

RESEARCH

Open Access

Ranking MEDLINE documents

Célia Talma Gonçalves^{1*}, Rui Camacho² and Eugénio Oliveira³

Abstract

Background: BioTextRetriever is a Web-based search tool for retrieving relevant literature in Molecular Biology and related domains from MEDLINE. The core of BioTextRetriever is the dynamic construction of a classifier capable of selecting relevant papers among the whole MEDLINE bibliographic database. “Relevant” papers, in this context, means papers related to a set of DNA or protein sequences provided as input to the tool by the user.

Methods: Since the number of retrieved papers may be very large, BioTextRetriever uses a novel ranking algorithm to retrieve the most relevant papers first. We have developed a new methodology that enables the automation of the assessment process based on a multi-criteria ranking function. This function combines six factors: MeSH terms, paper’s number of citations, author’s h-index, journals impact factor, author number of publications and journal similarity function.

Results: The best results highlight the number of citations and the h-index factors.

Conclusions: We have developed and a multi-criteria ranking function, that contemplates six factors, and that seems appropriate to retrieve relevant papers out of a huge repository such as MEDLINE.

Keywords: Ranking; Text mining; Machine learning

Background

It is very important for researchers to be aware of the relevant scientific research in their scientific area of knowledge. However the volume of scientific and technical publications in almost all areas of knowledge is growing at a phenomenal rate. Most of these publications are available on the Web. Thus, accessing the right and relevant information amidst this overwhelming amount of information available in the Web is indeed of great importance, albeit difficult in most cases [1].

When trying to find relevant publications, researchers turn to the well known traditional keyword-based search engines, which returns, as a result, a huge list of publications, that usually include a large number of irrelevant ones [2].

To tackle this problem, research in Text Mining and Information Retrieval has been applied to literature mining in order to help researchers to identify the most relevant publications [1,3].

We have developed a Web-based search tool, BioTextRetriever, to find relevant literature associated with a set of genomic or proteomic sequences. BioTextRetriever uses Text Mining and Machine learning techniques. Machine Learning techniques are used to automatically train a classifier that learns with the papers associated with each set of input sequences. The learned classifier is then used in the process of retrieving relevant papers from a larger repository such as MEDLINE^a.

BioTextRetriever also organizes the papers selected as relevant by the classifier using a ranking function. We have proposed and evaluated a ranking function that combines MeSH terms^b, paper’s number of citations, author’s h-index, journals impact factor, authors number of publications and journal similarity factor^c.

In the rest of this article we first present the related work (Section “State-of-the-art”). In Section “BioTextRetriever architecture” we describe the system’s architecture. The methodology used for the automatic classifier construction process is described in the Section “Methods”. The ranking process is presented in Section “The ranking process”. The proposed ranking function is explained in detail in Section “The ranking function”. The experimental evaluation of the ranking function is described in Section

*Correspondence: talma@fe.up.pt

¹Instituto Superior de Contabilidade e Administração do Porto (ISCAP)– Instituto Politécnico do Porto (IPP) LIACC & Faculdade de Engenharia da Universidade do Porto (FEUP), Portugal

Full list of author information is available at the end of the article

“Choosing the ranking function coefficients”. We discuss the results of such experiments in Section “Results and discussion”. Finally we draw the conclusions in Section “Conclusions”.

Methods

In this section we describe the methodology we used for designing BioTextRetriever.

State-of-the-art

The areas of Information Retrieval, Text Mining and Document ranking are in fact very active research areas. We now reference and comment on work done in those areas that is related to ours.

As text classification is concerned there is a lot of different approaches, including approaches that use Machine Learning. As far as we know there is no previous work that dynamically constructed a text. The main reason is that existing approaches (as is usual in classification problems) require the instances to be pre-classified by an oracle. In our application, when the system runs, we have no access to an oracle to pre-classify the instances. We have taken advantage of the fact that there are a few papers associated to the biological sequences stored in NCBI. We take those papers to be the relevant papers and automatically collect the instances for the alternative class (irrelevant papers) and, in this way assemble automatically a data set.

In [4] the authors use machine learning to order documents by popularity, or the predicted frequency that an article is viewed by the average PubMed user. The authors claim that the identified method for learning popularity from clicking through data shows that the topic of an article influences its popularity more than its publication date. Opposite to our approach [4] method relies on available measures of popularity collected during the use of the system. As seen below our approach relies only on information that is naturally part of the NCBI data bases and not the result of user interactions.

Deng et al. 2012 [5] proposes a unified model, PAV, for ranking heterogeneous objects, such as papers, authors, and venues. PAV explores object ranking in bibliographic information where objects are papers, authors and venues. In PAV the bibliographic information network is represented by a weighted directed graph, where a vertex stands for an object, an edge stands for the link between objects, and a weight over an edge stands for the degree of contribution that one object devotes to the importance or reputation of the corresponding object sharing the same edge with the object. The rank (importance or reputation) of an object is the probability that the corresponding vertex is accessed by random walk in the PAV graph. The authors claim PAV is an efficient solution for ranking author, paper, and venues simultaneously. According to their method, the importance or reputation of an author

is influenced by his co-authors, his papers, and the venues that published his papers. The importance or reputation of a paper is influenced by its authors, its venue, and the papers that cited it. The importance or reputation of a venue is influenced by the papers that it published and the authors who had papers published by the venue. PAV model transforms the problem of ranking objects into the problem of estimating probability parameters. For estimating probabilities the authors developed an algorithm based on matrix computing. The authors claim their algorithm could be ran efficiently by proving that the underlying computing method is convergent.

