getDictionary():IDictionary
checkFile(File):boolean
loadTermsFromFile(File,Properties):IResourceUpdateReport
<<Java Interface>>
IDictionary
pt.uminho.anote2.resource.dictionary
merge(IDictionary):IResourceMergeReport
merge(IDictionary,Set<Integer>):IResourceMergeReport
getTermSynomns(int):IResourceElementSet<IResourceElement>
addSynomyn(IResourceElement):boolean
getTerm(int):IResourceElement
getAllTermsAndSynonyms():IResourceElementSet<IResourceElement>
getAllTermByClass(String):IResourceElementSet<IResourceElement>
getAllTermByClass(int):IResourceElementSet<IResourceElement>
removeElementAllSynonyms(int):void
removeElementSynonyms(int,String):void
<<Java Class>>
DictionaryLoaderHelp
pt.uminho.anote2.datastructures.resources.Dictionary.Loaders
logger: Logger
DictionaryLoaderHelp(IDictionary,String)
addTermsAndSyn(String,String,Set<String>,List<IExternalID>):void
updateNewTerm(IResourceElement,String,int,Set<String>,List<IExternalID>):void
updateWhenTermAlreadyExist(IResourceElement,String,String,int,Set<String>,List<IExternalID>):void
updateSynonyms(IResourceElement,String,int,Set<String>):void
updateExternalIds(IResourceElement,List<IExternalID>):void
reportupdateTermConfilt(IResourceReport,IResourceElement):void
getNextResourceElementID():int
setNextResourceElementID(int):void
getSynonym(Set<String>):String
getInsertedTermIDList():List<Integer>
resetInsertedList():void
getNewClassesAdded():Set<Integer>
getResource():IDictionary
setResource(IDictionary):void
stop():void
isCancel():boolean

Figure 4.6: Class diagram to illustrate the creation of dictionary by @Note

After this task was complete, two different organisms *(Escherichia coli K-12 MG1665* and *Bacillus subtilis* were chosen to extract information from KREN and create the @Note dictionary. The results obtained in this task can be seen in Tables 4.1 and 4.2.

After the dictionary creation, the next step will be the process of extracting regulatory interactions among the biological entities involved in the TRNs.

Table 4.1: Statistics concerning the number of *E. coli* names and synonyms for genes proteins and transcription factors retrieved from KREN

Organism	**Terms**	**Quantity**
Escherichia coli K-12 MG-1665	Gene names	4586
	Gene synonyms	16747
	Protein names	6068
	Protein synonyms	6201
	Transcription Factors	181

Table 4.2: Statistics concerning the number of names and synonyms for genes proteins and transcription factors retrieved from *B. subtilis* retrieved from KREN

Organism	**Terms**	**Quantity**
Bacillus subtilis 168	Gene names	4435
	Gene synonyms	4314
	Protein names	2449
	Protein synonyms	0
	Transcription Factors	157

4.4 Relation Extraction Based on Regulatory Events

Automating the process of extraction relations concerning regulatory events has been a challenge in the Biomedical Text Mining area [110]. Even though there are several attempts focusing mainly in automatic recognition, normalization, and on mapping these biological entities [111], some advanced approaches must be implemented in order to be able to fully performing this task.

The main goal of this step is to use an approach to identify these interactions by using some Natural Language Processing (NLP) method [112] such as Shallow [113] and Deep processing [114] or Dependency parsing [115].

Even tough some of these approaches for identifying these events were already implemented on @Note, it was necessary to adapt these tools in

Verbs			
bind	binding	activate	activated
activates	activating	activation	activate
block	blocking	decrease	decreases
decreasing	down-regulate	down-regulates	down-regulation
encode	encodes	encoding	inactivate
inactivates	inactivating	inactivation	enhance
enhances	enhancing	express	expressing
increase	increases	increasing	induce
induces	inducing	inhibit	inhibiting
inhibition	interact	interacting	interacts
overexpress	positively regulate	positively regulation	positively regulates
promote	promotes	promoting	regulate
regulates	regulated	regulating	stimulate
stimulates	stimulating	stimulated	suppress
suppressed	suppressing	transcription	transcript
transcribed	synthesize	synthesized	synthesizes
up-regulate	up-regulates	up-regulated	unregulated
phosphorylation	phosphorylated	phosphorylates	phosphorylating
phosphorylate	phosphorylation	phosphorylated	unblocked

Table 4.3: Verbal forms that may represent regulatory events

order to allow perform the necessary relation extraction.

The starting point was to make a list of possible triggers (verbs that will be essential to determine an event, e.g.: regulation, inhibition, activation). Currently this list comprises 76 verbal forms (cf. Table 4.3) that may represent these events.

In order to accomplish this goal, it will be necessary to determine/choose interactions or relationships between pairs of entities like gene-gene, gene-protein, protein-protein and TF-gene interactions and then start the process of relation extraction, which is composed by several NLP tasks which will be described next.

The first step is to break the corpora from the previous step into phrases creating the syntactic and semantic layers (cf. Figure 4.7). The syntactic layer is responsible for categorizing the words in the sentence. The semantic layer conveys meaning by characterizing an identification with the biological entities.

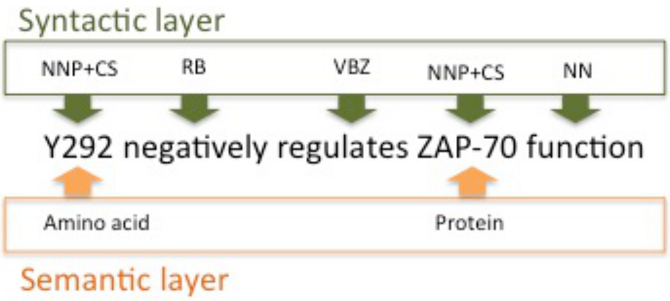


Figure 4.7: Illustration of the syntactic and semantic layers.

The next step, shown in Figure 4.8, involves the extraction and characterization of the biological relationships. It is composed of three main steps:

1. to perform a grammatical tagging based on the syntactic layer;

2. to match the syntactic layer with the morphological analysis; and

3. to extract and characterize the relationships using the verb to identify an interaction.

The relationship is delimited upstream by the previous verbal grouping (VG) or by the beginning of the phrase and downstream by the verbal grouping immediately following to it or by the ending of the phrase, cf. Figure 4.9.

In order to implement this pipeline, a short version of a tool for Natural Language Processing (NLP) that provides several resources to help perform the RE processing called GATE [116] was used. This tool allows the extraction of this type of relationships automatically.

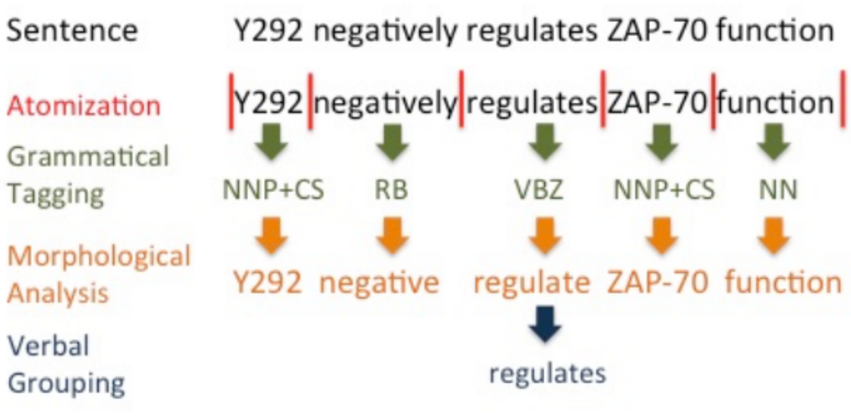


Figure 4.8: Atomization process for extracting possible relationships between biological entities

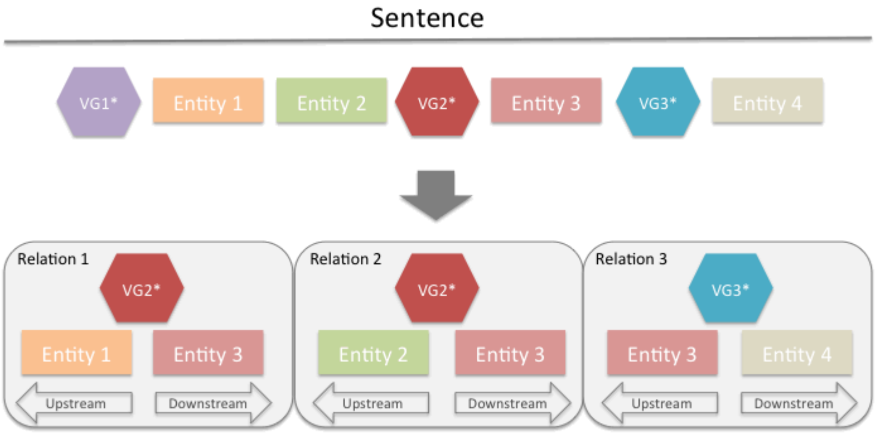


Figure 4.9: Approach to identify relationships in a phrase

In order to evaluate this task, two test cases using a large set of abstracts were performed, one using 33702 related to *E. coli* and the other using 6715 abstracts related with *B. subtilis*. The results obtained in this task can be seen in Table 4.4.

Organism	Amount of verbs(clues)	Number of relations
E. coli	6715	3326
B. subtilis	342	924

Table 4.4: Results from Relation Extraction task performed over *E. coli* and *B. subtilis*

4.5 Visualization of Transcriptional Networks

Through using the graphical interface developed in @Note, it is possible to see all relations extracted in the previous task in a complex table. However, due to the large amount of interactions found and taking into account that it is only possible to see the entities, their position regarding to the verb and the PubMed identifier, it is not feasible to obtain an overview about the connectivity of the network.

In this case, a graphic visualization may help researchers identify important aspects related to the network topology and has helped in studies of evolutionary conservations between individual genes and also identifying the relationship between genes and their products [117].

The goal of providing a way of visualizing TRNs in this work, is to show a large amount of information in a form that help visualize the interactions among biological entities (genes, proteins and transcription factors) in a way that can be readable and understandable by researchers.

To address this restriction identified in the @Note framework, a method was developed that creates a network visualization model from the generated data by the RE task, which can be loaded onto Cytoscape[6]. Cytoscape is an open source application that was developed manly for representing and integrating biomolecular interactions and network states [118].

In order to accomplish this task, a new feature was added to @Note that is capable of generating a structured file from the RE process that can be exported to Cytoscape. The RE output file is transformed into a graphic language based on eXtensible Markup Language (XML)[7] called eXtensible

[6] http://cytoscape.org

[7] https://www.w3.org/XML/

Graph Markup Modeling Language (XGMML) [119] which is interpreted by Cytoscape for creating the interaction network.

A XGMML file describes a graph structure, where the root element is used a *Graph* tag and which may contain nodes, edges and attributes (cf. Listing 4.2). Line 1 defines the XML version used to create the file and the standard encoding. The example defines a directed graph in line 2 identified by the *id* and the label and also defines the *namespaces* (xmlns) used by Cytoscape. The Cytoscape metadata corresponds to lines 3 to 12 inside a description in RDF[8] that defines a standard model for networks created by this application. Line 13 identifies the name of network and properties like interaction and type. The node that represents the protein is defined by lines 14 until 18. On the other hand, a gene is described by lines 19 through 23, where each node has associated an identifier used to identify the interaction event between these two entities, one of them classified as a source and other as the target (cf. line 24). Lines 25 through 30 are used to describe the interaction itself. The graph resulting from this file can be seen in Figure 4.10.

Listing 4.2: Basic structure of the XGMML file

```
 1 <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
 2 <graph id="74" label="Example" directed="1"
       cy:documentVersion="3.0" xmlns:dc="http://purl.org/dc/
       elements/1.1/" xmlns:xlink="http://www.w3.org/1999/xlink"
       xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
       xmlns:cy="http://www.cytoscape.org" xmlns="http://www.cs.
       rpi.edu/XGMML">
 3   <att name="networkMetadata">
 4     <rdf:RDF>
 5       <rdf:Description rdf:about="http://www.cytoscape.org/">
 6         <dc:type>Protein-Protein Interaction</dc:type>
 7         <dc:title>Example</dc:title>
 8         <dc:source>http://www.cytoscape.org/</dc:source>
 9         <dc:format>Cytoscape-XGMML</dc:format>
10       </rdf:Description>
11     </rdf:RDF>
```

[8] https://www.w3.org/RDF/

```
</att>
<att name="Example" value="Interaction" type="string"/>
<node id="88" label="Protein">
  <att name="shared name" value="Protein" type="string"/>
  <att name="name" value="Protein" type="string"/>
  <att name="selected" value="0" type="boolean"/>
</node>
<node id="86" label="Gene">
  <att name="shared name" value="Gene" type="string"/>
  <att name="name" value="Gene" type="string"/>
  <att name="selected" value="0" type="boolean"/>
</node>
<edge id="90" label="Gene (interaction) Protein" source="86" target="88" cy:directed="1">
  <att name="shared name" value="Gene (interaction) Protein" type="string"/>
  <att name="shared interaction" value="interaction" type="string"/>
  <att name="name" value="Gene (interaction) Protein" type="string"/>
  <att name="selected" value="0" type="boolean"/>
  <att name="interaction" value="interaction" type="string"/>
</edge>
</graph>
```

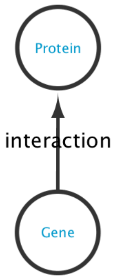


Figure 4.10: Example of a view from Cytoscape over the example XGMML file

Concerning the attributes which might be used in a XGMML file for describing and adding features to the networks, it is important to highlight those used by @Note. Each node belongs to a class of biological entities (gene, protein or transcription factor). Moreover, it is possible to add to a node their synonyms, external links for other databases and also graphic features like the geometric form which represents a determined class (e.g., diamond for a gene, square for a protein and octagon for a transcription factor), the color of each node is also defined by the class, genes being represented by blue, proteins are green and transcription factors are pink.

It is also possible to describe some attributes concerning nodes and edges beyond the source and target entities. It is possible to label the edge with the verb found in the RE process for identifying the relationship and also to associate to this edge the sentence that was used for recognizing this relationship along with the PubMed identifier. Several classes are implemented on @Note to deal with this exporting process, as can seen in the class diagram (cf. Figure 4.11). This process retrieves the information from the RE task that is stored on the @Note database, through using the *REProcessToNetwork* class and then writes the XGMML file.

In order to exemplify this task, a entire network from *E. coli K-12 substrain MG1665* was created, as can seen in Figure 4.12. It is composed by 3326 relationships retrieved from 33702 publication abstracts.

One of the advantages of this visualization approach is to provide several features about the network, as can seen in Figure 4.13. It is possible to find some properties related to the nodes like the shared name of each node, their class (gene, protein or transcription factor), common synonyms for the name and also external identifiers to other databases. Regarding the edges, it is also possible to find some properties such as the type of interaction recognized by the RE process as well as the PubMed identifier associated to the sentence which was retrieved corcerning the regulatory event found, thus facilitating the search for inconsistencies.

