# Intelligent Text Processing Techniques for Textual-Profile Gene Characterization

Floriana Esposito, Marenglen Biba, Stefano Ferilli
Department of Computer Science, University of Bari
Via, E. Orabona, 4, 70125, Bari Italy
{esposito@di.uniba.it,biba@di.uniba.it,ferilli@di.uniba.it}

*Keywords*: machine learning, rule induction, clustering, knowledge-based systems, gene prioritization.

**Abstract.** We present a suite of Machine Learning and knowledge-based components for textual-profile based gene prioritization. Most genetic diseases are characterized by many potential candidate genes that can cause the disease. Gene expression analysis typically produces a large number of co-expressed genes that could be potentially responsible for a given disease. Extracting prior knowledge from text-based genomic information sources is essential in order to reduce the list of potential candidate genes to be then further analyzed in laboratory. In this paper we present a suite of Machine Learning algorithms and knowledge-based components for improving the computational gene prioritization process. The suite includes basic Natural Language Processing capabilities, advanced text classification and clustering algorithms, robust information extraction components based on qualitative and quantitative keyword extraction methods and exploitation of lexical knowledge bases for semantic text processing.

## 1 Introduction

Many wide-spread diseases are still an important health concern for our society due to their complexity of functioning or to their unknown causes. Some of them can be acquired but some can be genetic and a large number of genes has already been associated to particular diseases. However, many potential candidate genes are still suspected to cause a certain disease and there is strong motivation for developing computational methods that can help reduce the number of these susceptibility genes. The number of suspected regions of the genome that encode probable disease genes for particular diseases is often estimated to be very large. This gives rise to a really large number of genes to be analyzed which would be infeasible in practice. In order to focus on most promising genes, prioritization methods are being developed in order to exploit prior knowledge on genetic diseases that can help guide the process of identifying novel genes.

Much of the prior background knowledge on genetic diseases is primarily reported in the form of free text. Extracting relevant information from unstructured data has always been a key challenge for Machine Learning methods [2]. These have the power to provide precious capabilities to rank genes based on their textual profile. Indeed, human knowledge on genome and genetic diseases is becoming huge and manual exploitation of the large amount of raw text of scientific papers is infeasible. For this reason, recently many automatic methods for information extraction from text sources have been developed. The approach presented in [1] links groups of genes with relevant MEDLINE abstracts through the PubMed engine, characterizing each group with a set of keywords from MeSH and the UMLS ontology. In [3] co-expression information is linked with the co-citation network constructed on purpose. In this way co-expressed genes are characterized by MeSH keywords appearing in the abstract about genes. In [4] the authors developed an approach called neighborhood divergence that quantifies the functional coherence of a group of genes through a database that links genes to documents. In [5] it was proposed a schema that links abstracts to genes in a probabilistic framework that

uses the EM algorithm to estimate the parameters of the word distributions. Genes are defined similar when the corresponding gene-by-documents representations are close. Finally, in [6] and [7] it is provided a proof of principle on how clustering of genes encoded in a keyword-based representation can be useful to discover relevant subpatterns.

In this paper we describe a suite of machine learning and knowledge-based methods for textual-profile based gene prioritization. We present a classification system based on inductive logic programming [8] that is able to semantically classify texts regarding genes or diseases. The power of the approach lies in the representation language, Horn logic, that permits to describe a text through its logical structure and thus reason about it. Classification theories are induced from examples and are then used to classify novel unseen documents. The input to this machine learning system is given by a knowledge-based component for text-processing. This rule-based system performs a series of NLP steps such as part-of-speech tagging and disambigation in order to produce an accurate representation of the text which is then transformed into a Prolog clause that serves as input for the learning system.

Then we present a novel distance between Horn clauses that is used in this context to define similarities between papers regarding genes. The distance is then used in an instance-based approach to cluster documents of disease genes and candidate genes. Through the mapping gene-to-document provided by EntrezGene, then the clustering of documents can be seen as a clustering of genes and further analysis can be performed to reduce the number of genes to be further analyzed.