The authors in [6] present an approach that jointly ranks publications, authors and venues. They first constructed a heterogeneous academic network which is composed of publications, authors and venues. A random walk over the network was performed hence yielding a global ranking result of the objects on the network. The mutual reinforcing relationship between user expertise and publication quality was based on users bookmarks. The authors claim that their experimental results with ACM data set show that their work outperforms all other baseline algorithms, such as Citation Count, PageRank, and PopRank.

In this paper [7], the authors present three different prestige score (ranking) functions for the context-based environment, namely, citation-based, text-based, and pattern-based score functions. Using biomedical publications as the test case and Gene Ontology as the context hierarchy, the authors have evaluated the proposed ranking functions in terms of their accuracy and separability. They concluded that text-based and pattern-based score functions yield better accuracy and separability than citation-based score functions.

The paper [8] proposes an iterative algorithm named AP Rank to quantify the scientists' prestige and the quality of their publications via their inter-relationship on an author paper bipartite network. In this method a paper is expected to be of high quality if it was cited by prestigious scientists, while high-quality papers will, in turn, raise their authors' prestige. AP rank weighs the prestige of quoters more than the number of citations. Given that old papers will have more chances to accumulate more citations than recent works the authors proposed a time-dependent AP rank (TAP rank). According to the authors the main advantages of AP rank are that it is parameter-free; it considers the interaction between the prestige of scientists and the quality of their publications and it is effective in distinguishing between prestige and popularity.

The authors in [9] determine whether algorithms developed for the World Wide Web can be applied to the biomedical literature in order to identify articles that are relevant for surgical oncology literature. For this study the authors have made a direct comparison of

eight algorithms: simple PubMed queries, clinical queries (sensitive and specific versions), vector cosine comparison, citation count, journal impact factor, PageRank, and machine learning based on polynomial support vector machines. As a result of this study they concluded that the mentioned algorithms can be applied to biomedical information retrieval and that citation-based algorithms were more effective than non citation-based algorithms at identifying important articles. The most effective strategies were simple citation count and PageRank and citation-based algorithms can help identify important articles within large sets of relevant results.

In [10] the authors propose a ranking function for the MEDLINE citations. This function integrates the Citation Count Per Year and the Journal Impact Factor which are two of the factors that integrate the ranking function we have developed. The goal of this work is to present to the users a reduced set of relevant citations, retrieved and organized from the MEDLINE citations into different topical groups and prioritized important citations in each group.

The referred work uses graphs, existing web-based algorithms, and some propose a more specific ranking function. We may conclude that to choose an existing ranking algorithm or to develop a new ranking function depends on the work to be applied and on what the researchers want to achieve. In our case we decided to develop a multi-criteria ranking function in order to satisfy all the issues we believe to perform better for ranking the MEDLINE papers.

OKAPI BM25 [11] uses a bag-of-words retrieval function that ranks a set of documents based on the query terms appearing in each document, regardless of the inter-relationship between the query terms within a document.

Cosine Ranking [12] finds the K docs in the collection “nearest” to the query $\geq K$ largest query-doc cosines. It allows the documents to be arranged according to their relation with the query.

In both [11,12] a direct comparison is made with the query made by the user. In our case there is no query to compare with, we have a set of relevant documents (associated with the similar biological sequences). With the approach we have taken we can generalize the common attributes of those relevant papers and use them (in the form of a classifier) to search for other relevant papers.

BioTextRetriever architecture

BioTextRetriever, accepts a set of genomic or proteomic sequences and returns an ordered set of scientific papers reporting work considered relevant for the study of the provided sequences. Figure 1 shows the overall processing implemented by the tool. In this paper we describe briefly the tool’s architecture and will focus mainly on step 6 (*rank paper list* in Figure 1), i.e., to order by relevance the papers returned by BioTextRetriever.

The sequences, provided by the user, are used as seeds to fetch similar sequences in the NCBI web site. This is the first task (step 1) performed by the tool. Along with the similar sequences, the NCBI web site stores a set of paper references associated with the sequences. In step 2, the references of the papers associated with the similar sequences are retrieved. For each of those papers the following information is retrieved from MEDLINE: PubMed unique identifier (pmid), journal title, journal ISSN, article title, abstract, list of authors, list of keywords, list of MeSH terms and publication date. Considering the scope of this research work we take into account paper references that have an abstract available in MEDLINE.