An important issue that appears in this task is related to the scientific papers which address some experiments with gene mutations, as is the case of MarA and SoxS whose edges are highlighted in red in Figure 4.14, where two edges were identified that correspond to two different interactions. This happens because the paper returns different results (*do not activate* and

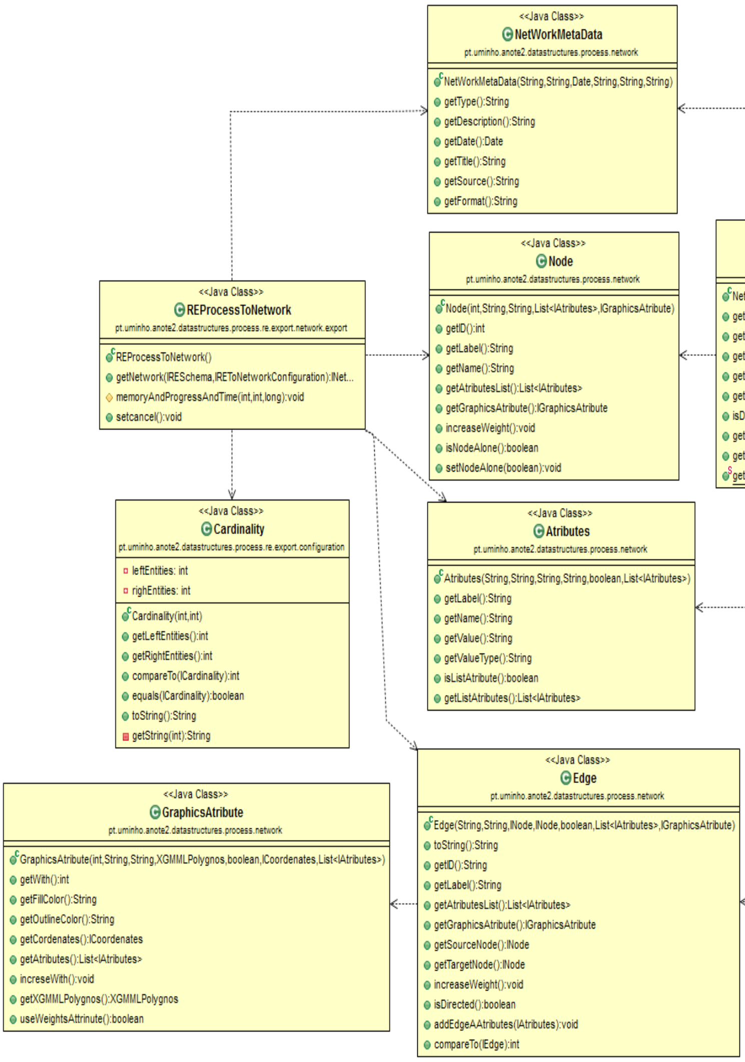

<<Java Class>>
NetWorkMetaData
pt.uminho.anote2.datastructures.process.network
NetWorkMetaData(String,String,Date,String,String,String)
getType():String
getDescription():String
getDate():Date
getTitle():String
getSource():String
getFormat():String
<<Java Class>>
Node
pt.uminho.anote2.datastructures.process.network
Node(int,String,String,List<IAtributes>,IGraphicsAtribute)
getID():int
getLabel():String
getName():String
getAtributesList():List<IAtributes>
getGraphicsAtribute():IGraphicsAtribute
increaseWeight():void
isNodeAlone():boolean
setNodeAlone(boolean):void
<<Java Class>>
Network
pt.uminho.anote2.datastructures.process.network
Network(String,List<INode>,List<IEdge>,String,String,bool...
getID():String
getLabel():String
getVersion():String
getMetaData():INetworkMetaData
getAtributesList():List<IAtributes>
isDirected():boolean
getNodes():List<INode>
getEdges():List<IEdge>
getNetWorkExample():INetwork<INode,IEdge>
<<Java Class>>
REProcessToNetwork
pt.uminho.anote2.datastructures.process.re.export.network.export
REProcessToNetwork()
getNetwork(IRESchema,IREToNetworkConfiguration):INet...
memoryAndProgressAndTime(int,int,long):void
setcancel():void
<<Java Class>>
Cardinality
pt.uminho.anote2.datastructures.process.re.export.configuration
leftEntities: int
righEntities: int
Cardinality(int,int)
getLeftEntities():int
getRightEntities():int
compareTo(ICardinality):int
equals(ICardinality):boolean
toString():String
getString(int):String
<<Java Class>>
Atributes
pt.uminho.anote2.datastructures.process.network
Atributes(String,String,String,String,boolean,List<IAtributes>)
getLabel():String
getName():String
getValue():String
getValueType():String
isListAtribute():boolean
getListAtributes():List<IAtributes>
<<Java Class>>
GraphicsAtribute
pt.uminho.anote2.datastructures.process.network
GraphicsAtribute(int,String,String,XGMMLPolygnos,boolean,ICoordenates,List<IAtributes>)
getWith():int
getFillColor():String
getOutlineColor():String
getCordenates():ICoordenates
getAtributes():List<IAtributes>
increseWith():void
getXGMMLPolygnos():XGMMLPolygnos
useWeightsAttrinute():boolean
<<Java Class>>
Edge
pt.uminho.anote2.datastructures.process.network
Edge(String,String,INode,INode,boolean,List<IAtributes>,IGraphicsAtribute)
toString():String
getID():String
getLabel():String
getAtributesList():List<IAtributes>
getGraphicsAtribute():IGraphicsAtribute
getSourceNode():INode
getTargetNode():INode
increaseWeight():void
isDirected():boolean
addEdgeAAtributes(IAtributes):void
compareTo(IEdge):int

Figure 4.11: Class diagram used to export networks to Cytoscape

activate). Thus, taking into account the sentence, it is possible to assume that the edge associated to the *do not activate* relation refers to a mutation process and also classify this regulatory event as "strongly as", the sentence is identified by the PubMed key (20008776) and states *"...To understand why MarA does not activate certain promoters as strongly as SoxS, we compared MarA , MarA mutants , and SoxS..."*.

Until now, every biological entity was recognized by the RE process whether this entity was identified as a mutant or not. In order to solve

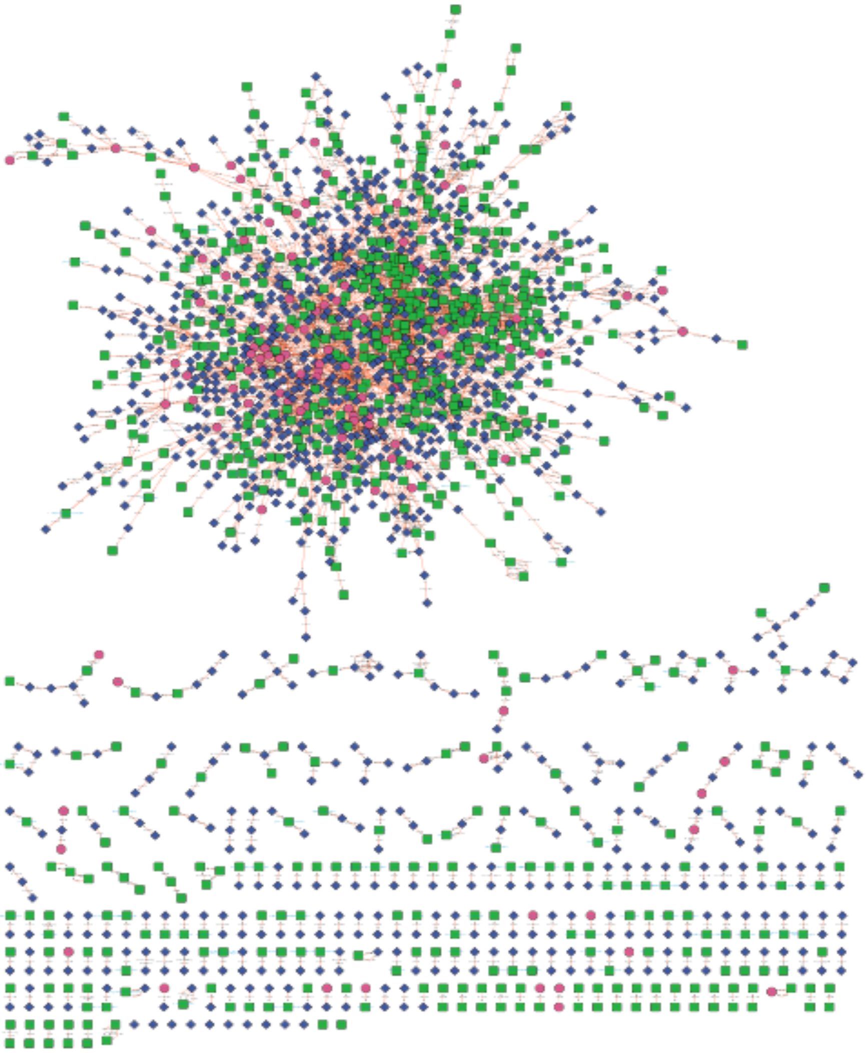
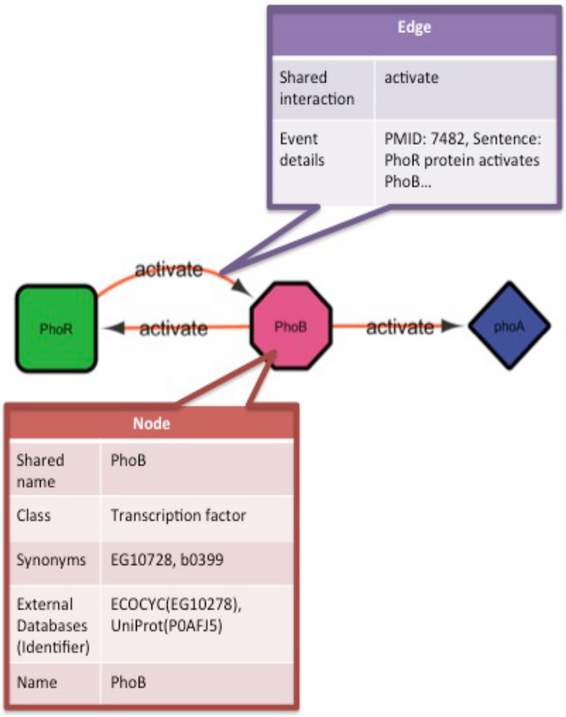


Figure 4.12: Network resulting from the exporting process performed by @Note

this issue, a post-processing over the RE process was performed in order to identify and delete interactions that present mutants or adverbs such as the case described above. The text classifier creates a copy of the corpus and searches for sentences that might contain the mutant word and their variations, and searches for text fragments containing adverbs like *"as strongly as"* or *"more than"*. If these conditions are met, the method inserts a new property in the database to label this interaction as dubious. Thus, when



Edge	
Shared interaction	activate
Event details	PMID: 7482, Sentence: PhoR protein activates PhoB...

Node	
Shared name	PhoB
Class	Transcription factor
Synonyms	EG10728, b0399
External Databases (Identifier)	ECOCYC(EG10278), UniProt(P0AFJ5)
Name	PhoB

Figure 4.13: Some features which are shown over the network visualization

the RE process is performed, all interactions assigned with this property will not appear in the visualization step. Figure 4.15 shows the class diagram designed for the text classifier task. The result of this task is a network that is more reliable and with less noise.

One way of representing this type of networks is by using Boolean rules.

It is assumed that each element of system has a binary state, where *true* corresponds to the element being active and *false* corresponds to it being inactive. The next section will describe the steps needed to perform the process of transforming these networks into Boolean rules.

4.6 Extracting Boolean Rules

In order to predict a regulatory network for *E. coli*, it is necessary to analyze several Boolean rules that determine when a gene will be activated or

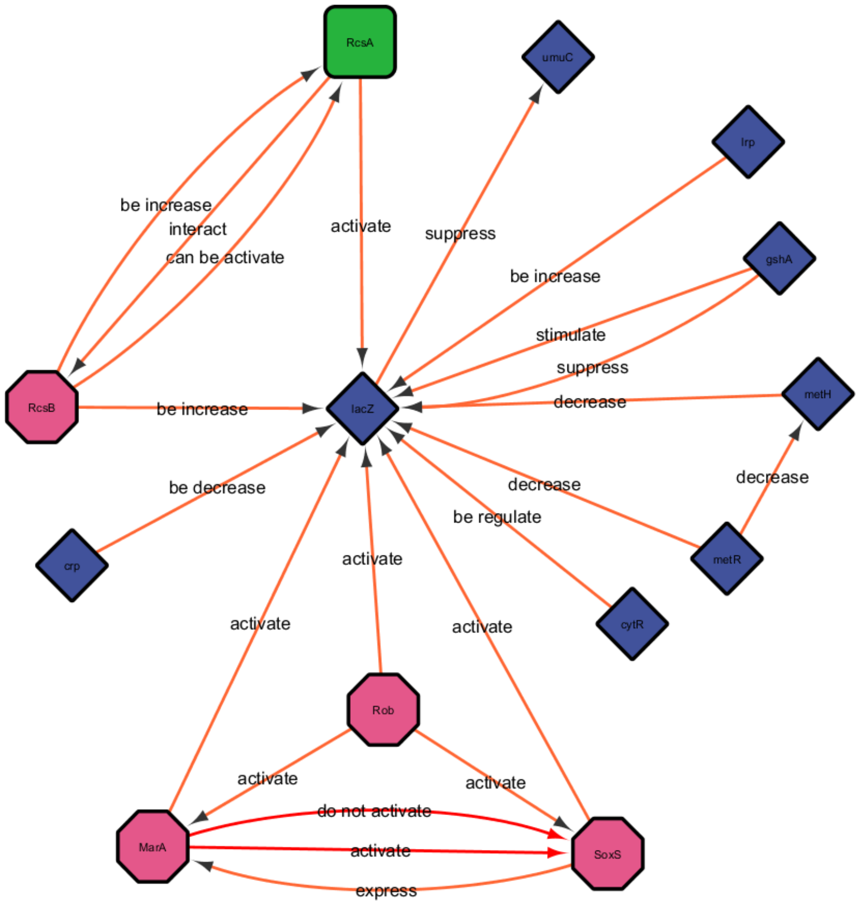


Figure 4.14: Inconsistencies found among the edges of the *lacZ* network

not. The model published by Covert et. al [120] provides a set of these rules, one associated with each gene. However, some rules defined for this regulatory model also include information about environmental conditions. Unfortunately, this work is not currently able of dealing with this type of conditions.

As an example, the rule, shown on Equation 4.1, which activates the gene

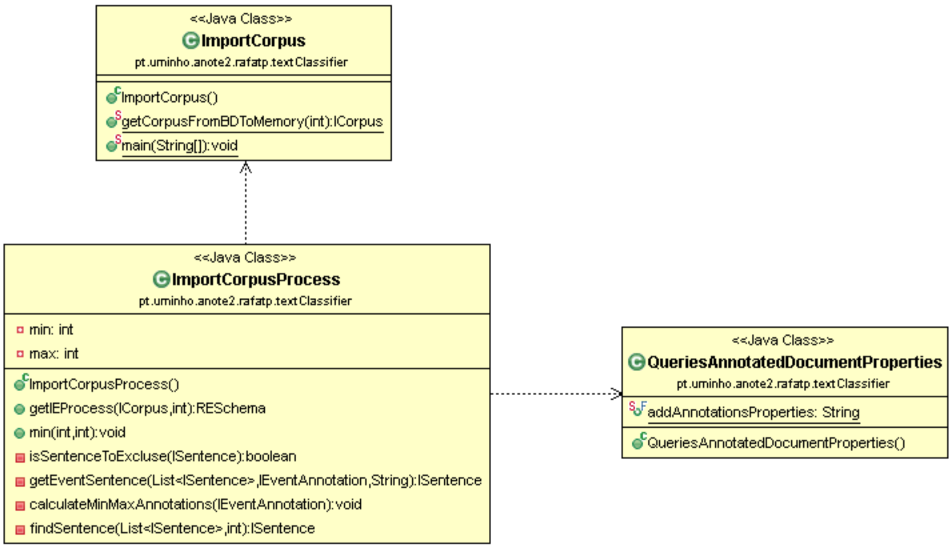

<<Java Class>>
ImportCorpus
pt.uminho.anote2.rafatp.textClassifier
ImportCorpus()
getCorpusFromBDToMemory(int):ICorpus
main(String[]):void
<<Java Class>>
ImportCorpusProcess
pt.uminho.anote2.rafatp.textClassifier
min: int
max: int
ImportCorpusProcess()
getIEProcess(ICorpus,int):RESchema
min(int,int):void
isSentenceToExcluse(ISentence):boolean
getEventSentence(List<ISentence>,IEventAnnotation,String):ISentence
calculateMinMaxAnnotations(IEventAnnotation):void
findSentence(List<ISentence>,int):ISentence
<<Java Class>>
QueriesAnnotatedDocumentProperties
pt.uminho.anote2.rafatp.textClassifier
addAnnotationsProperties: String
QueriesAnnotatedDocumentProperties()

Figure 4.15: Class diagram used to develop the text classifier task

sodA is given below:

$$sodA : (\mathrm{NOT}(ArcA\ OR\ Fur)\ OR\ (MarA\ OR\ Rob\ OR\ SoxS)) \tag{4.1}$$

For building a generic model for TRNs, it is necessary to create a small part of the entire network, only for the genes which are present in a given rule. Cytoscape provides a method for searching for the first neighbours of a node. Using this feature a network was created for each node present in the rule and then a merge process, also provided by Cytoscape, was performed. The result is shown in Figure 4.16, where the edges highlighted in red correspond to the rule. Thus, it is possible to compare the network reconstructed and the rule defined in Covert's model.

Using the information present on the XGMML file created for the network visualization on Cytoscape, a method was developed to generate all Boolean rules from it. Firstly, the verbs used for identifying the regulatory events were divided into two categories: one for representing the activation state and another for inhibition. The number 1 was assigned for the activation state while 0 was used for representing inhibition states.

For accomplishing this task, three classes were developed, cf. Figure 4.17.

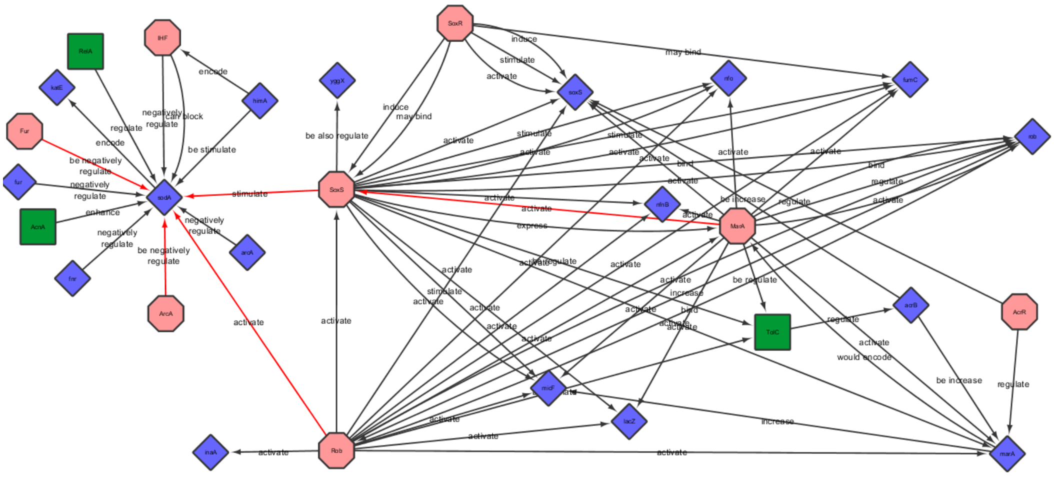


Figure 4.16: Small part of the Reconstruction of Transcriptional Regulatory Network from *Escherichia coli K-12 MG1665*

The first one, *xgmml2csv*, is able to read the XGMML file and write it in a CSV file that describes the nodes found in the network, the edges and also the action for each relationship, as shown in Listing 4.3. The first number is related to the edge identifier and is followed by the source node, the target node and the verb that identifies the regulatory event.