Both, the learning system and the instance-based approach are combined with the taxonomic knowledge base WordNET [9, 10]. In the case of the inductive approach WordNET is used to properly generalize theories in presence of similar words that share any common parent nodes in the hierarchy of concepts of WordNET. In the case of the instance-based approach, WordNET is used to semantically define a taxonomic distance between words in the text and include it in the overall distance between two texts. Clustering of documents is then mapped directly to gene clustering using the gene-to-docs mapping of EntrezGene.

We present also two keyword-based approaches for textual-profile gene clustering. The first approach is quantitative in that it combines into the Bayes theorem the frequency of a term (in a document and in a collection) and its position in the document. Through this formula, the probability that a term is a keyword is computed. On the other side, the qualitative approach exploits WordNET Domains [11, 12] to semantically extract keywords from documents using the hierarchy of categories defined in WordNET Domains for each of the synsets found in the document. Both methods are used to generate the $gene \times terms$ matrix which is then combined with the gene-to-docs mapping taken from EntrezGene, to give the final matrix $gene \times terms$. Then classical clustering algorithms are performed on this matrix to discover relationships between disease and candidate genes. The hypothesis is that couple of genes (candidate-disease) within the same cluster and that show high similarity as given by their textual-profile, could be probably involved in the same disease.

The paper is organized as follows: in Section 2 we describe the various components of the suite for gene prioritization, in Section 3 we describe some preliminary experimental scenarios, in Section 4 we discuss related work and the differences with our approach and in Section 5 we conclude.

## 2 Machine Learning and knowledge-based components

In this section we describe the various parts of the suite and how they interact with each other. We also describe how each component is used for the final goal of scoring candidate genes.

### 2.1 Rule-induction system for text classification on diseases

The rule-induction system INTHELEX [13], is an incremental learning system that induces and revises first-order logic (FOL) theories for multiple classes from examples. It uses two inductive refinement operators to fix wrong classifications of the current theory. In case a positive example is rejected, the system generalizes the definition of the corresponding concept by dropping some conditions (ensuring consistency with all the past negative examples), or adding to the current theory (if consistent) a new alternative definition of the concept. When a negative example is explained, a specialization of one of the theory definitions that concur in explaining the example is attempted by adding to it some (positive or negative) conditions which characterize all the past positive examples and discriminate them from the current negative one.

When dealing with examples describing text, the system finds difficulties to generalize theories if lexical knowledge is missing. Therefore, to handle theory refinement for text classification we have introduced in the system an operator for finding common parents of words in the WordNET knowledge base. The following example clarifies how generalization is performed using WordNET:

Let's consider the following phrases:

The enterprise bought shares.

The company acquired stocks.

These represent two examples for the learning system. When the first example comes, the learning system generates a theory to properly classify it. When the second example is given to the system, it fails to explain it and tries to generalize the current theory. It fails to generalize the theory since it does not have any further information on the proper logical leterals to use in the theory revision process. But if we use WordNET, we can navigate the concept hierarchy and find that the couples enterprise-company, buy-acquire and share-stock share common parents, therefore can be generalized using the most specific or general common parent in the WordNET graph.

Each example is expressed in Horn logic and describes the structural composition of the text (subject, verb, complement) and the part-of-speech of each component. This is performed using the rule-based approach described in Section 2.3. The Prolog representation of one of the above examples is:

```
observation(text1) :- phrase(text1,text1_e1), subject(text1_e1,text1_e2),
token(text1_e2,text1_e3), company(text1_e3), noun(text1_e3), verb(text1_e1,text1_e4),
token(text1_e4,text1_e5), buy(text1_e5), past_tense(text1_e5),
complement_object(text1_e1,text1_e6), token(text1_e6,text1_e7),
shares(text1_e7), noun(text1_e7).
```

The induction system is used in the context of gene prioritization on a certain disease to classify documents regarding this disease. The mapping gene-to-doc coming from EntrezGene does not specify if a document is on a particular disease, thus the learning system once trained is used to properly weight the matrix $gene \times documents$ (Figure 1). Training examples are taken from PubMed collecting positive examples on the interesting disease and negative examples on any other disease, where each example is constructed from the document abstract. The learning system induces theories that can be then used to classify unknown texts.