We take this initial set of relevant papers as the “positive examples” and then we add an equal number of “negative examples”. We have done a set of experiments to determine a proper way of collecting the negative examples (details in [13]). The negative examples are randomly collected among the MEDLINE papers that have MeSH terms in common with the positive examples. After step 2 we have a “proto data set”. The *proto data*

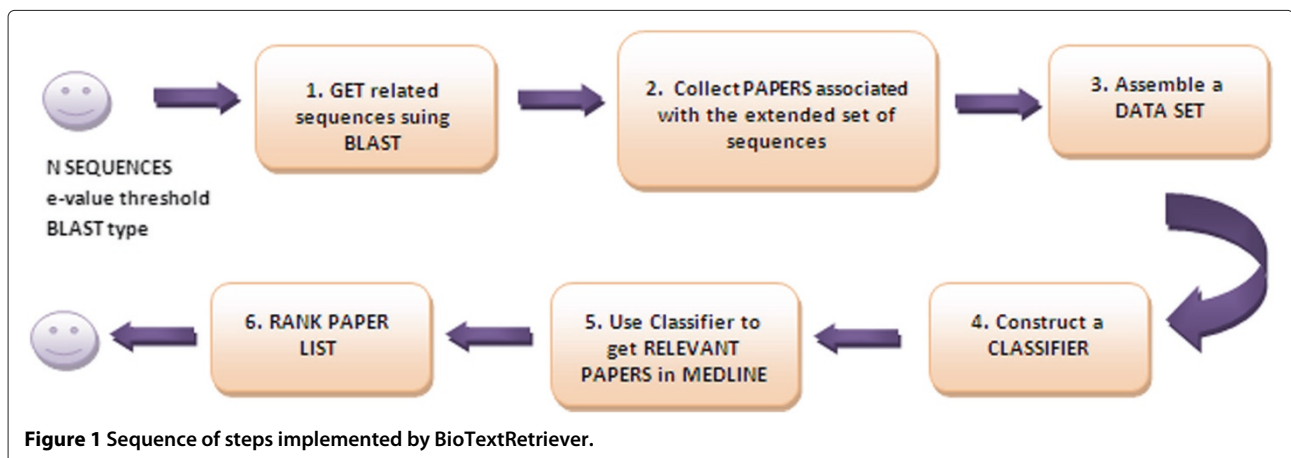


Figure 1 Sequence of steps implemented by BioTextRetriever.

set is subject to Text Mining pre-processing techniques and converted into a data set in step 3. The pre-processing techniques applied are: Handle Synonyms; Stop-words removal; Word validation using a dictionary and Stemming.

Step 4 is one of the most important stages of our work and consists in the dynamic construction of a classifier using Machine Learning techniques.

The resulting classifier is used as a filter to collect an extensive list of relevant articles from the whole MEDLINE (done in step 5). The final list of relevant papers is usually very large. We, therefore, need to order the by relevance. That is the goal of the last task carried out in step 6. For that last task we have developed a ranking function that will be described in detail below.

From sequences to papers

The main part (step 4) of BioTextRetriever's architecture (see Figure 1) is the construction of a classifier which output will be ordered by a ranking procedure, which is the aim of this paper.

The core of BioTextRetriever is the automatic construction of a classifier, acting as a filter, capable of selecting the relevant papers among the whole MEDLINE bibliographic database. "Relevant" papers, in this context, means papers related to the set of sequences that were provided as input to the tool. We now describe the methodology to construct such a classifier as well as a set of experiments that support the choices we have made.

Before we address the classifier construction issue, we must address the construction of the data set (step 3). For most of the learning algorithms we must provide positive and negative examples, in our case both relevant and irrelevant papers.

In [13] we have empirically evaluated three different ways of obtaining the irrelevant papers, which we named

as: Near-Miss Values (NMV), MeSH Random Values (MRV), and Random Values (RV).

To understand the process of selecting irrelevant papers we refer to Figure 2. The relevant papers are the ones associated with the sequences with an e-value^d lower than the e-value cutoff provided by the user (ev). The left box represents the similar sequences that will be used to identify the relevant papers. To obtain the Near-Miss Values (NMV) we collect the papers associated with the similar sequences that have e-value above and far apart from the first threshold (ev) but close to the second threshold (β - a constant of the system determined experimentally).

To make the distinction between relevant and irrelevant papers more clear we have established a "no man's land" zone^e. This "no man's land" zone is represented in Figure 2 by the gray region. The papers associated with the sequences in this gray region are discarded. In Figure 2 the box on the right represents the "not so near" sequences that provide the "near-misses" papers.

The [13] study suggested that generating randomly the negative examples produces better results.