Listing 4.3: CSV file created from the XGMML file

```
3,yiiP, FieF, encode
2,ArcA, sodA, be negatively regulate
10,OxyR, Fur, activate
1,Fur, fhuF, regulate
7,Fur, YggB, activate
6,Fur, Cfa, activate
5,Crp, YggB, activate
4,Crp, Cfa, activate
9,SoxR, ArcA, regulate
8,Fur, fhuA, be express
13,IscR, Fur, to bind
11,Fur, FtnA, be induce
12,Fur, ftnA, be induce
```

After this step, a structure is created with this data and it is passed as a parameter to the *CreateBooleanRules* class, that is responsible for identifying

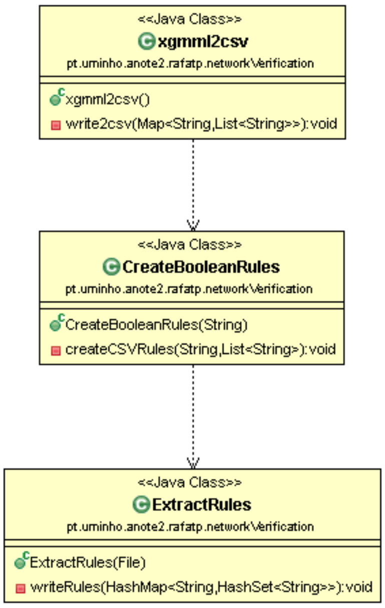

<<Java Class>>
xgmml2csv
pt.uminho.anote2.rafatp.networkVerification
xgmml2csv()
write2csv(Map<String,List<String>>):void
<<Java Class>>
CreateBooleanRules
pt.uminho.anote2.rafatp.networkVerification
CreateBooleanRules(String)
createCSVRules(String,List<String>):void
<<Java Class>>
ExtractRules
pt.uminho.anote2.rafatp.networkVerification
ExtractRules(File)
writeRules(HashMap<String,HashSet<String>>):void

Figure 4.17: Class diagram of the objects responsible for the creation of Boolean rules from XGMML file

the verb and compare with the already defined list to associate the Boolean state.

Finally, the *ExtractRules* class takes into account the target nodes and creates a list (without repeated entries) associated to their source nodes, thus resulting in a file composed by a set of Boolean rules (cf. Listing 4.4).

Listing 4.4: An example of Boolean rule extracted from the XGMML file

```
ftnA = Fur
fhuA = Fur
FtnA = Fur
ArcA = SoxR
YggB = Fur OR Crp
fhuF = Fur
sodA =  NOT ArcA
Cfa = Fur OR Crp
FieF = yiiP
Fur = OxyR OR IscR
```

Despite the advances in the field of Systems Biology, it is still difficult to predict all the possible outcomes of the cellular behavior. A possible solution is to integrate different types of biological networks, such as related with metabolic and regulatory events in order to improve the simulation of cellular behavior.

4.7 Summary

In this chapter, a Biomedical Text Mining pipeline implemented on @Note was described. This pipeline includes the process of extracting abstracts from the scientific literature stored on the PubMed database and the use of the KREN repository as a data source for building the dictionary for biological entities. Moreover, it described the Named Entity Recognition to identify the biological entities within the abstracts and shows a short version of the Linnaeus algorithm implemented to solve this task. After this step, the Relation Extraction process based on regulatory events by using the Natural Language Processing methods was introduced. Finally, this chapter also described the development of a method for visualizing the TRNs created and the process of extracting Boolean rules from these networks.

Chapter 5

Case Study

The case study described in this chapter involves the creation of a regulatory network by using the developed approach. For accomplish this task, the bacterium *Escherichia coli k-12 MG1665* was chosen because it is a well-known organism. The results found in this task will be compared with information found in databases related to regulatory events as well as the models already published.

5.1 Analyzing the protein complexes

This first analysis is performed to evaluate the findings of the developed approach regarding to specific complexes of proteins presented in *E. coli.* Protein complexes are groups of associated proteins which determine certain cellular functions [121] and the arrangement of these entities might generate new rules concerning regulatory events.

Proteins might be involved in several chemical reactions that lead change the original cell behavior because of external stimuli from environmental conditions and also several proteins catalyze biochemical reactions which form the base of the cellular process known by metabolism [122]. The main objective of this complex is to provide for an organism a mechanism to sense and respond to changes in different environmental conditions [123].

In the case of *E. coli*, these groups of protein are described on the EcoCyc database. Since 2011, the EcoCyc database has improved the information about two-component signal transduction, providing a new update representation that includes environmental signals and a curated description in order to provide pertinent information based on the published literature [124].

Table 5.1 shows all the two-component systems which form the protein complexes and are identified in this database.

Table 5.1: Protein complexes found on the EcoCyc database

Complex	
Alkanesulfonate monooxygenase	ZraSR
Aerotactic Signal Transduction System	YpdAB
ArcAB	YehUT
AtoSC	UhpBA
BaeSR	TorSR
BarA UvrY	RcsCDB
BasSR	PhoRB
Chemotactic Signal Transduction	QseBC
CpxAR	PhoQP
CreCB	NtrBC
CusSR	Nitrogen regulation
DcuSR	NarX
DpiBA	NarQ
EnvZ	KdpDE
EvgSA	GlrKR

In order to provide an evaluation of the developed approach, a search was performed on the entire *E. coli* network for the sub-networks that correspond to these protein groups. It is important to mention that the dictionary created in this approach does not consider environmental conditions or external factors inherent to the organism.

Cytoscape was used to accomplish this task by searching for nodes that correspond to these complexes. It is capable of identifying a node by name and also through the use of synonyms. After the node has been selected, it is possible to find its neighbors and then create a new sub-network, thus expecting to find a relationship between the proteins that are present in the protein complex.

Using the 31 protein complexes identified in the EcoCyc database, 15 of these complexes were validated by the developed approach. These networks agree with the information found on the database, as can seen in Figures: 5.1, 5.2, 5.3, 5.4, 5.5, 5.6, 5.7, 5.8, 5.9, 5.10, 5.11, 5.12, 5.13, 5.14 and 5.15.

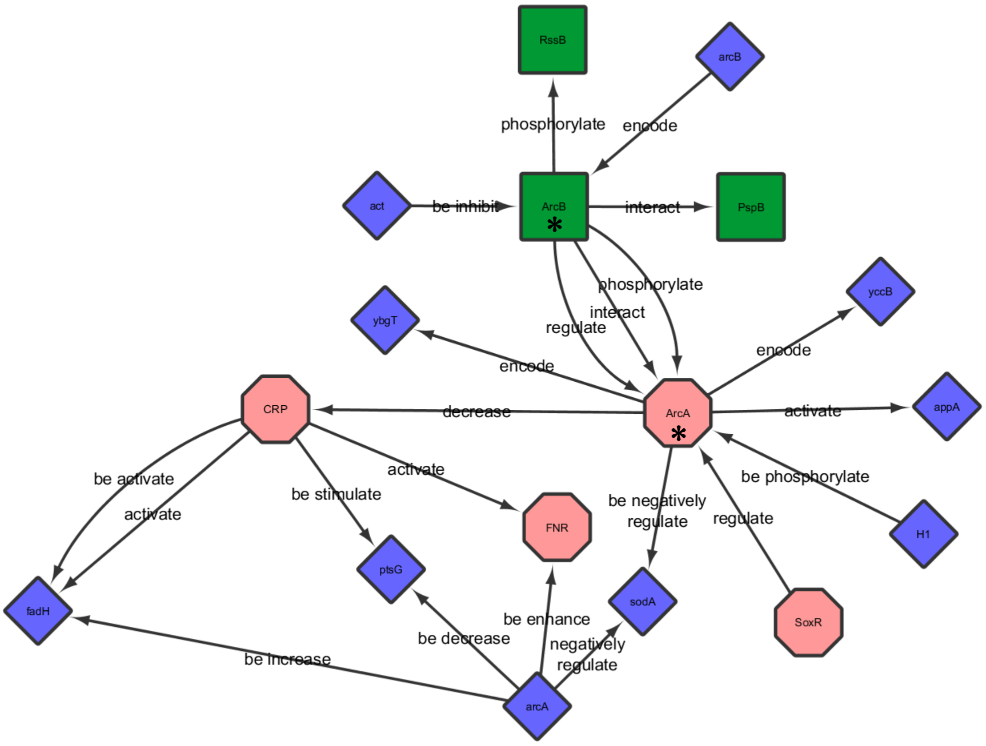


Figure 5.1: The ArcAB complex is able for respond to changing respiratory conditions of growth [125].

The results shown in this section, serve as a proof of concept concerning the developed work. Even though no external factors are identified in the dictionary, it was possible to find several regulatory events about of the protein complexes identified by the EcoCyc database.

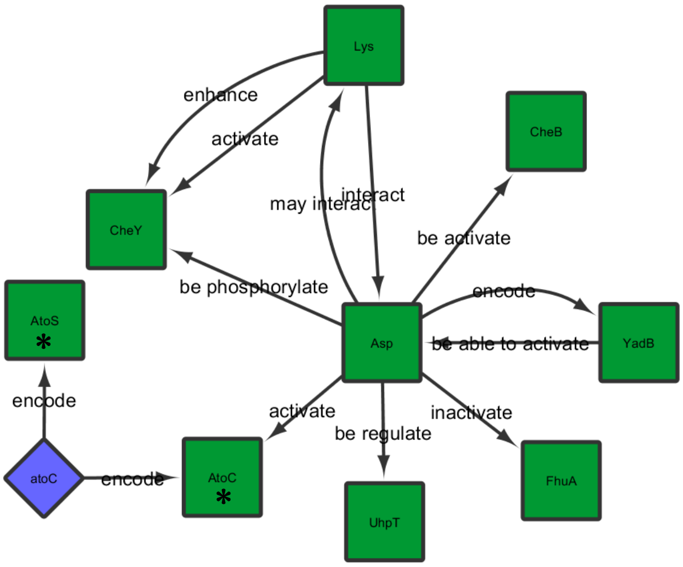


Figure 5.2: The AtoSC complex provides a regulator response for modulating the activity of AtoC [126].

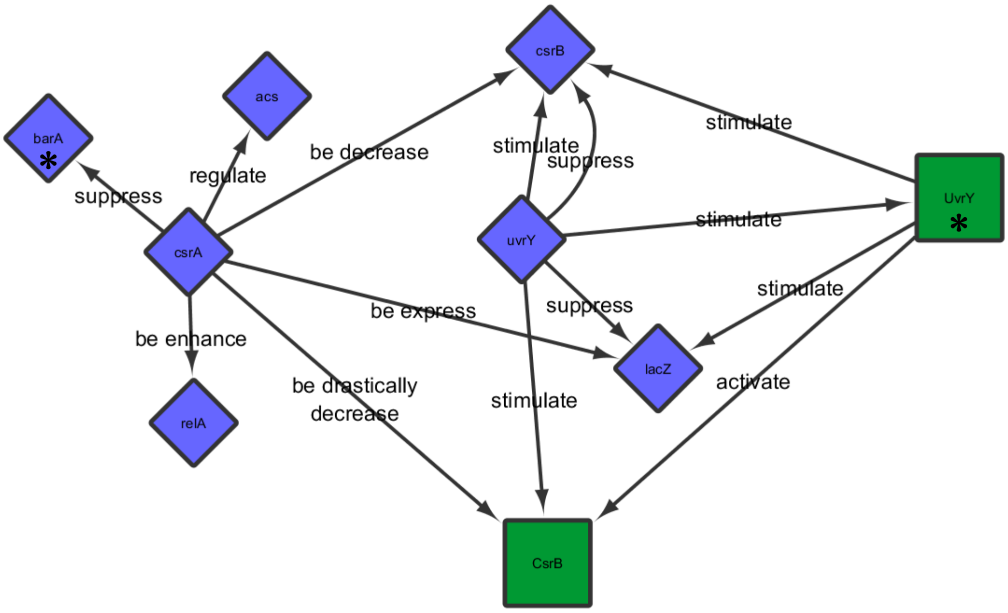


Figure 5.3: The BarA/UvrY complex comprises the BarA and UvrY proteins to play a role in the central carbon metabolism[127]

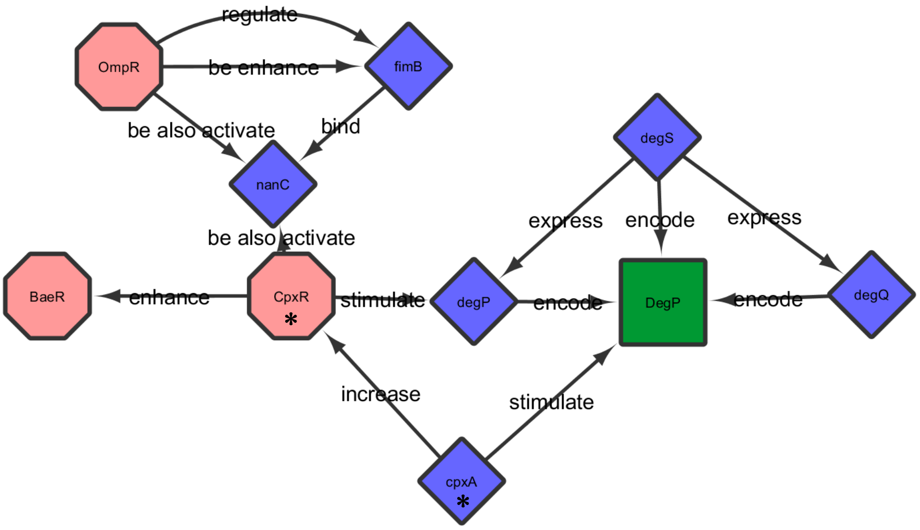


Figure 5.4: CpxAR is a stress complex in response to cell envelope damage [128].

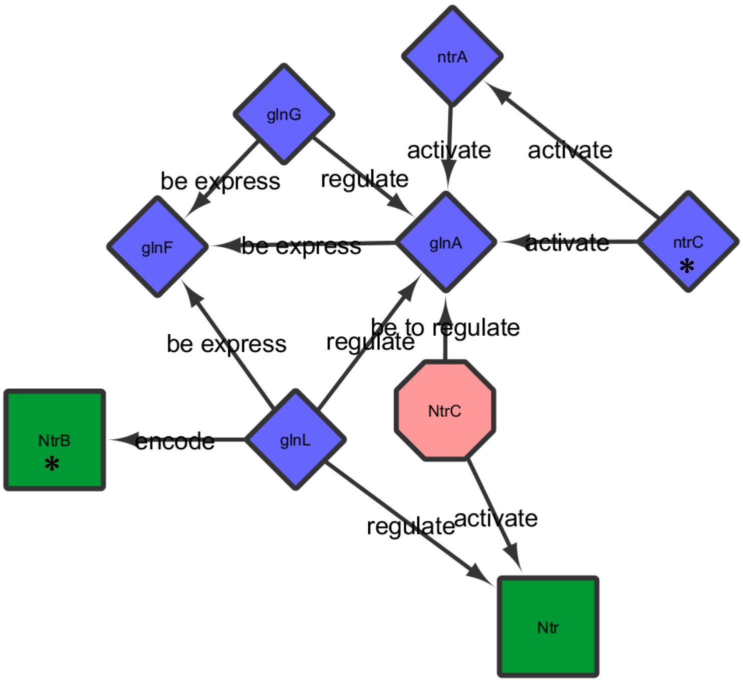


Figure 5.5: The NtrBC complex controls the transcription process of the Ntr regulon [129].

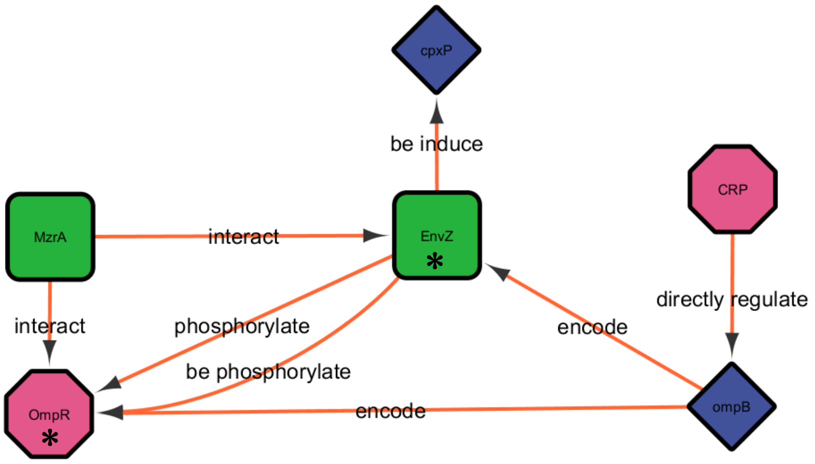


Figure 5.6: EnvZ is a signal transduction system involved in the regulation of over 100 genes concerning to response in the osmotic milieu of the cell [130].