### 2.2 Instance-based system for text clustering regarding genes

Clustering aims at organizing a set of unlabeled patterns into groups of homogeneous elements based on their similarity. The similarity measure exploited to evaluate the distance between elements determines the effectiveness of the clustering algorithms.

Differently from the case of attribute-value representations [17], completely new comparison criteria and measures must be defined for FOL descriptions since they do not induce an Euclidean space. Our proposal is based on a similarity assessment technique for Horn clauses introduced in [14], and exploits the resulting similarity value as a distance measure in a classical k-means clustering algorithm, based on medoids instead of centroids as cluster prototypes guiding the partitioning process (due to the non-Euclidean space issue).

In order to properly deal with text, the distance in [14] is enriched with lexical knowledge from WordNET defining a taxonomic similarity element as part of the overall similarity. This taxonomic similarity between couples of words in the text is defined as the intersection of the graphs of the parents in the is-a hierarchy of concepts in WordNET. The more nodes these graphs have in common the more similar the respective words are.

Clustering is applied to abstracts from EntrezGene in the following way. Disease genes are identified from the domain expert (there are already a large number of repositories which detail known genes about different diseases) and related documents for each gene are taken from EntrezGene. The same is performed from a series of candidate genes suspected to be responsible for the disease. The abstracts are pre-processed with the system presented in Section 2.3 and transformed in Horn clauses. These clauses are given in input to the clustering algorithm which produces a set of clusters containing documents on disease and candidate genes. These clusters provide precious information to be analyzed in order to discover in the same cluster elements regarding disease and candidate genes. This is useful to produce a final prioritization based on the similarity values computed during clustering.

### 2.3 Rule-based system for syntactic and logical analysis of text

Natural language processing (NLP) is one of the classical challenges for Artificial Intelligence and Machine Learning. Many NLP approaches rely on knowledge about natural language instead of statistical training on corpora. Our rule-based system falls among the knowledge-based approaches to NLP. The rule-based component is part of a larger system DOMINUS [15] which performs a number of pre-processing steps to the raw text before it is given in input to the rule-based component. Specifically, after a tokenization step that aims at splitting the text into homogeneous components such as words, values, dates, nouns and a language identification step, the system also carries out additional steps that are language- dependent: PoS-tagging (each word is assigned to the grammatical role it plays in the text), Stopword removal (less frequent or uniformly frequent items in the text, such as articles, prepositions, conjunctions, etc, are ignored to improve effectiveness and efficiency), Stemming (all forms of a term are reported to a standardized form, this way reducing the amount of elements).

After these basic NLP steps, the text is given in input to the rule-based system which performs Syntactic Analysis (yielding the grammatical structure of the sentences in the text) and Logical Analysis (providing the role of the various grammatical component in the sentences). After each grammatical component has been tagged with its semantic role, the structure is saved in an XML format for future use and then transformed in a Prolog representation as described in Section 2.1. The Prolog clause represents an example for the rule-induction system or a training instance for the clustering algorithm.

### 2.4 Quantitative keyword extraction Method

The quantitative keyword extraction method implemented in our system is a naïve Bayes technique based on the concepts of frequency and position of a term and on the independence of such concepts [16]. Indeed, a term is a possible keyword candidate if the frequency of the term is high both in the document and in the collection. Further-

more, the position of a term (both in the whole document and in a specific sentence or section) is an interesting feature to consider, since a keyword is usually positioned at the beginning/end of the text. Such features are combined according to the Bayes Theorem in a formula to calculate the probability of a term to be a keyword in the following way, $P(key|T, D, PT, PS)$ is given by:

$$\frac{P(T|key) * \sum_{i=1}^{|insD|} P(D_i|key) * \sum_{j=1}^{|insT|} P(PT_j|key) * \sum_{k=1}^{|insS|} P(PS_k|key)}{P(\sum_{i=1}^{|insD|} D_i + \sum_{j=1}^{|insT|} PT_j + \sum_{k=1}^{|insS|} PS_k)} \quad (1)$$

where $P(key)$ represents the probability a priori that a term is a keyword (the same for each term), $P(T|key)$ is the standard tf-idf value of the term, $P(D|key)$, respectively $P(PT|key)$ and $P(PS|key)$, are computed by dividing the distance of the first occurrence of the term from the beginning of the section $(D)$, document $(PT)$, sentence $(PS)$ with the number of the terms in the section, respectively document and sentence. Finally, $P(D, PT, PS)$ is computed by adding the distances of the first occurrence of the term from the beginning of the section, document and sentence. Since a term could occur in more than one document, section or sentence, the sum of the values are considered. In this way, the probability for the candidate keyword are calculated and the first k with the highest probability are considered as the final keywords for the document.

In the specific context, keywords are extracted from documents of EntrezGene regarding disease and candidate genes. For each document a set of weighted keywords is extracted and saved into a $document \times term$ matrix. This matrix is then combined with the gene-to-doc mapping given by EntrexGene in order to have the final matrix $gene \times terms$. Different clustering algorithms are then applied to these matrix such as simple k-means, Expectation-Maximization clustering or Cobweb.

### 2.5 Qualitative keyword extraction method that exploits lexical knowledge

In [18] it was shown that quantitative keyword extraction methods are complementary to qualitative ones. The quantitative method basically exploits terms frequency and is more related to a collection of documents, while the qualitative methods, exploiting lexical knowledge are in general more related to the single document. The qualitative method implemented in our suite exploits a WordNET-based density function defined in [19]. The method works as follows: terms not included in WordNet (frequent words such as articles, pronouns, conjunctions and prepositions) are not evaluated for classification, this way implicitly performing a stop-word removal. Given the set $W = t_1, ..., t_n$ of terms in a sentence, each having a set of associated synsets $S(t_i)$, a generic synset s will have weights:

- $p(S(t_i), s) = 1/|S(ti)|$ if $s_k \in S(t_i)$, 0 otherwise, in $S(t_i)$, and
- $p(W, s) = \sum_{i=1,..,n} p(S(t_i), s)/|W|$ in sentence W.

If a term $t$ is not present in WordNet, $S(t)$ is empty and $t$ will not contribute to computation of $|W|$. The weight of a synset associated to a single term $t_i$ is $1/(|W| * |S(ti)|)$. The normalized weight for a sentence is equal to 1. Given a document $D$ made up of $m$ sentences $W_i$ , each with associated weight $w_i > 0$, is $p(D, W_i) = w_i/(\sum_{k=1,..,h}^{w_k})$. The total weight for a document, given by the sum of weights of all its sentences, is equal to 1. Thus, the weight of a synset $s$ in a document can be defined as: $p(D, s) = \sum_{j=1,...,m} p(W_j, s) * p(D, W_j)$. In order to assign a document to a category, the weights of the synsets in the document that refer to the same WordNet Domains category are summed, and the category with highest score is chosen. This Text Categorization technique, differently from traditional ones, represents a static classifier that does not need a training phase, and takes the categories from WordNet Domains.