Classifier construction process

Once we have decided how to collect the relevant and irrelevant papers we can now address the question of how to automatically construct a classifier to filter relevant papers in MEDLINE. To address this problem we have considered to combine different partitions of the original data set with different ways of using the Machine Learning algorithms (either isolated or in an ensemble of classifiers). Since the classifier is constructed dynamically and BioTextRetriever is an online tool we were interested in an approach that would be efficient and accurate. We have measured the accuracy and speed of the individual algorithms and the ensembles. We were looking for a possible trade-off between accuracy for speed in the case of close

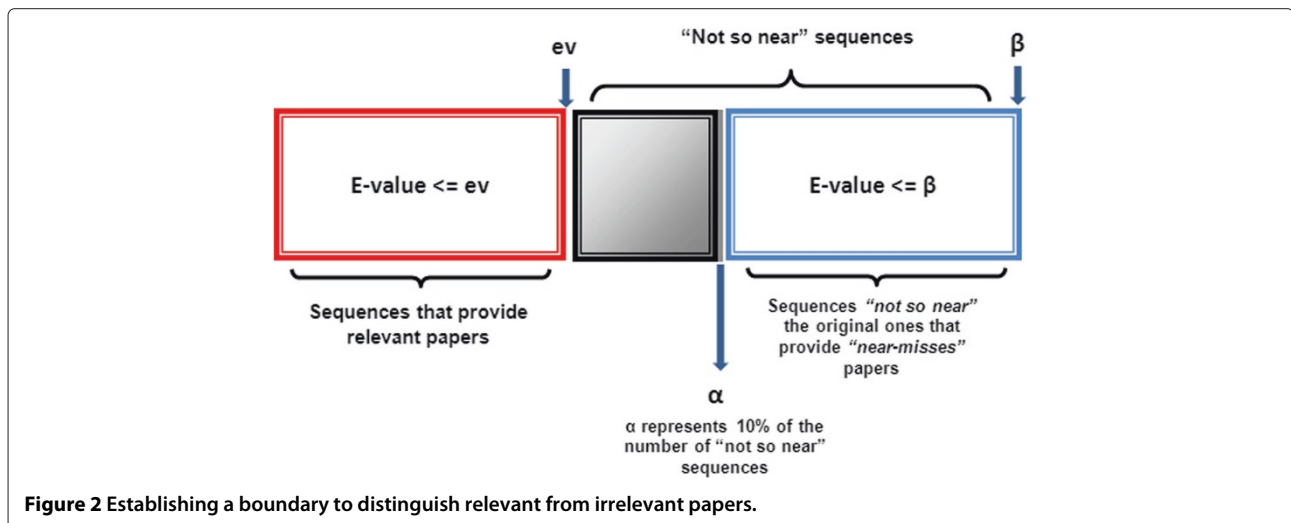
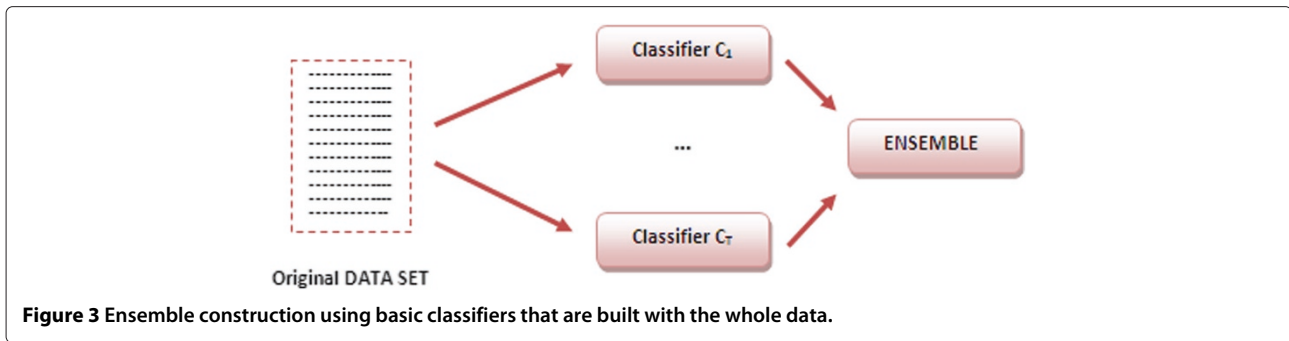


Figure 2 Establishing a boundary to distinguish relevant from irrelevant papers.



values in accuracy. That was not the case and therefore we have adopted the ensemble model (although slower). Details of this study may be found in [14]. We will now detail and explain the best alternative that resulted from this study (shown in Figure 3). Individual classifiers may not perform well on some domains. By combining the results of several individual classifiers the possible specificity of some classifiers may be attenuated by the performance of others. This lead us to include in the ensemble several models of classifiers that include Decision Trees, IBL, Bayesian classifiers and SVMs.

In the alternative adopted we have used the whole data and the use of an ensemble of T basic classifiers (C_i) when each basic classifier uses the whole data. For the Ensemble we have evaluated three well known algorithms: AdaBoost, Bagging and Ensemble Selection. Different “ensemble parameters” were tested in the experimental evaluation of this alternative is described in detail^f.

Data characterization

The data sets used in our experiments are characterized in Table 1. The name of the data sets reflect the domain of the sequence associated. The data sets used for the

experiments have more than 150 (positive and negative) examples.

Evaluating the results

As expected through the literature [15] ensemble learners have a higher and uniform performance than base learners.

The ensemble is made using the WEKA’s ensemble classifiers Bagging, AdaBoost and Ensemble Selection.