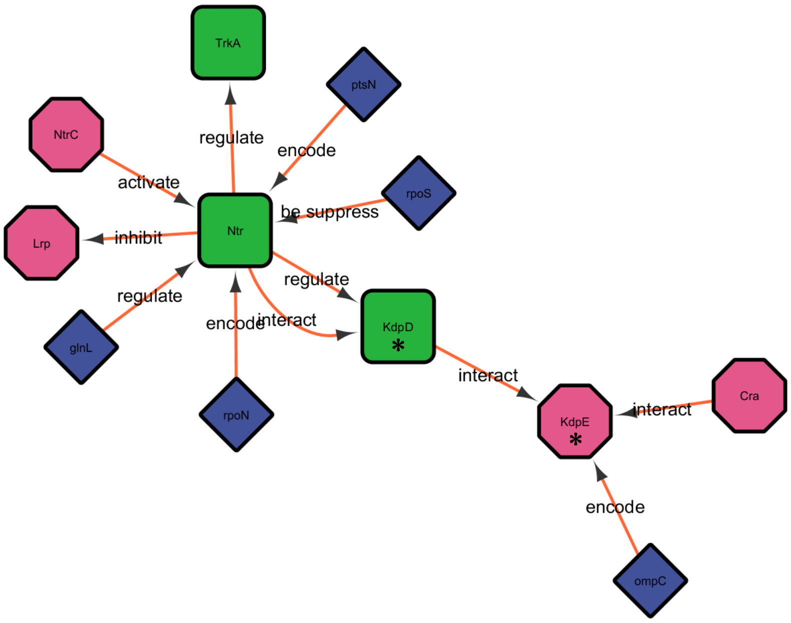


Figure 5.7: The KdpDE complex activates the expression of the kdpFABC operon in response to a limitation or salt stress [131].

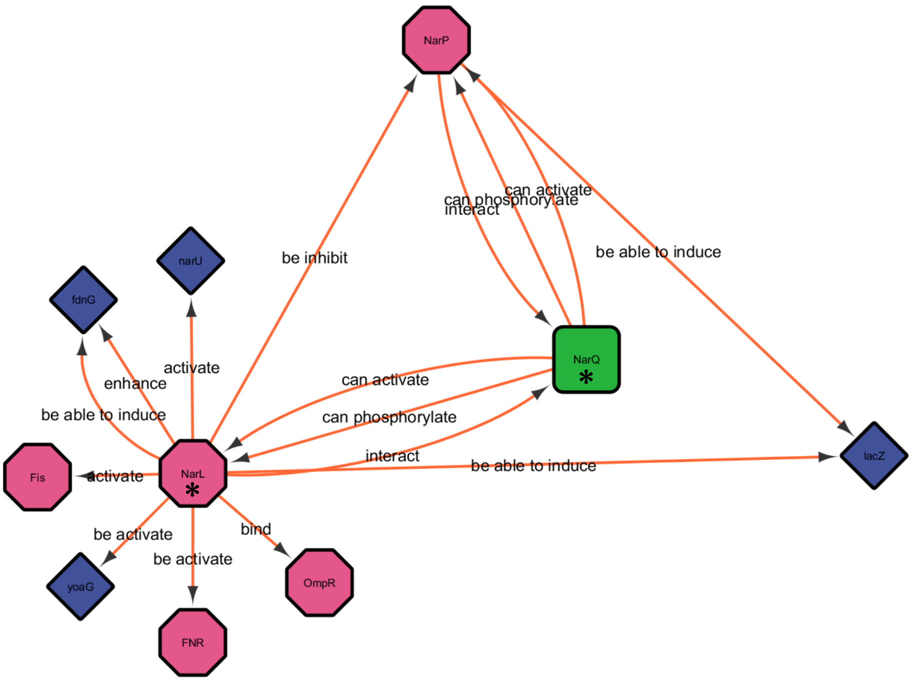


Figure 5.8: The NarQ complex provides a support to control anaerobic respiratory gene expression in response to nitrate and nitrite [132].

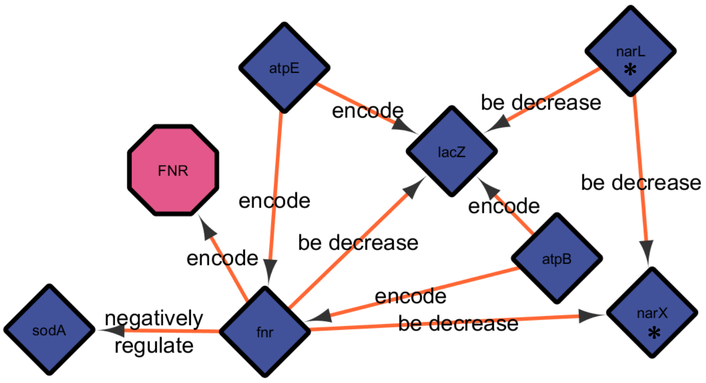


Figure 5.9: The NarX complex, and also the NarQ system, collaborates to control anaerobic respiratory gene expression in response to the viability of nitrate and nitrite [133].

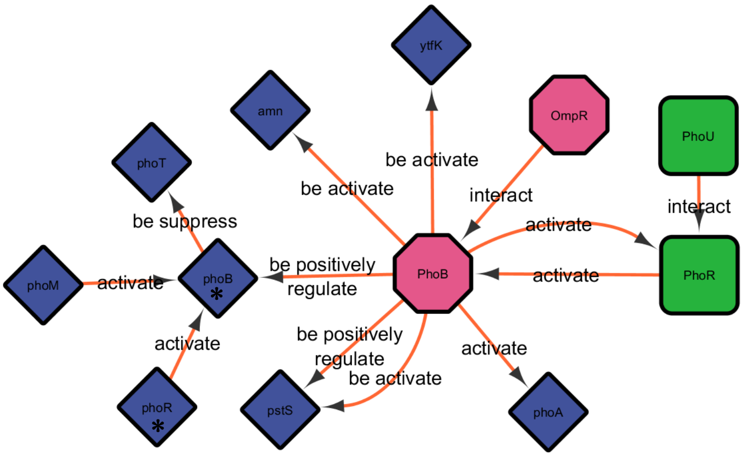


Figure 5.10: The PhoBR complex controls the activity of the cytoplasmic response regulator and also to the transcription activity of factor PhoB [134].

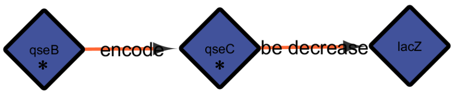


Figure 5.11: The QseBC complex might be considered as the system that transcriptionally regulates the expression of flagella [135].

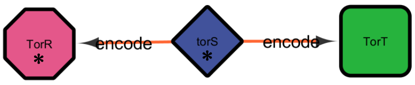


Figure 5.12: The TorSR complex is responsible for regulating genes involved in the acid-stress defense [136].

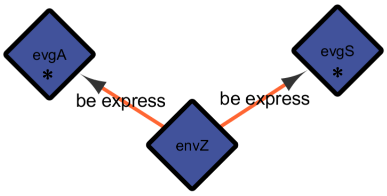


Figure 5.13: EvgSA is a complex that provides acid resistance to *E. coli* cells [137].

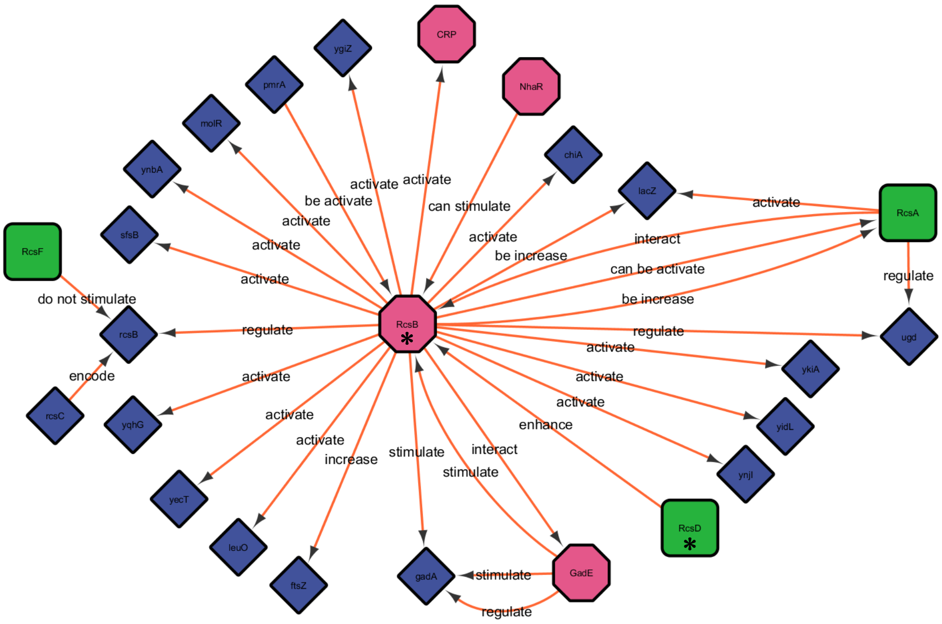


Figure 5.14: The RcsCDB complex is responsible for activating a wide range of genes, providing a cellular stress response [138].

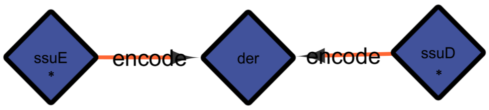


Figure 5.15: The Alkanesulfonate monooxygenase complex works as a sulfur source [139].

5.2 Comparing gene targets between the models

The model published by Covert [120] in the year of 2004 is still used as a gold-standard when dealing with reconstructions or simulation processes of the *E. coli* organism. This model has information about 1010 genes, where 104 are regulatory genes and was built mainly using information from scientific literature and databases. This model is available from the supplementary material section provided by the Nature journal[1]. In order attempt to validate the knowledge discovered using Biomedical Text Mining, the approach used was compared with Covert's model.

In order to accomplish this task, a method was developed to search for the target genes present in Covert's model and the ones found using the proposed Biomedical Text Mining approach.

The proposed analysis was to identify which genes appeared on both models and which gene only appear in one of the approaches (i.e., only on the Biomedical Text Mining model but not on Covert's model or vice-versa). The results are shown in Figure 5.16.

These results indicate that a large numbers of genes found using the approach proposed in this work could help discover new knowledge information about the regulatory mechanism of organisms. In order to verify the importance of the genes discovered in this work for *E. coli*, they were classified using the Gene Ontology (GO)[2] enrichment analysis. GO is a project whose goal is to provide a consistent description of genes and their products across several databases, helping researchers agree on the use and meaning of terms [140].

The goal is to classify each gene according to their biological function. To accomplish this step, a list of gene identifiers is loaded onto the GO enrichment analysis tool. This task is implemented by the PANTHER[3] (Protein ANnotation THrough Evolutionary Relationship) system where the genes are

[1] http://www.nature.com/nature/journal/v429/n6987/suppinfo/nature02456.html

[2] http://geneontology.org/page/go-enrichment-analysis

[3] http://www.pantherdb.org/

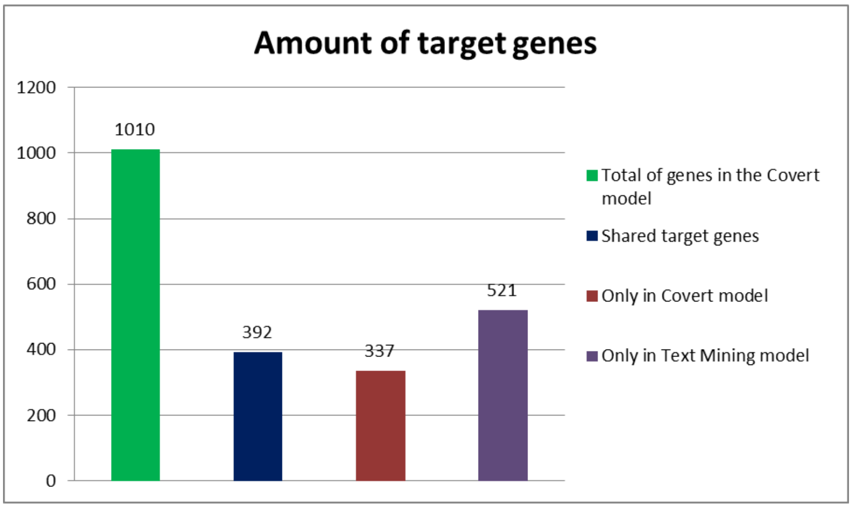


Figure 5.16: Comparison of targets genes retrieved from the Covert model with target genes discovery from the Biomedical Text Mining processing.

classified according to their function in different classes: families and subfamilies which are annotated using ontology terms provided by GO [141]. For each gene, a phylogenetic tree is built for representing the evolutionary relationships between all the genes in their family [142].

The outcome of this step is a list of GO classification that determines a biological process associated to each gene and also a *p-value*, whose value is can be interpreter as the probability of a given gene being given the same GO classification by chance [143]. This *p-value* helps understand if the genes found to be related to a given biological process are deemed interesting (if they have a low *p-value*) or if the findings are somewhat irrelevant. The *P-value* cut-off used in PANTHER approach is 0.05 [144].

521 target genes discovered by the Text Mining processing were submitted to this tool for further analysis and the outcome obtained was that 433 genes were related to 134 biological processes, while the 88 remaining genes could not be mapped with any GO identifier. These results were exported to a text file and that was subsequently uploaded to a tool called REVIGO[4] for providing a better visualization of this outcome.

[4] http://revigo.irb.hr/

The goal of the REVIGO Web Server is to summarize long lists of GO terms by searching a representative subset of terms, thus reducing possible functional redundancies belonging to the GO list. It is able to visualize the GO terms in plots, interactive graphs, tree maps or tag clouds [145]. Figure 5.17 shows the results of this analysis, where it is possible to visualize a two-level hierarchy of GO terms, the first corresponding to the key words which are overrepresented in the GO terms list provided by the PANTHER tool; the second level is represented by the keywords which are correlated to the *p-value* also supplied by the PANTHER system.

Based on this analysis, it is possible to conclude that several genes uncovered by the Biomedical Text Mining study, and that were classified by the GO reference list, play important roles in the *E. coli* organism such as: positive regulation of gene expression and heterocycle metabolism, as illustrated in Figure 5.17. This seems to indicate that new approaches like this one are necessary to improve the existing regulatory model of the *E. coli*.

5.3 Improving the regulatory model for *in silico* strain optimization

Nowadays, efforts in Metabolic Engineering (ME) and Systems Biology provide the development of genome-scale metabolic models for several organisms. These models may be used to predict the cellular behaviour, their metabolism and also to find genetic modifications that increase the productivity of desired compounds [146]. These models allow the simulation of the microbe's phenotype under different environmental conditions (e.g, aerobic/anaerobic, nutrients) and to predict the phenotypes of mutant strains (e.g, gene knockouts).

The phenotype can be characterized by the observable characteristics of an organism such as their biochemical properties, behaviour and morphology [147]. It is determined by the genotype, the set of genes present in the organism, and also the environmental impacts that can affect these genes

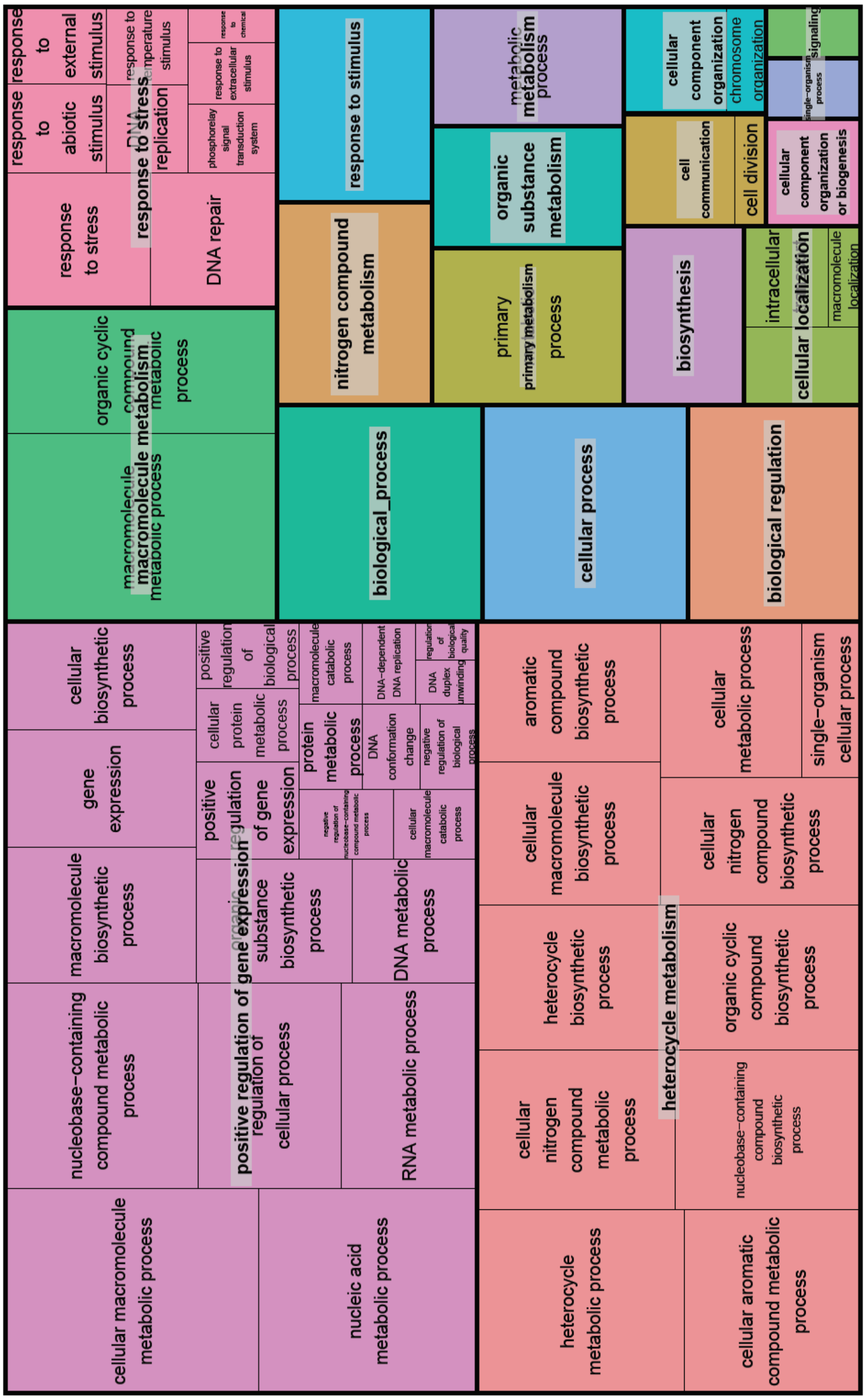


Figure 5.17: Treemap representation of GO terms list generated from PANTHER tool with a list of GO terms retrieved by genes discovered in the Text Mining process.