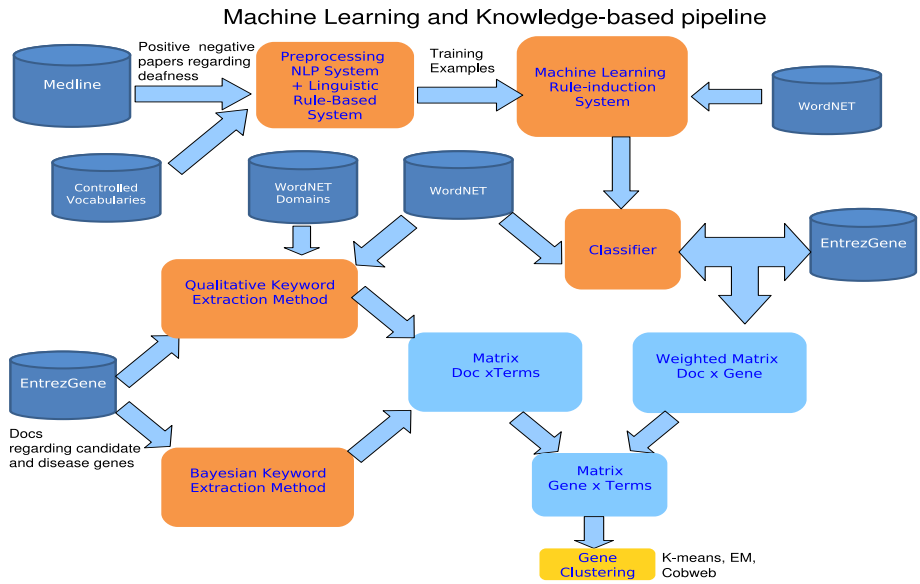


Figure 1: The Machine Learning and knowledge-based framework

The keyword extraction algorithm, after computing the density function, proceeds as follows:

1. sort the list of synsets in the document by decreasing weight;
2. assign to the document the first k terms in the document referred to the synsets with highest weight in the list;
3. for each pair synset-weight create the pair label-weight where label is the one that WordNet Domains assignes to that synset
4. sort by decreasing weight the pairs label-weight;
5. select the first n domain labels that are above a given quality threshold.

After assigning weights to all synsets expressed by the document under processing, the synsets with highest ranking can be selected, and the corresponding terms can be extracted from the document as best representatives of its content.

Keyword extracted with the qualitative methods can be used in the same way as for the quantitative methods as described in the previous section. An interesting point to investigate is the intersection regarding common keywords found by both methods. In [18] it was found that in a large collection of documents, there is an important intersection of the two methods and this can be exploited by taking into consideration only the keywords found by both methods.

## 3 Experimental evaluation

The suite of components is currently being experimented on a number of candidate genes. There are different scenarios in which the framework could be used and currently it is being evalualed how to use it in practice for an important disease. Here we describe each preliminary experiment that we are performing. Figures 1 and 2 present the various components of the pipeline.

**Scenario 1.** Disease and candidate genes are taken from known lists of genes. Abstracts of documents regarding candidate and disease genes are taken from EntrezGene and given in input to the two keyword extraction methods. A $document \times term$ matrix is produced from each of them (a single matrix can be obtained by taking the keywords at the intersection of both methods). Then this matrix is combined with the gen-to-doc mapping of EntrezGene to produce the final matrix $gene \times terms$. This matrix is then transformed into a suitable representation for clustering algorithms such as k-means, EM or Cobweb.