For the Bagging and AdaBoost classifiers we need to specify a base learner. For each base classifier and data set we have used the parameter combinations that achieved the best results. We have used and developed a wrapper for the ensembles that automatically tunes ensemble-level parameters. The Ensemble Selection algorithm allows us to specify the set of base learners as well as the best options for each individual learner. Table 2 shows the results obtained.

As a global result we can see that all algorithms have in general very good performance, well above the majority class predictor (the ZeroR result are around 50%). The three ensemble learners used (Bagging, AdaBoost and Ensemble Selection) according to the t-student test

Table 1 Characterization of data sets used to assess the Ensemble algorithms (AdaBoost, Bagging and Ensemble Selection)

Data Sets	Number of Attributes	Positive Examples	Negative Examples	Total Examples
S12	1602	128	128	256
H11	1461	120	120	240
ERYT21	1592	118	118	236
HYP11	1706	130	130	260
HYP21	1944	194	194	388
BG11	1546	97	97	194
BG21	1631	115	115	230
BG31	1859	149	149	298
LUNG21	1535	120	120	240

Table 2 Ensemble’s Accuracy results

Data Sets	AdaBoost	Bagging	Ensemble Selection
S12	99.2 (1.7)	98.1 (3.3)	97.3 (3.7)
H11	99.2 (1.8)	99.2 (1.8)	96.3 (5.0)
ERYT21RV	95.4 (4.2)	95.8 (3.9)	93.3 (4.5)
HYP11	91.2 (4.1)	90.0 (6.3)	91.2 (4.1)
HYP21	95.1 (1.9)	95.1 (3.3)	89.7 (6.5)
BG11	93.8 (3.4)	94.3 (4.6)	94.3 (5.1)
BG21	95.7 (3.6)	94.4 (4.6)	93.9 (5.1)
BG31	93.9 (3.2)	91.6 (4.8)	92.3 (5.0)
LUNG21	93.8 (4.1)	92.5 (4.7)	92.5 (4.3)
Overall Average	95.3 (3.1)	94.6 (4.1)	93.4 (4.8)

The bold values are statistically different from the second best values according to the t-student test ($\alpha = 0.05$).

($\alpha = 0.05$) have no statistically significant difference. Thus we conclude that either one can be chosen.

The ranking process

In order to understand the procedure involved in the last step of the tool we will use Figure 4 to provide a context.

We will first present a summary of the tool's steps up to this point. The user provides a set of sequences, to which the NCBI BLAST tool [16] associates a set of similar sequences. Each of these similar sequences has in turn a set of papers associated. We collect all the papers associated to these similar sequences to build the "relevant papers" part of a data set. The data set is completed by adding a set of "irrelevant papers", through the Random Value method described in Section 'BioTextRetriever architecture'. With this data set as input to a machine learning algorithm a classifier is constructed. This classifier will be used, later in the process, to filter the relevant/irrelevant papers from MEDLINE.

From the set of relevant papers associated to the input sequences we extract a set of MeSH terms appearing in these papers. We then search the MEDLINE database for the papers that have the MeSH terms in common with

the set's. Since the whole MEDLINE has a huge number of papers, we use the MeSH terms as a filter to reduce the initial volume of data to a reduced potentially relevant number of papers. Thus we create a MEDLINE sample with papers for classification, that are, somehow related to the introduced sequences because they possess the highest number of MeSH terms in common with the relevant papers we have so far. With this we construct a data set of papers to be given to the classifier for classification. After classification we get a set of papers classified as relevant. To this data set of relevant papers we add a random set of 50 papers, extracted from the set of papers associated to the similar sequences, associated themselves in the first step with the initial sequences entered by the biologist. This procedure was only performed in the experiments intending to decide on the components of the ranking function. The procedure is not performed in the current use of the tool.

To make the tool efficient we have to address some implementation issues. In the 20 million references to scientific papers of MEDLINE only 9 million have abstracts. To apply our classifier to 9 million abstracts is unfeasible in an acceptable time. Besides we want references to