[148].

In order to improve phenotype predictions, several constraint-based models were developed relying only on information about the metabolic capacities of an organism, namely Flux Balance Analysis (FBA) [149], ROOM [150] and MOMA [151].

The combination of reliable models with simulation methods is the basis for strain optimization algorithms, where the goal is to find a set of genetic modifications to apply to a specific strain in order to produce a desirable compound, generally related with the industrial production of a metabolite.

The outcome of these methods and the need felt in this field for appropriate computational tools fostered the development of the OptFlux[5], a software system that integrates a number of useful features to support ME [152]. Since the year of 2011, several methods for the integration of regulatory information in phenotype simulation and strain optimization have been added [153]. Thus, the methods and tools for integrating metabolic and regulatory models are concentrated all in the same software system, Figure 5.18 shows a hypothetical example of an integrated model.

Recently, a new plug-in developed for integrating the OptFlux system was designed, called Reg4OptFlux [154] that aims to design *in silico* strain optimization, by performing the tasks of metabolic/regulatory model reconstruction, phenotype simulation and strain optimization able to deal with different types of models. This approach was chosen to provide some possible results to this work because it was developed in the same research group and also because this tool is open-source and quite user-friendly.

However, there is a lack of suitable regulatory models that can be used to support this effective use. Moreover, it is not possible to find a large number of models that can be used to support ME and there is a pressing need for adequate methods and tools for genome-scale regulatory reconstruction.

In order to measure the feasibility of the pipeline that was developed in this work by finding evidence that the regulatory network outcome designed in this approach can help improve existing models, a simulation and also a strain optimization, using the OptFlux and Reg4OptFlux plug-in, were per-

[5] http://optflux.org/

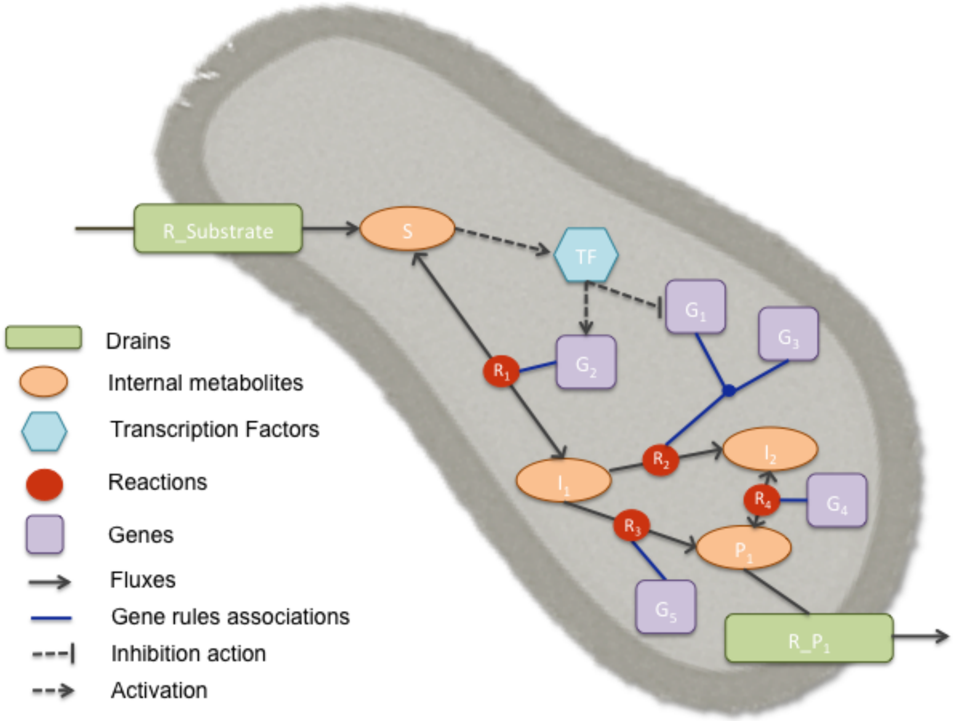


Figure 5.18: This model — whose objective is to produce P_1 — includes a Transcription factor (TF) that is activated when the metabolite (S) is present, and that activates the gene (G_2) that catalyzes the reaction (R_1) converting the internal metabolite (S) into (I_1) mutually because this reaction is reversible; the TF also inhibits gene (G_1) that together with gene (G_3) catalyze the reaction (R_2) that converts (I_1) into (I_2); gene (G_5) catalyzes the reaction (R_3) that convertis (I_1) into (P_1); gene (G_4) is able to catalyze a reversible reaction (R_4) that converts (I_2) into (P_1) and vice-versa.

formed. These used both the regulatory and metabolic networks published by Orth [2] and available on the site of Systems Biology Research Group[6] from the University of California. This model is considered a core model for *E. coli*, because it represents a sub-set of genome-scale metabolic reconstruction, consisting of a total of 95 reactions.

Firstly, the metabolic model is loaded on OptFlux, using a file encoded using the Systems Biology Markup Language[7] (SBML) protocol whose structure is composed by a list of unit definitions (e.g, mole, gram, second), a list

[6] http://systemsbiology.ucsd.edu/Downloads/EcoliCore

[7] http://sbml.org/

of cellular compartments (e.g, cytoplasm, extracellular), a list of species that is defined by the amount of metabolites present in the model (in this case there are 92 metabolites) and finally the list of 95 reactions, described by their identifier, name, reversibility, the genes that are associated with each reaction, the list of reactants, the list of products and their kinetic parameters.

During this step, a Comma-Separated Values (CSV) file is loaded onto OptFlux, containing the gene rules, where the first column is associated to a gene identifier, the second column specifies the gene name, the third column refers to the Transcription Factor (TF) name, the fourth column gives the Boolean rules. Each line of this file represents only one gene and the rule related with this gene as can seen in Table 5.2.

Representation of CSV file content			
B-number	Gene name	TF	Rule
b0114	aceE	AceE	NOT PdhR OR Fis
b3952	pflC	PflC	ArcA OR Fnr
b1611	fumC	FumC	NOT ArcA
...	...	...	...

Table 5.2: An example for illustrating the composition of the CSV file of a regulatory model.

After that, it is necessary to choose the environmental conditions described in the SBML file and also to select the reaction related to the biomass growth as an objective function for running the simulation [155]. The result of this step is shown in Figure 5.19, where it is possible to see the output of the simulation process. The view is composed by the clipboard on the left side that comprises a list of all the processes performed in OptFlux, in the middle is the visualization of the genes' status (ON/OFF), that indicate whether they are active or inhibited in this simulation and on the right side are the conditions specified by the user.

With the aim of providing a comparison between the existing model and the model improved by this work, several phenotype simulations using different environmental conditions were performed. 16 out of the 74 regulatory

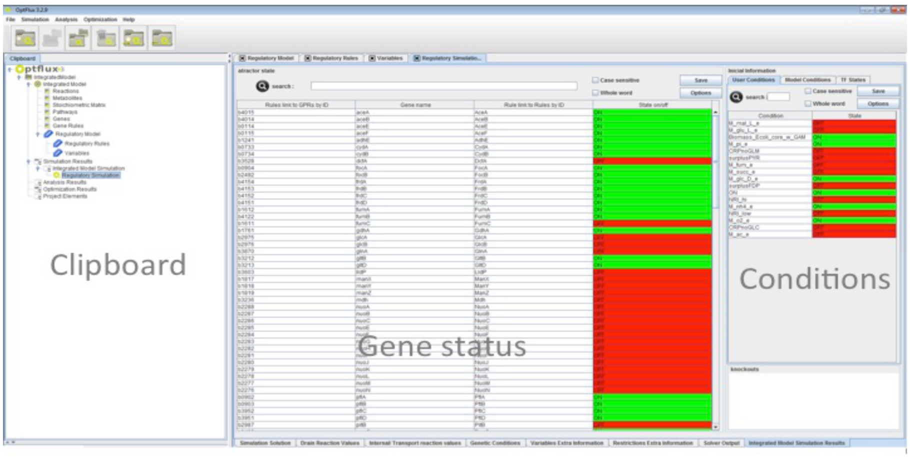


Figure 5.19: OptFlux graphical interface

rules present in the model published by Orth were improved. Table 5.3 shows the regulatory rules that were modified by the addition of Transcription Factors (in red) that were retrieved from the scientific literature by using the Biomedical Text Mining approach developed in this work.

Initially, it is necessary to define some basic conditions to execute a simulation like the biomass reaction that is defined in the metabolic model and also the metabolites that are essential for the organism's growth. In this case, the environmental conditions were already defined in the metabolic model. All that was needed was to choose which were inserted in the minimal medium: glucose (glc-d[e]), oxygen (o2[e]), phosphate (pi[e]) and ammonium (nh4[e]) based on [120].

Two integrated model simulations were obtained and the outcome is very interesting because it shows that the model that was modified by the approach proposed in this work is feasible since the value of biomass produced remained the same (0.87391038) in comparison with the original model. The results can seen in Table 5.4.

The next step to be analysed concerns the strain optimization, whose goal is to produce genetic modifications leading to a mutant strain allowing to overproduce a given chemical compound with an industrial interest [156]. The field of producing biofuels and bioproducts has increased in the last years in order to supply renewable energy while ensuring environmental health

B-number	Gene	Rule
b4015	aceA	(NOT IclR) AND (NOT ArcA) OR (FruR OR AceK)
b4014	aceB	(NOT IclR) AND (NOT ArcA) OR (FruR OR AceK OR IHF)
b1241	adhE	(NOT o2[e]) OR (NOT (o2[e] AND FurR)) OR Fis OR Cra
b3528	dctA	CRPnoGLM AND (NOT ArcA) AND DcuR OR Crp
b1612	fumA	(NOT(ArcA OR Fnr)) OR ManA
b1611	fumC	(MarA OR SoxS OR Rob) AND NOT(ArcA)
b3870	glnA	Crp OR NtrC or GlnG OR NtrA OR GlnL
b3236	mdh	Not(ArcA) OR LdhA
b1101	ptsG	NOT(Mlc OR FruR) OR SoxS OR Crp
b0721	sdhC	(NOT(ArcA OR Fnr)) OR Crp OR Fis OR Fur
b0722	sdhD	(NOT(ArcA OR Fnr)) OR Crp OR Fis OR SdhC
b4124	dcuR	Dcus OR CitA OR DpiB
b1187	fadR	glc-D[e] OR (NOT ac[e]) OR (NanR OR FadM)
b1594	mlC	NOT glc-D[e] OR (PstG OR Arg OR MtfA)
b0399	phoB	PhoR OR OmpR
b0400	phoR	NOT pi[e] OR (PhoB OR PhoU)

Table 5.3: The rules are composed by Transcription Factors and also environmental conditions needed to growth the cell, such as: o2[e] refers to oxygen, glc-D[e] represents the glucose, ac[e] is associated to acetate and pi[e] to phosphate, these metabolites are present in the minimum media.

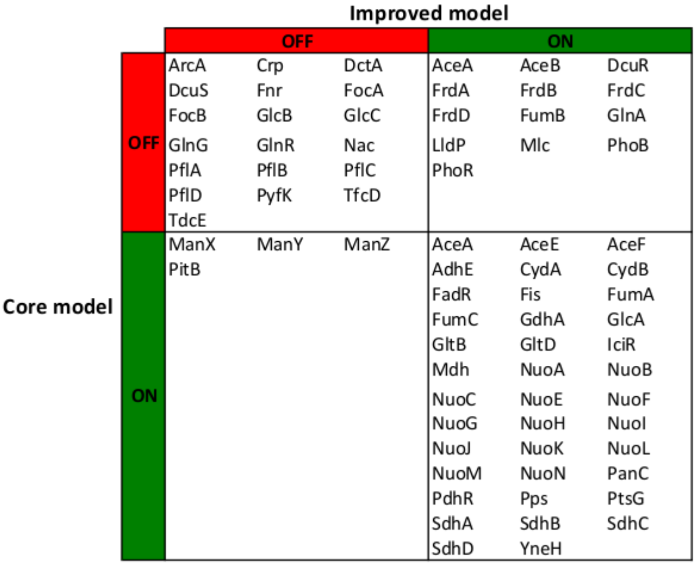
Original model				Improved model			
Consumption		Production		Consumption		Production	
Ammonium	4.76526	CO$_2$	22.81031	Ammonium	4.76526	CO$_2$	22.81031
Phosphate	3.22359	H$_2$O	29.17622	Phosphate	3.22359	H$_2$O	29.17622
Oxygen	21.7998	H	17.53064	Oxygen	21.7998	H	17.53064
Glucose	10.0			Glucose	10.0		

Table 5.4: This result means that to obtain a growth rate the cells need consume: ammonium, phosphate, oxygen and glucose thus producing: carbon dioxide, water and hydrogen.

[157].

The first step is to compare both regulatory simulations, which means reproducing the gene status according to the regulatory rules. In order to perform an optimization process by using the improved regulatory model, the scenario chosen was to produce succinate in aerobic conditions for comparing both models. Succinate was chosen since it has aroused the interest as a precursor of several chemicals used in the industry [158].

Initially, a regulatory simulation was performed in order to compare the gene status in both models. Both models were simulated under the same environmental conditions and the carbon source used to produce biomass was glucose. Figure 5.20, shows all genes present in the regulatory rules and for each gene their status (ON = activated or OFF = inhibited) in both models.



		Improved model	
		OFF	ON
Core model	OFF	ArcA Crp DctA DcuS Fnr FocA FocB GlcB GlcC GlnG GlnR Nac PflA PflB PflC PflD PyfK TfcD TdcE	AceA AceB DcuR FrdA FrdB FrdC FrdD FumB GlnA LldP Mlc PhoB PhoR
	ON	ManX ManY ManZ PitB	AceA AceE AceF AdhE CydA CydB FadR Fis FumA FumC GdhA GlcA GltB GltD IciR Mdh NuoA NuoB NuoC NuoE NuoF NuoG NuoH NuoI NuoJ NuoK NuoL NuoM NuoN PanC PdhR Pps PtsG SdhA SdhB SdhC SdhD YneH

Figure 5.20: A comparison between gene status from the original model and the improved one

These differences can potentially represent significantly changes in some metabolic fluxes, causing genetic perturbations on the metabolism [150] and also lead to under/overproduction of chemical compounds. Since the improved model has 19 new Transcription Factors that were not present in the original model it is possible to add these new TFs to the environmental conditions in OptFlux.

Therefore, it is necessary to perform an analysis to ascertain whether these new TFs that were added to the model may cause some initial perturbation in their phenotype stability. Thus, a comparison is proposed under the same conditions for both models, in this case, the scenario chosen was to use glucose as a carbon source, in an aerobic system, also containing ammonium and phosphate [159].

The results obtained by this comparison are shown in Figure 5.21, where it is possible to conclude that although new TFs have been added they are not causing any changes in the original phenotype.

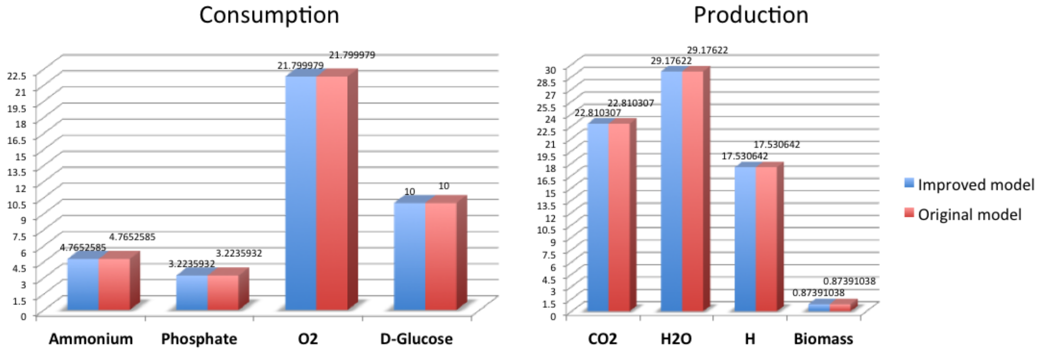


Figure 5.21: Comparison of simulation output from both models

Based on the work published by Lee [160] and their colleagues, where they proposed an overproduction of succinate in *E. coli* under aerobic conditions and using glucose as a carbon source to the biomass growth, it was decided to compare both regulatory models under the same conditions, indeed only the mutant strains because the wild type of *E. coli* is not able to produce succinate in this environmental conditions [152]. Lee's work suggested to perform some gene knockouts (*ptsG*, *pykF*, *aceBA*, *sdhA* and *mqo*) in order to improve the production of succinate and also to maintain the biomass growth. However, it is necessary to take into account that the metabolic model used in this analysis is a core model published by Orth [2] and the model used by Lee refers to the *E. coli W3110* [161], that is more complex than the core one and that has both more genes and reactions. After verifying the main reactions involved in the production of succinate from the core model based on Lee's work, it was decided to chose which genes could be deleted in order to optimize the pathway from glucose until the succinate excretion. Thus the set of genes deleted was: *ldhA*, *dld*, *pta*, *eutD*, *adhE*, *mhpF*, *sdhC*.