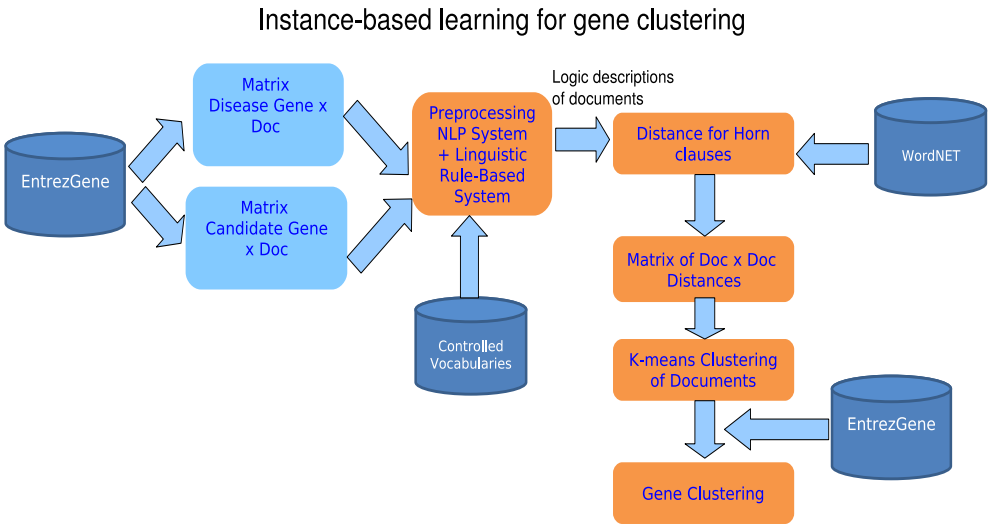


Figure 2: Instance-based learning for gene clustering on first-order descriptions

**Scenario 2.** Scientific papers regarding a certain disease (and not) are taken from MEDLINE, serving as positive and negative examples for the learning system. These are first given in input to the pre-processing engine which performs the basic NLP steps and syntactic and logical analysis of the text. The output of this phase is a structural representation of text with the semantic role of each text element. This representation is transformed into Prolog examples for the rule-induction system which learns a semantic text classifier exploiting also the WordNET lexical knowledge base. The classifier is then used to properly weight papers from EntrezGene (regarding the genes involved) but that do not have any tagging or labeling about the disease. The papers classified as not regarding the given disease are weighted differently from the papers classified as treating that disease. This produces a weighted matrix $gen \times documents$ that combined with the matrix in Scenario 1 gives the final matrix $gene \times terms$ for the clustering of genes.

**Scenario 3.** Disease and candidate genes are taken from domain experts of known lists from online repositories and given in input to the pre-processing component which produces a Prolog clause for each abstract. These are given to the clustering algorithm that uses the similarity function defined on Horn clauses. Clustered documents are then mapped to a gene clustering through the mapping of EntrexGene gene-to-doc.

**Scenario 4.** Scientific papers regarding specific genes are taken from MEDLINE and labeled with the aid of a domain expert. These are given to the pre-processing engine and then to the learning system which produces an accurate text classifier for each gene. Then other unknown papers from MEDLINE are given in input to the classifier which assigns each document to a gene. This process produces a novel gene-to-doc mapping which combined with the matrix $documents \times term$ from Scenario 1, gives the final matrix $gene \times terms$ on which clustering is then performed.

The pipeline is designed also to use domain vocabolaries. Currently the vocabolary to be used is MeSH but we intend to extend to others such as eVOC, GO, OMIM and LDDB.

Current experiments regard genes which are suspected to be involved in a certain disease. Respective documents are planned to be taken from MEDLINE according to the mapping of EntrezGene. **Scenario 1** will be experimented on a collection of a very large number of documents that regard disease and candidate genes. Keywords will be extracted and weighted with the quantitative method described in the previous Sections. For each gene a vector of weights will be produced and then it will be computed the euclidean distance of the vectors. Finally, for each candidate gene the score is given from the sum of the distances of the respective vectors. Complete results on these experiments will be reported in an extended version of this work.

## 4 Related work and discussion

Text-based gene prioritization is receiving much attention and different approaches have been proposed for this purpose. In [20] was proposed a tool that scores concepts in GO according to their relevance to each disease via text mining. Potential candidate genes are then scored through a BLASTX search on reference sequence. Another approach proposed in [21] uses shared GO annotations to identify similarities for genes to be involved in the same disease. A similar text-mining approach was proposed in [22] where candidate gene selection is performed using the eVOC ontology as a controlled vocabulary. eVOC terms are first associated with disease names according to co-occurrence in MEDLINE abstracts. Then the identified terms are ranked and the genes annotated with the top-ranking terms are selected.