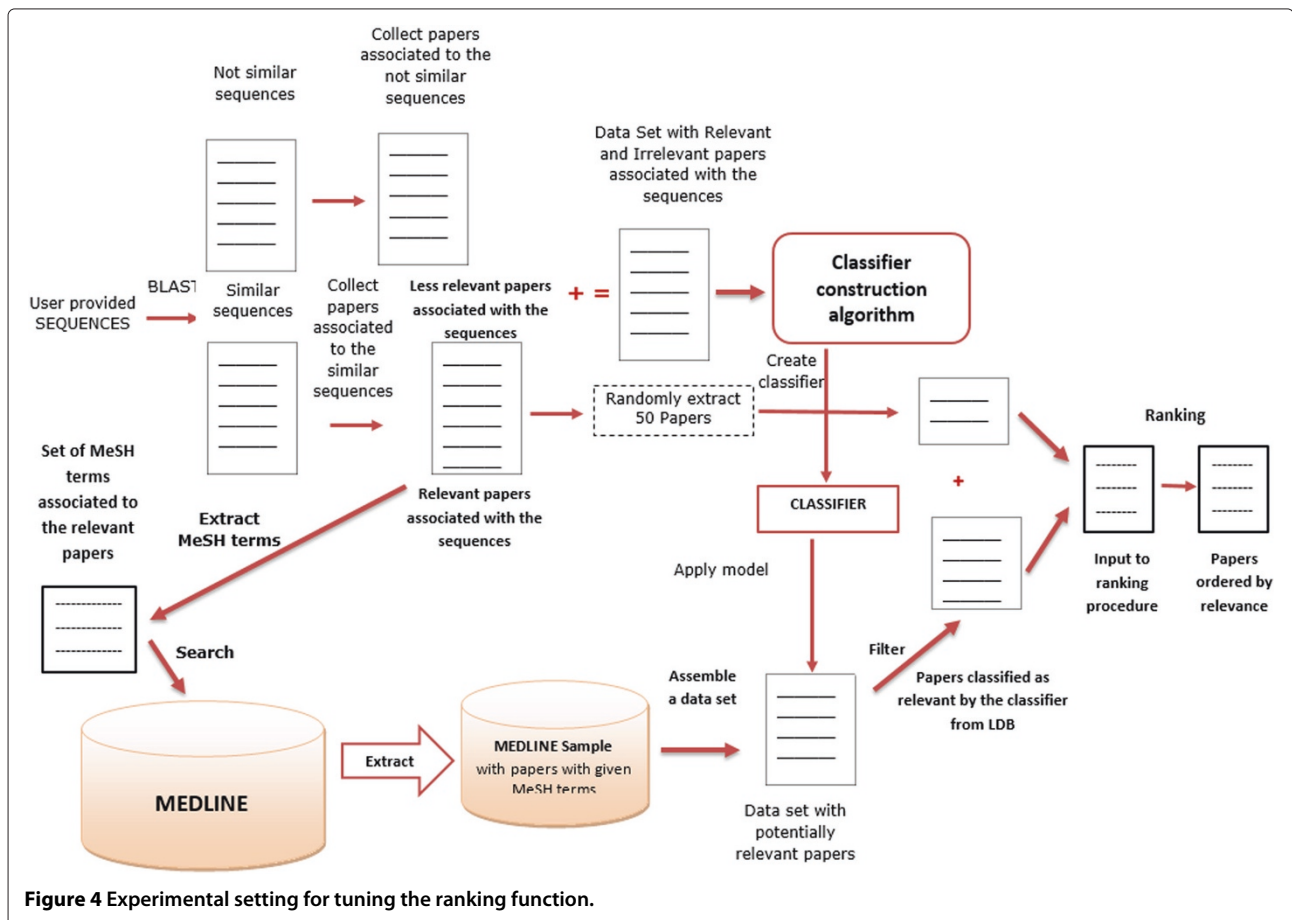


Figure 4 Experimental setting for tuning the ranking function.

papers that are related to the input sequences. One way to do this is to make an a priori selection of these papers based on specific criteria. The chosen criteria is to select the papers that have the highest number of MeSH terms in common with those extracted from the relevant examples associated to the input sequence. Figure 5 shows this procedure. In this way we assure that the set of papers that will be classified by BioTextRetriever are somehow related to the input sequences. The number of papers included in the MEDLINE sample was also subject to experimental testing because it affects the efficiency of the BioTextRetriever. We have made an initial attempt to select 20000 papers but the SQL instruction to collect the papers was very time consuming. So we reduced this subset to a 5000 papers. The 5000 subset is created in an acceptable time and we believe that 5000 papers are more than enough to be classified by our model and to be presented to the user in the end. The SQL instruction filters the papers with highest number of MeSH terms that will be further filtered by the classifier. The set of paper classified as relevant by the classifier are further subject to a ranking process.

The ranking function

Despite the potential relevance of the papers returned by the BioTextRetriever, we need to point out which are the most important papers to present to the user first. A ranking is an ordering of the documents that should reflect their relevance to a user query [12].

The traditional methods for ranking web pages are not suitable to rank scientific articles, since the traditional ranking algorithms (PageRank, Hits, Salsa and Ranknet to name a few) are based on the number of links to a web page. Besides this reason, the existent algorithms do not take into account the items we believe are the most

important to consider in the ranking such as the MeSH terms, the number of publications of the authors, the number of citations of a paper, the h-index of the author of a paper, the journal impact factor and the Journal Similarity Factor. None of the mentioned algorithms involves a function that contemplates the mentioned items. Thus, we have proposed a function that reflects the specific criteria we believe to be the best to use in this case. We propose an integrated ranking of MeSH terms, PubMed number of citations, author PubMed h-index, journals impact factor, authors' number of PubMed publications and the journal similarity factor where the relevant papers were published. The combined use of several indicators that give information on different aspects of scientific output is generally recommended [17].

As explained in the previous section we have used a sample of MEDLINE with papers that have the highest number of common MeSH terms, that we believe to be the best one to use with the relevant papers associated with the introduced sequence(s).