After performing an FBA analysis using the same gene knockouts that were identified above, the results obtained diverge between both models. The results are summarized in Figure 5.22, where it is possible to see that the improved model was able to produce succinate while this was not observed in the original model. Even as a lower value than expected, it shows that by performing the needed improvement on the regulatory rules, it is possible to enhance the production of a given chemical compound by using the present approach. This is due to the improvement performed on the regulatory rules. Table 5.5 shows the comparison between the original model and the improved one, concerning the reactions which were activated due to gene knockouts. It suggests that an alternative pathway able to produce the succinate compound that was not found in the original model was discovered.

Reaction ID	Reaction name	Original model flux value	Improved model flux value
R_ICL	Isocitrate lyase	0.0	0.00023935473
R_MALS	Malate synthase	0.0	0.00023935473
R_MDH	Malate dehydrogenase	0.0	0.00023935473
R_SUCCt3	Succinate transport out	0.0	0.00023935473

Table 5.5: This table summarizes the reactions which were affected by the gene deletions performed on simulation and also present the flux value for each one.

In order to demonstrate the reactions which allow the improved model to produce succinate, Figure 5.23 highlights the pathway found by those reactions mentioned above from the carbon source (glucose) until the excretion, into the external environment, of the succinate.

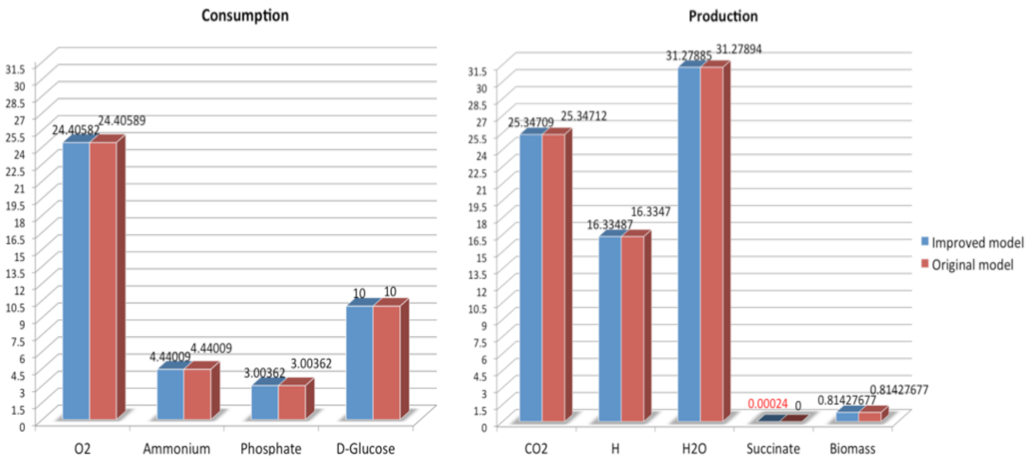


Figure 5.22: Simulation output

The results obtained in this chapter are quite promising and suggest that the automated knowledge discovery from the literature are a good way of identifying regulatory events.

Although the present study uses an evaluation with a limited set of simulations and optimizations, the outcome obtained brings a valuable comparative insight between the original model published by Orth [2] and the model improved by using the approach proposed in this work. The present contribution is therefore a progress in providing a better solution for discovering new knowledge concerning Transcriptional Regulatory Networks using data integration and information retrieved from scientific literature. The performed analysis also indicates that, for some simulations methods, the improved model yields better results.

5.4 Summary

In this chapter, a proof of concept was described by performing several comparisons of the results obtained using the current work with existing approaches. The first analysis task regarding the protein complexes present in *E. coli* organism, identified 17 sub-networks in a total of 31 retrieved from the EcoCyc database, even though this approach was not able to recognize

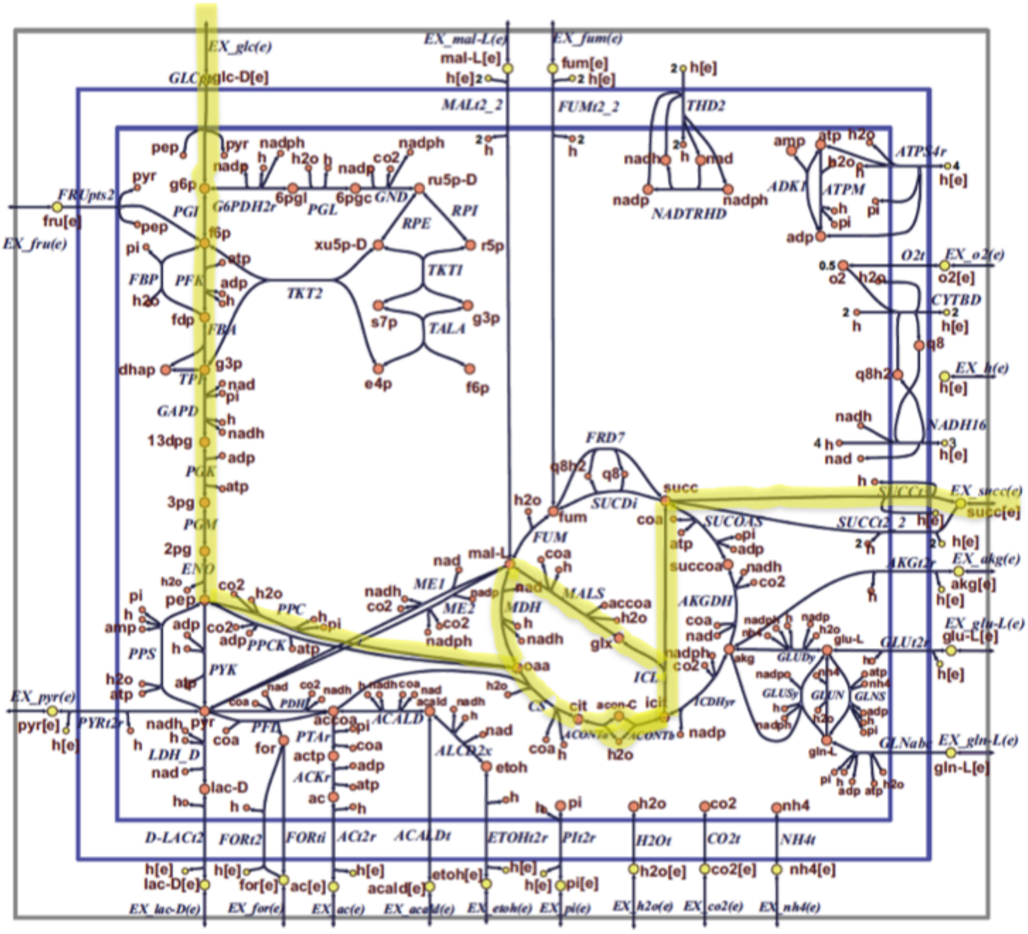


Figure 5.23: In this figure, adapted from [2], is shown the core fluxes present in the core model. The first rectangle in gray color, represents the frontier between the model and environmental source of substrates. The inner rectangles in blue are used to identify the inside and outer surface of cytoplasmic membrane. Metabolites present on cytosolic space are represented by orange circles and extracellular metabolites the yellow ones

external factors that are associated to these complexes. Furthermore, taking into account the work published by Covert [120], a comparison was established between the existent model and the regulatory network inferred in the currently work, using the same target organism (*E. coli*). The outcome of this task enabled the discovery of a set of genes which were not present in the model published by Covert. A gene classification task using the PAN-

THER and REVIGO tools seemed to indicate that several genes might play an important role in the biological processes of *E. coli.* The last analysis performed in this chapter aimed to improve an existing regulatory model from *E. coli* published by [2] and performed several simulations and optimizations using the OptFlux software system under different test scenarios and it was possible to conclude that the approach proposed in this work had a similar performance when compared both models and for some tests the improved model has shown better results.

Chapter 6

Conclusions

This chapter provides the main conclusions of the present work as well as the achieved objectives, the main contribution of this thesis and finally some considerations about the future work.

6.1 Summary

Nowadays, Systems Biology is a field that is attracting much interest due to the interest in the process of biological simulation, whose aim is to perform a reconstruction, *in silico* and *in vivo*, of all processes that occur inside the cells, both metabolic or regulatory. In this field, Transcriptional Regulatory Networks (TRNs) are powerful tools for representing interactions between biological entities within a cell and their study helps understand the process of regulatory interactions that link the Transcription Factors (TFs) to their target genes.

In the case of TRNs, it is important to retrieve different types of information about genes, proteins, TFs and scientific literature. The vast amount and complexity of biological data retrieved in recent years requires an integrated approach.

There is currently a vast wealth of information in biomedical literature to help researchers understand regulatory interactions and build TRN models. However, due to the rapidly increasing number of scientific papers concerning this subject, the task of gathering all the necessary information available is daunting.

This context has motivated the present research work, whose main objective is to provide an approach for discovering new knowledge using information from different data sources and scientific literature in order to build regulatory models.

This has led to the development of an integrated biological repository gathered from several databases that is able to provide all information needed to build TRNs. Afterwards, Text Mining techniques were applied in order to extract specific regulatory events from the literature, an exporting process was designed to allow the graphical visualization of these networks and finally improvement was made to an existing model using the information uncovered during this work.

6.2 Objectives and outcome

This section will review the objectives that were established in the beginning of this work and will highlight their outcomes.

- **A proposal for building an integrated repository**

  After the literature review, it was apparent that there was a vast amount of information concerning regulatory networks spread through several databases and that integrating data from different sources is still a very difficult task to perform within the bioinformatics domain. This is due to the heterogeneity of the biological databases, their identifiers, structure, synonyms and standards. This context motivated the first goal achieved in this research work, that was to develop an integrated biological repository. This repository, was called KREN [4], using information extracted from 4 main databases: KEGG, RegulonDB, EcoCyc and NCBI. Several methods were implemented in order to facilitate the task of gathering information through the use of Web Services and after that to store it in a structured document created in the eXtensible Markup Language[1] (XML). This approach addressed the process of biological data integration in the field of TRNs, bringing an alternative approach to retrieve, integrate and store information from some the databases chosen doe this study. It can be easily extended in order to incorporate data from other organisms that are available in the databases described previously.

- **Retrieving information from KREN and applying Text Mining Techniques** Since there is a lot of new information being published in articles, it makes sense to incorporate it into the repository. Most of the information available concerning regulatory events is located in the scientific literature. Since the number of papers is vast, the importance of using Biomedical Text Mining fostered a new goal of this work.

  Thus, after integrating the information found on the databases into the KREN repository, a new Text Mining pipeline was created with the purpose of building TRNs. In order to accomplish this task, several methods were implemented in order to load the information retrieved from KERN and load it onto the @Note[2] software system.

  Using that information, the @Note software system was used for retrieving the relevant documents from the PubMed database. A novel pipeline was designed using @Note for building TRNs, comprising the main Text Mining methods such as Information Retrieval and Information Extraction.

  The first outcome of this goal is the information concerning the regulatory events found in the documents and is represented as a list of sources nodes, target nodes and edges that represent the regulatory interaction (activated/inhibited) between these nodes. Following this step, an export process was implemented by creating a structured graph file that can be visualized using the Cytoscape[3] framework.

- **Proof of concept** In order to validate this work, several tasks were

[1] https://www.w3.org/XML/

[2] http://www.anote-project.org

[3] http://cytoscape.org

defined to evaluate the discovered knowledge. Firstly, a search for several specific protein complexes present in the *E. coli*[120] model was performed. A list of protein complexes (containing two proteins) was retrieved from the EcoCyc database the network created during the previous step was searched in order to validate them. To this end, a search was performed using Cytoscape for the neighbors of one of the proteins in the complex in order to verify that the other protein was among its neighbors. The outcome of this approach was very successful since it was able to identify more than half of the protein complexes described in the EcoCyc.

The next task performs a comparison between the network that was created during this work and the regulatory model published by Covert [120]. The purpose was to identify, using the classification defined on Gene Ontology[4], possible target genes that might be related with regulatory events and are not present in the published model. The final evaluation was of the current work was to perform simulations using an integrated model (metabolic and regulatory) published by Orth [2] in order to identify significant changes regarding the regulatory model improved by the present work. Finally, several simulations were performed using different phenotype conditions and gene knockouts in order to produce an overproduction of a specific chemical compound. It was possible to conclude that the approach developed during this research work, can bring new valuable knowledge in the field of regulatory networks.

6.3 Future Work

During this work, several topics have been deemed interesting and were identified as opportunities for extending this research.

- To extend the KREN repository to other tasks.The approach developed in this work only uses a part of the repository; other types of informa-

[4] http://geneontology.org

tion collected during this work such as the annotated sequences of each protein could be used to help implement algorithms for searching for motifs in order to predict the protein function.

- To analyse the regulatory networks among others organisms. As the development of the present research work shows a novel way of discovering knowledge about regulatory networks and with the large amount of scientific papers published steadily increasing in the last years, it is possible to consider the improvement of regulatory models in other organisms like the *Bacillus subtilis*, *Helicobacter Pylori* and *Mycobacterium tuberculosis* or even on large genome scale regulatory models.
- The development of a plug-in for allowing users to visualize the networks within @Note. This would gain independence from Cytoscape. Actually this task is already being developed, since the mechanism used for exporting regulatory networks implemented in the present research uses a Java library called GraphStream [5]. This library allows users to create, visualize, model and analyse dynamic graphs. This approach considers a Graph ($G = V,E$) composed by a set of V (vertices) and E (edges), as well as defined relationship between biological entities in this work. Moreover, GraphStream provides several tools for representing graphs, generating, importing/exporting and also algorithms concerning to graph theory [162]. It might be a nice approach to implement a visualization plug-in on @Note software system.

6.4 Final Considerations

The present work was mainly focused in proposing a solution for integrating biological data related to Transcriptional Regulatory Networks and to extend an already developed software system (@Note) with methods which allow the creation of these networks for any type of prokaryotic organism. Moreover, it suggests an approach that can be used as a reference to improve models and

[5] http://graphstream-project.org

use it to perform *in silico* strain optimization. Although there is a lack of tools for building these networks, the reviewed literature and the outcomes of this work clearly show that this is a promising area of study.

Bibliography

[1] Simon I, Barnett J, Hannett N, Harbison CT, Rinaldi NJ, Volkert TL, Wyrick JJ, Zeitlinger J, Gifford DK, Jaakkola TS, and Young Ra. *Serial regulation of transcriptional regulators in the yeast cell cycle.* Cell, 106(6), 697, September 2001. ISSN 0092-8674.

[2] Orth JD, Fleming RM, and Palsson BØ. *Reconstruction and use of microbial metabolic networks: the core Escherichia coli metabolic model as an educational guide.* EcoSal Plus, 4(1), 2010.

[3] Pereira R and Mendes R. *Current Trends in Bio-Ontologies and Data Integration.* In *Distributed Computing and Artificial Intelligence*, volume 217 of *Advances in Intelligent Systems and Computing*, pages 579–586. Springer International Publishing, 2013. ISBN 978-3-319-00550-8.

[4] Pereira R and Mendes R. *Integrating Biological Databases in the Context of Transcriptional Regulatory Networks.* International Journal of Bioscience, Biochemistry and Bioinformatics, 4, 345, 2014. ISSN 20103638.

[5] Pereira R, Costa H, Carneiro S, Rocha M, and Mendes R. *Reconstructing transcriptional Regulatory Networks using data integration and Text Mining.* In *2015 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 1552–1558. IEEE, November 2015. ISBN 978-1-4673-6799-8.

[6] Crick F. *Central Dogma of Molecular Biology.* Nature, 227, 561, 1970.

[7] Nomura M, Mizushima S, Ozaki M, Traub P, and Lowry C. *Structure and function of ribosomes and their molecular components.* In *Cold Spring Harbor symposia on quantitative biology*, volume 34, pages 49–61. Cold Spring Harbor Laboratory Press, 1969.

[8] Ideker T, Galitski T, and Hood L. *A new approach to decoding life: Systems Biology.* Annu. Rev. Genomics Hum. Genet., 2, 343, 2001. ISSN 1527-8204.

[9] Browne F, Wang H, Zheng H, and Azuaje F. *GRIP: A web-based system for constructing Gold Standard datasets for protein-protein interaction prediction.* Source code for biology and medicine, 4, 2, 2009. ISSN 1751-0473.

[10] Barabási AL and Oltvai ZN. *Network biology: understanding the cell's functional organization.* Nature reviews. Genetics, 5(2), 101, March 2004. ISSN 1471-0056.

[11] Mulder NJ, Akinola RO, Mazandu GK, and Rapanoel H. *Using biological networks to improve our understanding of infectious diseases.* Computational and structural biotechnology journal, 11(18), 1, 2014. ISSN 2001-0370.

[12] Macneil LT and Walhout AJM. *Gene regulatory networks and the role of robustness and stochasticity in the control of gene expression.* Genome research, 21, 645, 2011.