One of the few approaches that exploits machine learning and one of the most promising ones is that proposed in [23]. The authors use machine learning to build a model and then rank the test set of candidate genes according to the similarity to the model. The similarity is computed as the correlation for vector space data and BLAST score for sequence data. The advantage of the method is that it incorporates different multiple genomic data sources (microarray, InterPro, BIND, sequence, GO annotation, Motif, Kegg, EST, and text mining). Recently, in [24] was proposed a gene prioritization tool for complex traits which ranks genes by comparing the standard correlation of term-frequency vectors (TF profiles) of annotated terms in different ontological descriptions and integrates multiple ranking results by arithmetical (min, max, and average) and parametric integrations. The most closely related to our approach is that of [25] which exploits the textual profile of genes for their clustering.

The suite of algorithms and components that we propose here differs in many points from these previous approaches. First, to the best of our knowledge, machine learning in the form of rule-induction has not been used before for text-based gene prioritization or clustering. Rule-induction, due to the power of the approach, has been mainly used in biological domains for structural classification of molecules. Relation extration is from text is another important task often faced with rule-induction approaches. But this relations have not yet been used to produce gene characterization profiles. The use of relations could lead to novel and more reliable gene profiles because relations could involve different biological entities of interest and thus important information that was ignored before can be now used to strengthen informative power of the gene profile.

Second, similarity measures used previously for gene prioritization has always been on attribute-value representations, whereas here we use a novel similarity function defined on first-order descriptions. This has the advantage that first-order languages allow for more thorough description of text and this can help capture hidden features of the entities. Moreover, we adopt a novel representation of texts not simply as bag-of-words but as a Horn clause incorporating the syntactic and logical role of elements in the sentence. This helps perform through rule-induction more robust text classification compared to attribute-value based methods. In addition, taxonomic similarity is another novel feature that we exploit in our similarity function in order to better capture the meaning of each text and properly define the similarity between texts. Finally, qualitative and quantitative keyword extraction methods are plugged in together to boost extraction of most significant terms. The qualitative methods, being related to the single document, can find more specialized words that can properly represent the single document. On the other side, the quantitative method, being related to a collection of documents, tries to capture more general words found in the entire collection and that can distinguish between the documents.

## 5 Conclusion

Most genetic diseases are characterized by many potential candidate genes that can cause the disease. Gene expression analysis typically produces a large number of co-expressed genes that could be potentially responsible for a given disease. Extracting prior knowledge from text-based genomic information sources is essential in order to reduce the list of potential candidate genes to be then further analyzed in laboratory. In this paper we present a suite of Machine Learning algorithms and knowledge-based components for improving the computational gene prioritization process. The pipeline includes basic Natural Language Processing capabilities, advanced text classification and clustering algorithms, robust information extraction components based on qualitative and quantitative keyword extraction methods and exploitation of lexical knowledge bases for semantic text processing.

## References


[1] D.R. Masys, J.B. Welsh, J.L. Fink, M. Gribskov, I. Klacansky, J. Corbeil. "Use of keyword hierarchies to interpret gene expression". Bioinformatics, 17:319-326, 2001.

[2] R. Feldman and J. Sanger. "Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data", Cambridge University Press, 2006.

[3] T. Jenssen, A. Laegreid, J. Komorowski, E. Hovig. "A literature network of human genes for high-throughput analysis of gene expression". Nat Genet, 28:21-28, 2001.

[4] S. Raychaudhuri, H. Schutze, R.B. Altman. "Using text analysis to identify functionally coherent gene groups". Genome Res, 12:1582-1590, 2002.

[5] H. Shatkay, S. Edwards, M. Boguski. "Information retrieval meets gene analysis". IEEE Intelligent Systems (Special Issue on Intelligent Systems in Biology), 17:45-53, 2002.

[6] D. Chaussabel, A. Sher. "Mining microarray expression data by literature profiling". Genome Biol 3, 2002.

[7] P. Glenisson, P. Antal, J. Mathys, Y. Moreau, B.D. Moor. "Evaluation of the vector space representation in text-based gene clustering". Pacific Symposium on Biocomputing, 391-402, 2003.

[8] N. Lavrac and S. Dzeroski. "Inductive Logic Programming: Techniques and applications", UK: Ellis Horwood, Chichester, 1994.