After classification the resulting^g paper references are ordered by the following ranking function:

$$C1 * MeSH + C2 * Citations + C3 * hindex + C4 * IFactor + C5 * Pub + C6 * JSFactor \text{ where:}$$

- *MeSH* is a weighted sum of MeSH terms in common with the papers associated with the introduced sequence(s);
- *Citations* is the paper's number of citations in MEDLINE;
- *hindex* is the highest h-index among the authors' h-index;
- *IFactor* is the Journal Impact Factor;
- *Pub* is the number of publications of the author with the highest number among them all;

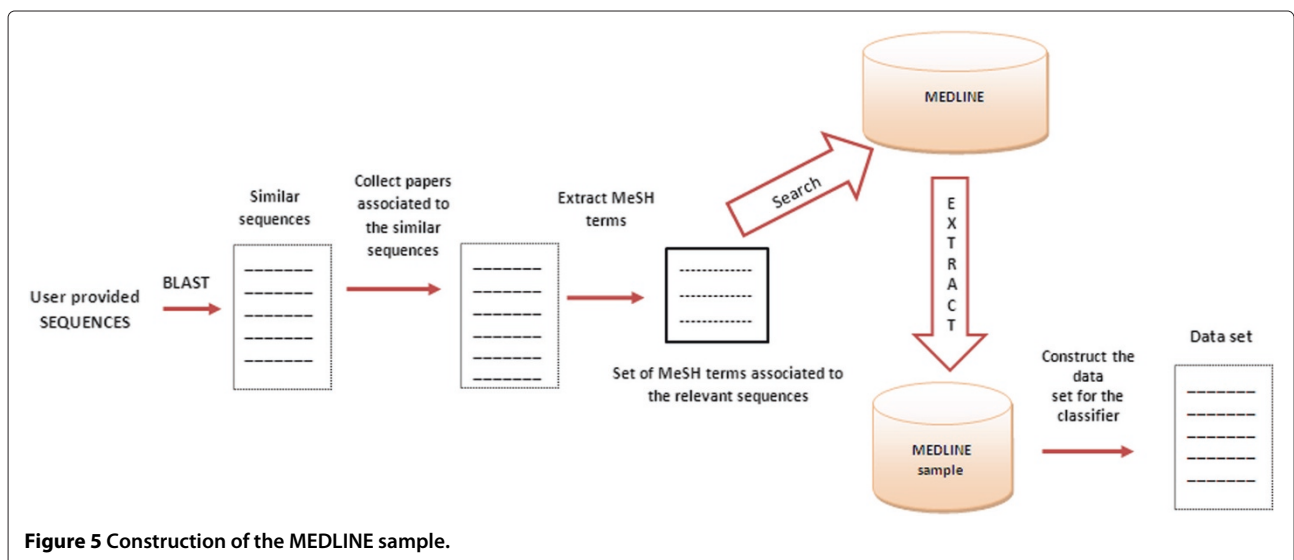


Figure 5 Construction of the MEDLINE sample.

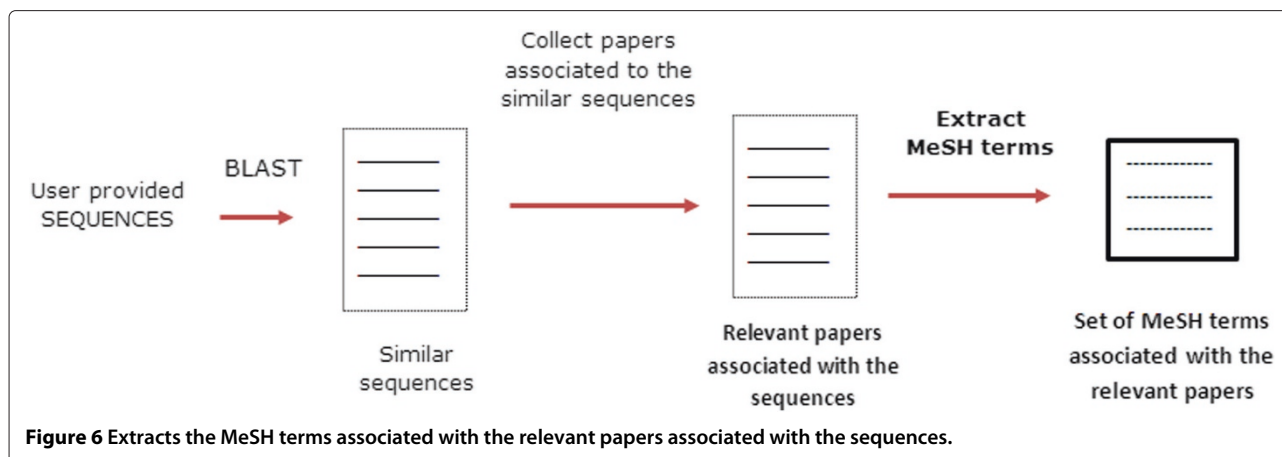


Figure 6 Extracts the MeSH terms associated with the relevant papers associated with the sequences.

- *JSFactor* is a weighted sum of the number of papers published in a journal that has papers associated to the introduced sequence(s).

Coefficients C1, C2, C3, C4, C5 and C6 may vary between 0 and 100 and their sum must be 100. The set of experiments to determine the values of these coefficients is described in detail in Section ‘Experimental settings’.