[13] Schlitt T and Brazma A. *Current approaches to gene regulatory network modelling.* BMC bioinformatics, 8 Suppl 6, S9, January 2007. ISSN 1471-2105.

[14] Sanz J, Navarro J, Arbués A, Martín C, Marijuán PC, and Moreno Y. *The transcriptional regulatory network of Mycobacterium tuberculosis.* PloS one, 6(7), e22178, January 2011. ISSN 1932-6203.

[15] Barrett CL and Palsson BO. *Iterative reconstruction of transcriptional regulatory networks: an algorithmic approach.* PLoS computational biology, 2(5), e52, May 2006. ISSN 1553-7358.

[16] Ben-Tabou de Leon S and Davidson EH. *Gene regulation: gene control network in development.* Annual review of biophysics and biomolecular structure, 36, 191, January 2007. ISSN 1056-8700.

[17] Babu MM, Luscombe NM, Aravind L, Gerstein M, and Teichmann SA. *Structure and evolution of transcriptional regulatory networks.* Current opinion in structural biology, 14(3), 283, 2004.

[18] Carrera J, Rodrigo G, Jaramillo A, and Elena SF. *Reverse-engineering the Arabidopsis thaliana transcriptional network under changing environmental conditions.* Genome biology, 10(9), R96, January 2009. ISSN 1465-6914.

[19] Yusuf D, Butland SL, Swanson MI, Bolotin E, Ticoll A, Cheung Wa, Zhang XYC, Dickman CTD, Fulton DL, Lim JS, Schnabl JM, Ramos OHP, Vasseur-Cognet M, de Leeuw CN, Simpson EM, Ryffel GU, Lam EWF, Kist R, Wilson MSC, Marco-Ferreres R, Brosens JJ, Beccari LL, Bovolenta P, Benayoun Ba, Monteiro LJ, Schwenen HDC, Grontved L, Wederell E, Mandrup S, Veitia Ra, Chakravarthy H, Hoodless Pa, Mancarelli MM, Torbett BE, Banham AH, Reddy SP, Cullum RL, Liedtke M, Tschan MP, Vaz M, Rizzino A, Zannini M, Frietze S, Farnham PJ, Eijkelenboom A, Brown PJ, Laperrière D, Leprince D, de Cristofaro T, Prince KL, Putker M, del Peso L, Camenisch G, Wenger RH, Mikula M, Rozendaal M, Mader S, Ostrowski J, Rhodes SJ, Van Rechem C, Boulay G, Olechnowicz SWZ, Breslin MB, Lan MS, Nanan KK, Wegner M, Hou J, Mullen RD, Colvin SC, Noy PJ, Webb CF, Witek ME, Ferrell S, Daniel JM, Park J, Waldman Sa, Peet DJ, Taggart M, Jayaraman PS, Karrich JJ, Blom B, Vesuna F, O'Geen H, Sun Y, Gronostajski RM, Woodcroft MW, Hough MR, Chen E, Europe-Finner GN, Karolczak-Bayatti M, Bailey J, Hankinson O, Raman V, LeBrun DP, Biswal S, Harvey CJ, DeBruyne JP, Hogenesch JB, Hevner

RF, Héligon C, Luo XM, Blank MC, Millen KJ, Sharlin DS, Forrest D, Dahlman-Wright K, Zhao C, Mishima Y, Sinha S, Chakrabarti R, Portales-Casamar E, Sladek FM, Bradley PH, and Wasserman WW. *The transcription factor encyclopedia.* Genome biology, 13(3), R24, January 2012. ISSN 1465-6914.

[20] Sun N and Zhao H. *Reconstructing transcriptional regulatory networks through genomics data.* Statistical methods in medical research, 18(6), 595, December 2009. ISSN 1477-0334.

[21] Shen-Orr SS, Milo R, Mangan S, and Alon U. *Network motifs in the transcriptional regulation network of Escherichia coli.* Nature genetics, 31(1), 64, May 2002. ISSN 1061-4036.

[22] Kauffman Sa. *Metabolic stability and epigenesis in randomly constructed genetic nets.* Journal of theoretical biology, 22(3), 437, 1969. ISSN 00225193.

[23] Lovrics A, Gao Y, Juhász B, Bock I, Byrne HM, Dinnyés A, and Kovács Ka. *Boolean Modelling Reveals New Regulatory Connections between Transcription Factors Orchestrating the Development of the Ventral Spinal Cord.* PloS one, 9(11), e111430, 2014. ISSN 1932-6203.

[24] van Someren EP, Wessels LF, and Reinders MJ. *Linear modeling of genetic networks from experimental data.* Proceedings of International Conference on Intelligent Systems for Molecular Biology, 8, 355, 2000. ISSN 1553-0833.

[25] Friedman N, Linial M, Nachman I, and Pe'er D. *Using Bayesian Networks to Analyze Expression Data.* Journal of Computational Biology, 7(3-4), 601, 2000. ISSN 1066-5277.

[26] Weaver DC. *Modeling regulatory networks with weight matrices.* In *Pacific Symposium on Biocomputing*, volume 4, pages 112–123. 1999.

[27] Shmulevich I, Dougherty ER, Kim S, and Zhang W. *Probabilistic Boolean Networks: a rule-based uncertainty model for gene regulatory networks.* Bioinformatics, 18(2), 261, 2002. ISSN 1367-4803.

[28] US National Library of Medicine. *National Center for Biotechnology Information.* http://www.ncbi.nlm.nih.gov/. [Online; accessed December-2012].

[29] Karp PD, Riley M, Paley SM, Pellegrini-Toole a, and Krummenacker M. *EcoCyc: Encyclopedia of E. coli Genes and Metabolism.* Nucleic Acids Research, 25(1), 43, 1997.

[30] Keseler IM, Collado-Vides J, Gama-Castro S, Ingraham J, Paley S, Paulsen IT, Peralta-Gil M, and Karp PD. *EcoCyc: A comprehensive database resource for Escherichia coli.* Nucleic Acids Research, 33, 334, 2005. ISSN 03051048.

[31] Moszer I, Medigue C, and Viari A. *PyloriGene Web Server.* http://genolist.pasteur.fr/PyloriGene/. [Online; accessed December-2012].

[32] Boneca IG, Reuse HD, Epinat Jc, Pupin M, Labigne A, and Moszer I. *A revised annotation and comparative analysis of Helicobacter pylori genomes.* Nucleic Acids Research, 31(6), 1704, March 2003. ISSN 13624962.

[33] Aoki KF and Kanehisa M. *Using the KEGG database resource.* Current Protocols in Bioinformatics, pages 1–12, 2005.

[34] Kanehisa M and Goto S. *KEGG: kyoto encyclopedia of genes and genomes.* Nucleic acids research, 28(1), 27, January 2000. ISSN 0305-1048.

[35] Tecnische Universitat Braunschweig. *BRENDA - Braunschweig Enzyme Database.* http://brenda-enzymes.org. [Online; accessed January-2012].

[36] Schomburg I, Chang A, and Schomburg D. *BRENDA, enzyme data and metabolic information.* Nucleic acids research, 30(1), 47, 2002.

[37] Jensen LJ, Kuhn M, Stark M, Chaffron S, Creevey C, Muller J, Doerks T, Julien P, Roth A, Simonovic M, Bork P, and von Mering C. *STRING 8–a global view on proteins and their functional interactions in 630 organisms.* Nucleic acids research, 37(Database issue), D412, January 2009. ISSN 1362-4962.

[38] Franceschini a, Szklarczyk D, Frankild S, Kuhn M, Simonovic M, Roth a, Lin J, Minguez P, Bork P, von Mering C, and Jensen LJ. *STRING v9.1: protein-protein interaction networks, with increased coverage and integration.* Nucleic Acids Research, 41(November 2012), 808, November 2012. ISSN 0305-1048.

[39] Szklarczyk D, Franceschini A, Kuhn M, Simonovic M, Roth A, Minguez P, Doerks T, Stark M, Muller J, Bork P, Jensen LJ, and von Mering C. *The STRING database in 2011: functional interaction networks of proteins, globally integrated and scored.* Nucleic acids research, 39(Database issue), D561, January 2011. ISSN 1362-4962.

[40] Consortium TU. *The Universal Protein Resource (UniProt).* Nucleic acids research, 35(Database issue), D193, January 2007. ISSN 1362-4962.

[41] Consortium TU. *The Universal Protein Resource (UniProt) in 2010.* Nucleic acids research, 38(Database issue), D142, January 2010. ISSN 1362-4962.

[42] Salgado H, Gama-Castro S, Peralta-Gil M, Díaz-Peredo E, Sánchez-Solano F, Santos-Zavaleta A, Martínez-Flores I, Jiménez-Jacinto V, Bonavides-Martínez C, Segura-Salazar J, Martínez-Antonio A, and Collado-Vides J. *RegulonDB (version 5.0): Escherichia coli K-12 transcriptional regulatory network, operon organization, and growth conditions.* Nucleic acids research, 34(Database issue), D394, January 2006. ISSN 1362-4962.

[43] Salgado H, Santos a, Garza-Ramos U, van Helden J, Díaz E, and Collado-Vides J. *RegulonDB (version 2.0): a database on transcriptional regulation in Escherichia coli.* Nucleic acids research, 27(1), 59, January 1999. ISSN 0305-1048.

[44] Gasteiger E, Jung E, and Bairoch a. *SWISS-PROT: connecting biomolecular knowledge via a protein database.* Current issues in molecular biology, 3(3), 47, July 2001. ISSN 1467-3037.

[45] Bairoch a and Apweiler R. *The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000.* Nucleic acids research, 28(1), 45, January 2000. ISSN 0305-1048.

[46] UniProt Consortium. *UniProtKB/Swiss-Prot.* `http://http://web.expasy.org/docs/swiss-prot_guideline.html`. [Online; accessed January-2012].

[47] Caragea D, Pathak J, and Bao J. *Information integration and knowledge acquisition from semantically heterogeneous biological data sources.* Data Integration in the Life Sciences, pages 175–190, 2005.

[48] Jenkinson AM, Albrecht M, Birney E, Blankenburg H, Down T, Finn RD, Hermjakob H, Hubbard TJP, Jimenez RC, Jones P, Kähäri A, Kulesha E, Macías JR, Reeves Ga, and Prlić A. *Integrating biological data–the Distributed Annotation System.* BMC bioinformatics, 9 Suppl 8, S3, January 2008. ISSN 1471-2105.

[49] Etzold T, Harris H, and Beaulah S. *SRS: An integration platform for databanks and analysis tools in bioinformatics.* In Lacroix Z and Critchlow T, editors, *Bioinformatics – Managing Scientific Data*, pages 109–146. Morgan Kaufmann Publishers, 2003.

[50] Davidson SB, Crabtree J, Brunk BP, Schug J, Tannen V, Overton GC, and Stoeckert CJ Jr. *K2/Kleisli and GUS: Experiments in integrated access to genomic data sources.* IBM Systems Journal, 40(2), 512, 2001. ISSN 0018-8670.

[51] Davidson S, Overton C, Tannen V, and Wong L. *BioKleisli: A digital library for biomedical researchers.* International Journal on Digital Libraries, pages 36–53, 1997.

[52] Haas LM, Schwarz PM, Kodali P, Kotlar E, Rice JE, and Swope WC. *DiscoveryLink: a system for integrated access to life sciences data sources.* IBM Syst. J., 40(2), 489, February 2001. ISSN 0018-8670.

[53] Stevens R, Baker P, Bechhofer S, Ng G, Jacoby A, Paton NW, and Goble CA. *TAMBIS : Transparent Access to Multiple.* Bioinformatics Application Notes, 16(2), 184, 2000.

[54] Kosky A, Chen I, Markowitz V, and Szeto E. *Exploring heterogeneous biological databases: Tools and applications.* Lecture Notes in Computer Science, 1377, 499, 1998.

[55] Donelson L, Tarczy-hornoch P, Mork P, Dolan C, Mitchell JA, Barrier M, and Mei H. *The BioMediator System as a Data Integration Tool to Answer Diverse Biologic Queries BioMediator System Overview.* Studies in Health Technology and Informatics, 107, 768, 2004.

[56] Zdobnov EM, Lopez R, Apweiler R, and Etzold T. *The EBI SRS server–recent developments.* Bioinformatics (Oxford, England), 18(2), 368, February 2002. ISSN 1367-4803.

[57] Wong L. *The Collection Programming Language.* Kent Ridge Digital Labs, 21, 1, 1996.

[58] Wong L. *Technologies for integrating biological data.* Briefings in bioinformatics, 3(4), 389, December 2002. ISSN 1467-5463.

[59] Chen IMa and Markowitz VM. *An overview of the Object Protocol Model (OPM) and the OPM data management tools.* Information Systems, 20(5), 393, July 1995. ISSN 03064379.

[60] Services C, Demonstration D, Sciences L, and Team S. *IBM Life Sciences Solutions : Turning Data into Discovery with DiscoveryLink.*

IBM Corporation, International Technical Support Organization, San Jose, California, first edition, 2002.

[61] Fluck J and Hofmann-Apitius M. *Text mining for systems biology.* Drug Discovery Today, 19(2), 140, 2014. ISSN 13596446.

[62] Krallinger M and Valencia A. *Text-mining and information-retrieval services for molecular biology.* Genome biology, 6(7), 224, 2005. ISSN 1465-6914.

[63] Jensen LJ, Saric J, and Bork P. *Literature mining for the biologist: from information retrieval to biological discovery.* Nature reviews. Genetics, 7(2), 119, 2006. ISSN 1471-0056.

[64] Yang Y, Adelstein SJ, and Kassis AI. *Target discovery from data mining approaches.* Drug discovery today, 17S(3-4), S216, 2012. ISSN 1878-5832.

[65] Mansouri A, Affendey LS, and Mamat A. *Named Entity Recognition Approaches.* Journal of Computer Science, 8(2), 339, 2008. ISSN 03784169.

[66] Lawrence R and Biing-Hwang J. *An Introduction to Hidden Markov Models.* ASSP Magazine, 3.1(January), 4, 1986.

[67] Herst M, Dumais T, Osman E, Platt J, and Scholkopf B. *Support vector machines.* In *IEEE Inteligent Systems and Their Applications*, pages 18–28. IEEE Comput. Soc, 1998. ISBN 9780387772424 0387772421 9780387772417 0387772413. ISSN 1613-9011.

[68] Lafferty J, McCallum A, and Pereira FCN. *Conditional random fields: Probabilistic models for segmenting and labeling sequence data.* In *ICML '01 Proceedings of the Eighteenth International Conference on Machine Learning*, volume 8, pages 282–289. 2001. ISBN 1558607781. ISSN 1750-2799.

[69] Settles B. *ABNER: An open source tool for automatically tagging genes, proteins and other entity names in text.* Bioinformatics, 21(14), 3191, 2005. ISSN 13674803.

[70] Leaman R and Gonzalez G. *BANNER: an executable survey of advances in biomedical named entity recognition.* In *Pacific Symposium on Biocomputing.*, volume 663, pages 652–663. 2008. ISBN 2335-6936 (Print). ISSN 2335-6936.

[71] Usié A, Alves R, Solsona F, Vázquez M, and Valencia A. *CheNER: Chemical named entity recognizer.* Bioinformatics Applications Note, 30(7), 1039, 2014. ISSN 14602059.

[72] Neves ML, Carazo JM, and Pascual-Montano A. *Moara: a Java library for extracting and normalizing gene and protein mentions.* BMC bioinformatics, 11, 157, 2010. ISSN 1471-2105.

[73] Hunter L, Lu Z, Firby J, Baumgartner Wa, Johnson HL, Ogren PV, and Cohen KB. *OpenDMAP: an open source, ontology-driven concept analysis engine, with applications to capturing knowledge regarding protein transport, protein interactions and cell-type-specific gene expression.* BMC bioinformatics, 9, 78, 2008. ISSN 1471-2105.

[74] Jessop DM, Adams SE, Willighagen EL, Hawizy L, and Murray-Rust P. *OSCAR4: a flexible architecture for chemical text-mining.* Journal of cheminformatics, 3(1), 41, 2011. ISSN 1758-2946.

[75] Gerner M, Nenadic G, and Bergman CM. *LINNAEUS: a species name identification system for biomedical literature.* BMC bioinformatics, 11, 85, 2010. ISSN 1471-2105.

[76] Cunningham H. *GATE, a General Architecture for Text Engineering.* Computers and the Humanities, 36(2), 223, 2002. ISSN 0010-4817.

[77] Campos D, Matos S, and Oliveira J. *Neji: a tool for heterogeneous biomedical concept identification.* Proceedings of BioLINK SIG, 20, 1178, 2013. ISSN 1367-4803.

[78] Huang M, Liu J, and Zhu X. *GeneTUKit: a software for document-level gene normalization.* Bioinformatics (Oxford, England), 27(7), 1032, 2011. ISSN 1367-4811.

[79] Hakenberg J, Gerner M, Haeussler M, Solt I, Plake C, Schroeder M, Gonzalez G, Nenadic G, and Bergman CM. *The GNAT library for local and remote gene mention normalization.* Bioinformatics, 27(19), 2769, 2011. ISSN 1367-4803.

[80] Lourenço A, Carreira R, Carneiro S, Maia P, Glez-Peña D, Fdez-Riverola F, Ferreira EC, Rocha I, and Rocha M. *@Note: A workbench for Biomedical Text Mining.* Journal of Biomedical Informatics, 42(4), 710, 2009. ISSN 15320464.