[9] G. A. Miller, R. Beckwith, C. Fellbaum, D. Gross, K. Miller. "Introduction to WordNet: An On-line Lexical Database". International Journal of Lexicography, Vol. 3, No. 4, pp. 235-244, 1990.

[10] C. Fellbaum. "WordNet an Electronic Database", Cambridge: MIT Press. pp. 1-23, 1998.

[11] B. Magnini, C.Strapparava, G. Pezzulo, A. Gliozzo. "The role of domain Information in Word Sense Disambiguation". Natural Language Engineering, Vol. 8, No. 4 , pp. 359- 373, 2002.

[12] B. Magnini, G. Cavagli. "Integrating Subject Field Codes into WordNet". ITC-irst, Proc. Second International Conference on Language Resources and Evaluation, LREC2000, pp.1-6, 2000.

[13] F. Esposito, S. Ferilli, N. Fanizzi, T. M. Basile, and N. Di Mauro. "Incremental multistrategy learning for document processing". Applied Artificial Intelligence: An International Journal, 17(8/9):859883, 2003.

[14] S. Ferilli, T.M.A. Basile, M. Biba, N. Di Mauro, F. Esposito. "A General Similarity Framework for Horn Clause Logic". Fundamenta Informaticae Journal, 90(1-2): 43-66, IOS Press, 2009.

[15] F. Esposito, S. Ferilli, T.M.A. Basile, N. Di Mauro. "Machine Learning for Digital Document Processing: From Layout Analysis To Metadata Extraction" - Machine Learning in Document Analysis and Recognition, pp. 105-138, 2008.

[16] Y. Uzun, "Keyword Extraction Using Naïve Bayes", Bilkent University, Department of Computer Science, 2005.

[17] M. Li, X. Chen, X. Li, B. Ma, and P. Vitanyi. "The similarity metric", IEEE Transactions On Information Theory, Vol. 50, No. 12, December, 2004.

[18] S. Ferilli, M. Biba, T.M.A. Basile, F. Esposito. "Combining Qualitative and Quantitative Keyword Extraction Methods with Document Layout Analysis". In Proceedings of 5th Italian Research Conference on Digital Libraries, (IRCDL 2009), DELOS: an Association for Digital Libraries 2009.

[19] M. Angioni, R. Demontis, F. Tuveri. “A Semantic Approach for Resource Cataloguing and Query Resolution”, Communications of SIWN, ISSN 1757-4439, Vol.5, pp. 62-66, 2008.

[20] C. Perez-Iratxeta, M. Wjst, P. Bork, and M.A. Andrade. “G2D: a tool for mining genes associated with disease”. BMC Genet, 6, 45, 2005.

[21] F.S. Turner, D.R. Clutterbuck, and C.A.M. Semple. “POCUS: mining genomic sequence annotation to predict disease genes. Genome Biol”, 4(11), R75, 2003.

[22] N. Tiffin, J.F. Kelso, A.R. Powell, H. Pan, V.B. Bajic, and W.A. Hide. “Integration of text- and data-mining using ontologies successfully selects disease gene candidates”. Nucleic Acids Res, 33(5), 1544-1552, 2005.

[23] S. Aerts, D. Lambrechts, S. Maity, P. Van Loo, B. Coessens, F. De Smet, L.C. Tranchevent, B. De Moor, P. Marynen, B. Hassan, P. Carmeliet, and Y. Moreau. “Gene prioritization through genomic data fusion”. Nat Biotechnol, 24(5), 537-544, 2006.

[24] K.J. Gaulton, K.L. Mohlke, and T. Vision. “A computational system to select candidate genes for complex human traits. Bioinformatics”, 23(9), 1132-1140, 2007.

[25] P. Glenisson, B. Coessens, S. Van Vooren, J. Mathys, Y. Moreau, and B. De Moor. “TXTGate: profiling gene groups with text-based information”. Genome Biol., 5(6), R43, 2004.