[81] Grishman R. *Natural Language Processing.* Journal of the American Society for Information Science, 35(1), 291, 1984. ISSN 0267-1905.

[82] Nadkarni PM, Ohno-Machado L, and Chapman WW. *Natural language processing: an introduction.* Journal of the American Medical Informatics Association, 18(5), 544, 2011. ISSN 1067-5027.

[83] Habert B, Adda G, and Adda-Decker M. *Towards tokenization evaluation.* In *Proceedings of First International Conference on Language Resources and Evaluation (LREC)*, pages 427–431. 1998.

[84] Webster J and Kit C. *Tokenization as the intial phase in NLP.* In *Proceedings of 3th Conference on Computational Linguistics*, pages 1106–1110. Nantes, 1992.

[85] Tomanek K, Wermter J, and Hahn U. *Sentence and token splitting based on conditional random fields.* In *Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics*, pages 49–57. 2007.

[86] Tsuruoka Y, Tateishi Y, Kim JD, Ohta T, McNaught J, Ananiadou S, and Tsujii J. *Developing a robust part-of-speech tagger for biomedical text.* Advances in informatics, pages 382–392, 2005.

[87] Nadkarni PM, Ohno-Machado L, and Chapman WW. *Natural language processing: an introduction.* Journal of the American Medical Informatics Association, 18(5), 544, 2011. ISSN 1067-5027.

[88] Smeaton AF. *Using NLP or NLP Resources for Information Retrieval Tasks.* In Strzalkowski T, editor, *Natural Language Information Retrieval*, volume 7 of *Text, Speech and Language Technology*, pages 99–111. Springer Netherlands, 1999. ISBN 978-90-481-5209-4.

[89] Berwick RC. *Principle-based parsing.* Report, Massachusetts Institute of Technology, June 1987.

[90] Balfourier JM, Blache P, and Van Rullen T. *From shallow to deep parsing using constraint satisfaction.* In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. 2002.

[91] Levy JP, Bullinaria Ja, and Patel M. *Explorations in the derivation of semantic representations from word co-occurrence statistics.* South Pacific Journal of Psychology, 10, 111, 1998.

[92] Franceschini A, Szklarczyk D, Frankild S, Kuhn M, Simonovic M, Roth A, Lin J, Minguez P, Bork P, von Mering C, and Jensen LJ. *STRING v9.1: protein-protein interaction networks, with increased coverage and integration.* Nucleic Acids Research, 41(D1), D808, 2013.

[93] Li G, Ross KE, Arighi CN, Peng Y, Wu CH, and Vijay-Shanker K. *miRTex: A Text Mining System for miRNA-Gene Relation Extraction.* PLOS Computational Biology, 11, e1004391, 2015. ISSN 1553-7358.

[94] Michie D, Spiegelhalter DJ, Taylor CC, and Campbell J, editors. *Machine Learning, Neural and Statistical Classification.* Ellis Horwood, Upper Saddle River, NJ, USA, 1994. ISBN 0-13-106360-X.

[95] Ben Abacha A, Zweigenbaum P, Abacha AB, and Zweigenbaum P. *A hybrid approach for the extraction of semantic relations from MEDLINE abstracts.* Lecture Notes in Computer Science (including sub-

series Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 6609 LNCS, 139, 2011. ISSN 03029743.

[96] Hsu CW, Chang CC, and Lin CJ. *A Practical Guide to Support Vector Classification.* Technical Report 1, Department of Computer Science, National University of Taiwan, Taiwan, 2008.

[97] Andrade Ma and Valencia A. *Automatic extraction of keywords from scientific text: Application to the knowledge domain of protein families.* Bioinformatics, 14(7), 600, 1998. ISSN 13674803.

[98] Gruber TR. *A translation approach to portable ontology specifications.* Knowledge Acquisition, 5(2), 199, 1993. ISSN 1042-8143.

[99] Krallinger M, Rodriguez-Penagos C, Tendulkar A, and Valencia A. *PLAN2L: a web tool for integrated text mining and literature-based bioentity relation extraction.* Nucleic acids research, 37(Web Server issue), W160, 2009. ISSN 1362-4962.

[100] Ananiadou S, Pyysalo S, Tsujii J, and Kell DB. *Event extraction for systems biology by text mining the literature.* Trends in biotechnology, 28(7), 381, 2010. ISSN 1879-3096.

[101] Consortium TGO. *Gene Ontology : tool for the unification of biology.* Nature Genetics, 25(may), 25, 2000. ISSN 1061-4036.

[102] Witten IH, Don KJ, Dewsnip M, and Tablan V. *Text mining in a digital library.* International Journal on Digital Libraries, 4(1), 56, 2004. ISSN 14325012.

[103] Neerincx PBT and Leunissen JAM. *Evolution of web services in bioinformatics.* Briefings in Bioinformatics, 6(2), 178, 2005.

[104] Cokelaer T, Pultz D, Harder LM, Serra-Musach J, and Saez-Rodriguez J. *BioServices: a common Python package to access biological Web Services programmatically.* Bioinformatics (Oxford, England), 29(24), 3241, 2013. ISSN 1367-4811.

[105] Schultheiss S, Münch M, Andreeva G, and Rätsch G. *Persistence and availability of Web services in computational biology.* PloS one, 6(9), e24914, January 2011. ISSN 1932-6203.

[106] Blattner FR. *The Complete Genome Sequence of Escherichia coli K-12.* Science, 277(5331), 1453, September 1997. ISSN 00368075.

[107] Sayers E. *Entrez Programming Utilities Help.* National Center for Biotechnology Information (US), 2008.

[108] Fielding R and Taylor R. *Principled design of the modern Web architecture.* Proceedings of the 2000 International Conference on Software Engineering. ICSE 2000 the New Millennium, 2(2), 115, 2000. ISSN 0270-5257.

[109] Shi X. *Sharing service semantics using SOAP-based and REST Web services.* IT Professional, 8(2), 18, 2006. ISSN 15209202.

[110] Friedman C, Kra P, Yu H, and Rzhetsky A. *GENIES : a natural-language processing system journal articles.* Bioinformatics, 17, 2001.

[111] Temkin JM and Gilder MR. *Extraction of protein interaction information from unstructured text using a context-free grammar.* Bioinformatics, 19(16), 2046, October 2003. ISSN 1367-4803.

[112] Spyns P. *Natural language processing in medicine: An overview.* Methods of Information in Medicine, 35(4-5), 285, 1996. ISSN 00261270.

[113] Neumann G and Piskorski J. *A Shallow Text Processing Core Engine.* Computational Intelligence, 18, 451, 2002.

[114] Crysmann B, Frank A, Kiefer B, Krieger HU, Müller S, Neumann G, Piskorski J, Schäfer U, Siegel M, Uszkoreit H, and Xu F. *An Integrated Architecture for Shallow and Deep Processing.* In *University of Pennsylvania*, pages 441–448. 2002.

[115] Kubler S, McDonald R, Nivre J, and Hirst G. *Dependency Parsing.* Morgan and Claypool Publishers, 2009. ISBN 1598295969, 9781598295962.

[116] Cunningham H. *GATE, a general architecture for text engineering.* Computers and the Humanities, 36(2), 223, 2002. ISSN 00104817.

[117] Suderman M and Hallett M. *Tools for visually exploring biological networks.* Bioinformatics, 23(20), 2651, 2007. ISSN 13674803.

[118] Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, Amin N, Schwikowski B, and Ideker T. *Cytoscape: a software environment for integrated models of biomolecular interaction networks.* Genome research, 13(Karp 2001), 2498, 2003. ISSN 1088-9051.

[119] Punin J and Krishnamoorthy M. *XGMML (eXtensible Graph Markup and Modeling Language) 1.0 Draft Specification.*, 2001.

[120] Covert MW, Knight EM, Reed JL, Herrgard MJ, and Palsson BO. *Integrating high-throughput and computational data elucidates bacterial networks.* Nature, 429(6987), 92, May 2004. ISSN 0028-0836.

[121] Gavin AC, Aloy P, Grandi P, Krause R, Boesche M, Marzioch M, Rau C, Jensen LJ, Bastuck S, Dümpelfeld B, Edelmann A, Heurtier MA, Hoffman V, Hoefert C, Klein K, Hudak M, Michon AM, Schelder M, Schirle M, Remor M, Rudi T, Hooper S, Bauer A, Bouwmeester T, Casari G, Drewes G, Neubauer G, Rick JM, Kuster B, Bork P, Russell RB, and Superti-Furga G. *Proteome survey reveals modularity of the yeast cell machinery.* Nature, 440(7084), 631, 2006. ISSN 0028-0836.

[122] Albert R. *Boolean Modeling of Genetic Regulatory Networks.* In *Complex networks*, pages 459–481. Springer, 2004.

[123] Stock AM, Robinson VL, and Goudreau PN. *Two-component signal transduction.* Annual review of biochemistry, 69(1), 183, 2000.

[124] Keseler IM, Collado-Vides J, Santos-Zavaleta A, Peralta-Gil M, Gama-Castro S, Muniz-Rascado L, Bonavides-Martinez C, Paley S, Krummenacker M, Altman T, Kaipa P, Spaulding A, Pacheco J, Latendresse M, Fulcher C, Sarker M, Shearer AG, Mackie A, Paulsen I, Gunsalus RP, and Karp PD. *EcoCyc: A comprehensive database of Escherichia coli biology.* Nucleic Acids Research, 39(November 2010), 583, 2011. ISSN 03051048.

[125] Alvarez AF and Georgellis D. *Chapter 12 - In Vitro and In Vivo Analysis of the ArcB/A Redox Signaling Pathway.* In *Methods in Enzymology: Two-Component Signaling Systems, Part C*, volume 471 of *Methods in Enzymology*, pages 205 – 228. Academic Press, 2010.

[126] Lioliou EE and Kyriakidis Da. *The role of bacterial antizyme: From an inhibitory protein to AtoC transcriptional regulator.* Microbial cell factories, 3(1), 8, 2004. ISSN 1475-2859.

[127] Suzuki K, Wang X, Weilbacher T, Pernestig Ak, Georgellis D, Babitzke P, and Romeo T. *Regulatory Circuitry of the CsrA / CsrB and BarA / UvrY Systems of Escherichia coli.* Journal of bacteriology, 184(18), 5130, 2002. ISSN 0021-9193.

[128] Vogt SL and Raivio TL. *Just scratching the surface: An expanding view of the Cpx envelope stress response.* FEMS Microbiology Letters, 326(1), 2, 2012. ISSN 03781097.

[129] Reitzer L. *Nitrogen assimilation and global regulation in Escherichia coli.* Annual Reviews in Microbiology, 57(1), 155, 2003.

[130] Christopher B, Ti W, and Stock AM. *Comprehensive Analysis of OmpR Phosphorylation, Dimerization and DNA Binding Supports a Canonical Model for Activation.* Journal of molecular biology, 425(10), 1612, 2013. ISSN 08966273.

[131] Heermann R and Jung K. *The complexity of the 'simple'two-component system KdpD/KdpE in Escherichia coli.* FEMS microbiology letters, 304(2), 97, 2010.

[132] Stewart V. *Nitrate-and nitrite-responsive sensors NarX and NarQ of proteobacteria.* Biochemical Society Transactions, 31(1), 1, 2003.

[133] Stewart V. *Nitrate regulation of anaerobic respiratory gene expression in Escherichia coli.* Molecular microbiology, 9(3), 425, 1993.

[134] Hsieh YJ and Wanner BL. *Global regulation by the seven-component P i signaling system.* Current opinion in microbiology, 13(2), 198, 2010.

[135] Clarke MB and Sperandio V. *Transcriptional regulation of flhDC by QseBC and Sigma 28 (FliA) in enterohaemorrhagic Escherichia coli.* Molecular Microbiology, 57(6), 1734, 2005. ISSN 0950382X.

[136] Bordi C, Théraulaz L, Méjean V, and Jourlin-Castelli C. *Anticipating an alkaline stress through the Tor phosphorelay system in Escherichia coli.* Molecular microbiology, 48(1), 211, 2003.

[137] Eguchi Y and Utsumi R. *Alkali metals in addition to acidic pH activate the EvgS histidine kinase sensor in Escherichia coli.* Journal of bacteriology, 196(17), 3140, 2014.

[138] Ranjit DK and Young KD. *The Rcs stress response and accessory envelope proteins are required for de novo generation of cell shape in Escherichia coli.* Journal of bacteriology, 195(11), 2452, 2013.

[139] Eichhorn E, van der Ploeg JR, and Leisinger T. *Characterization of a two-component alkanesulfonate monooxygenase from Escherichia coli.* Journal of Biological Chemistry, 274(38), 26639, 1999.

[140] The Gene Ontology Consortium. *Gene Ontology: tool for the unification of biology.* Nature Genetics, 25(may), 25, 2000. ISSN 1061-4036.

[141] Mi H, Muruganujan A, Casagrande JT, and Thomas PD. *Large-scale gene function analysis with the PANTHER classification system.* Nature protocols, 8(8), 1551, 2013.

[142] Mi H, Poudel S, Muruganujan A, Casagrande JT, and Thomas PD. *PANTHER version 10: expanded protein families and functions, and analysis tools.* Nucleic Acids Res, 44(D1), D336, 2016. ISSN 1362-4962.

[143] Mi H, Muruganujan A, and Thomas PD. *PANTHER in 2013: Modeling the evolution of gene function, and other gene attributes, in the context of phylogenetic trees.* Nucleic Acids Research, 41(D1), 377, 2013. ISSN 03051048.

[144] Mi H and Thomas P. *PANTHER pathway: an ontology-based pathway database coupled with data analysis tools.* In *Protein Networks and Pathway Analysis*, pages 123–140. Springer, 2009.

[145] Supek F, Bošnjak M, Škunca N, and Šmuc T. *Revigo summarizes and visualizes long lists of gene ontology terms.* PLoS ONE, 6(7), 2011. ISSN 19326203.

[146] Nielsen J. *Metabolic engineering.* Applied Microbiology and Biotechnology, 55(3), 263, 2001. ISSN 01757598.

[147] Mahner M and Kary M. *What exactly are genomes, genotypes and phenotypes? And what about phenomes?* Journal of theoretical biology, 186(1), 55, 1997. ISSN 0022-5193.

[148] Sauer U. *Evolutionary engineering of industrially important microbial phenotypes.* Advances in biochemical engineering/biotechnology, 73, 129, 2001. ISSN 0724-6145.

[149] Kauffman KJ, Prakash P, and Edwards JS. *Advances in flux balance analysis.* Current Opinion in Biotechnology, 14(5), 491, 2003. ISSN 09581669.

[150] Shlomi T, Berkman O, and Ruppin E. *Regulatory on/off minimization of metabolic flux.* Pnas, 102(21), 7695, 2005.

[151] Segrè D, Vitkup D, and Church GM. *Analysis of optimality in natural and perturbed metabolic networks.* Proceedings of the National

Academy of Sciences of the United States of America, 99(23), 15112, 2002. ISSN 0027-8424.

[152] Rocha I, Maia P, Evangelista P, Vilaça P, Soares S, Pinto JP, Nielsen J, Patil KR, Ferreira EC, and Rocha M. *OptFlux: an open-source software platform for in silico metabolic engineering.* BMC systems biology, 4(1), 45, 2010. ISSN 1752-0509.

[153] Vilaça P, Ocha I, and Rocha M. *A computational tool for the simulation and optimization of microbial strains accounting integrated metabolic/regulatory information.* BioSystems, 103(3), 435, 2011. ISSN 03032647.

[154] Rocha O, Vilaça P, Rocha M, and Mendes R. *Reg4OptFlux: An OptFlux plug-in that comprises meta-heuristics approaches for Metabolic engineering using integrated models.* 2014 6th World Congress on Nature and Biologically Inspired Computing, NaBIC 2014, pages 226–231, 2014.

[155] Kim J and Reed JL. *OptORF: Optimal metabolic and regulatory perturbations for metabolic engineering of microbial strains.* BMC systems biology, 4, 53, 2010. ISSN 1752-0509.

[156] Maia P, Rocha I, and Rocha M. *An integrated framework for strain optimization.* 2013 IEEE Congress on Evolutionary Computation, CEC 2013, pages 198–205, 2013.

[157] Lee JW. *Advanced biofuels and bioproducts.* Springer Science & Business Media, 2012.

[158] Wang J, Zhu J, Bennett GN, and San KY. *Succinate production from sucrose by metabolic engineered escherichia coli strains under aerobic conditions.* Biotechnology Progress, 27(5), 1242, 2011. ISSN 87567938.

[159] Cleland N and Enfors So. *Control of Glucose-Fed Batch Cultivations of E . coli by Means of an Oxygen Stabilized Enzyme Electrode.* Eu-

ropean Journal of Applied Microbiology and Biotechnology, 18, 141, 1983. ISSN 01711741.

[160] Lee SY, Hong SH, and Moon SY. *In silico metabolic pathway analysis and design: succinic acid production by metabolically engineered Escherichia coli as an example.* Genome Informatics, 13, 214, 2002.

[161] Varma A, Palsson BO, Varma A, and Palsson BO. *Stoichiometric Flux Balance Models Quantitatively Predict Growth and Metabolic By-Product Secretion in Wild-Type Escherichia coli W3110.* Appl. Environ. Microbiol., 60(10), 3724, 1994.

[162] Pigné Y, Dutot A, Guinand F, and Olivier D. *GraphStream: A Tool for bridging the gap between Complex Systems and Dynamic Graphs.* Computing Research Repository, abs/0803.2093, 2008.