Besides the information contained in MEDLINE, we have added some extra information to the Local Data Base (LDB). Besides the Journal Impact Factor all the terms in the ranking function (number of MeSH terms, number of citations, author h-index, author number of publications and the Journal Similarity Factor) are computed using LDB. The Journal Impact Factor was obtained from the ISI Web of Knowledge website powered by Thomson Reuters. We have normalized the coefficient factors between 0 and 100 to obtain a coherent formula.

The paper references are ordered by the ranking function and presented in decreasing order of relevance to the user. Although it is impossible for a human to make an exhaustive reading of all the presented papers, the user can access and see all the papers returned in decreasing order of relevance. The tool presents 30 results per page.

We will now detail each of the terms that integrate the ranking function.

Number of MeSH terms

BioTextRetriever collects all papers related to the relevant sequences. The MeSH terms of those papers are extracted as explained in Figure 6. After constructing a classification model, based on this set of papers, the classifier applies the model to a MEDLINE sample. This sample is composed by the papers that have the most common MeSH terms with the MeSH terms of the relevant papers.

With this procedure we guarantee that the MEDLINE sample has papers that have a higher number of common MeSH terms with the sequence related papers, and is taken as the first step to a collection of relevant papers.

In order to better highlight even more the number of MeSH terms in common with the ones associated with the sequences, we have introduced a ponderation for the papers that have more of these MeSH terms in common. Table 3 shows an example of four papers associated with the input sequences. Suppose that we have a paper in MEDLINE that has MeSH2, MeSH3 and MeSH4. The ponderation factor for this paper should be equal to $3 + 4 + 3$, which equals 10. If we now have a paper in MEDLINE with MeSH1, MeSH7 and MeSH8 its weight would be 2 reflecting the potential weak “connection” with the relevant papers.

This way the papers that have more common MeSH terms associated to the sequences are valued.

Author’s number of publications

The number of publications of an author has also been considered in the formula. However it may happen that an author may have a large number of publications but with few citations, and these citations are in journals with a small impact factor, whilst another author may have a smaller number of publications with a high number of citations published in journals with high impact factors, which should be more relevant in the ranking formula. For each author we count the number of publications available. The authors with more than fifty publications get a number of publications of fifty. We believe that fifty is a good number of publications for a good author.

Table 3 The table shows an example of four papers associated with the input sequences

Papers	MeSH1	MeSH2	MeSH3	MeSH4
Paper1	0	1	1	1
Paper2	1	1	1	1
Paper3	0	1	1	0
Paper4	1	0	1	1
Total	2	3	4	3

The authors with a lower number of publications are not immediately discarded but are ranked lower. However the most common case is paper's with more than one author. In this case, we consider the highest number of publications. As we do not disambiguate the author's name, there is a slight chance that authors with the same name induce an error in the effective number of publications of a particular author.

Number of citations

A citation is a unique reference to an article, a book, a Web page, a technical report, a thesis or other published item. The number of citations of a paper has become a major indicator to evaluate scientific work. Although it has some drawbacks and it is not unanimously accepted by the scientific community, nevertheless we consider that it is one of the most important measures to estimate the impact of scientific published work in the scientific community. Citation indexes provide a means with which to measure the relative impact of articles in a collection of scientific literature [18]. The concept of citation indexing and searching was invented by Eugene Garfield [19] and anticipated the Science Citation Index. There are several citation indexing systems such as Google Scholar, CiteSeer and Scopus. These systems allow to search for a researcher's number of citations however, we could not use them to obtain the number of citations of around 20 million MEDLINE publications due to PubMed service restriction policies.

We have computed, using LDB, the number of citations of MEDLINE scientific papers. Each sequence has a set of paper references associated, and each of these references has the bibliography associated. Most of the referenced papers are available in MEDLINE so we can obtain the number of citations of a paper cited by other MEDLINE papers inside MEDLINE.

The number of citations for a particular paper is shown to be more relevant and important in comparison with the number of publications. This is because an author may have a higher number of publications but that are not cited, whilst another author may have a smaller number of publications but highly cited. The impact of a piece of research is the degree to which it has been useful to other researchers [20]. However, the number of citations does not take into account the distribution of the several publications, e.g., a high number of citations but with very few highly cited scientific papers. However, we could not use them to obtain around 20 million citations for the MEDLINE publications due to PubMed service restriction policies.

Papers that have a high number of citations but that are not recent, e.g., should be devalued when compared with recent papers with a high number of citations. In fact, recent papers have naturally less citations than older

papers. We have implemented the following formula for the number of citations.

$$Number\ of\ citations = \frac{Effective\ Number\ of\ citations}{\alpha * Number\ of\ years}$$

Where α represents the devaluation coefficient used, specified in Table 4.

For example, the paper "The sequence of the human genome", a well known paper in the Biological and Medical communities, has a score of 100 for the item "number of citations" in MEDLINE, and in Google it has 10240 citations (searched in 18.December.2012). As we are only considering counting in LDB, on one hand we have found less papers inside MEDLINE that cite the scientific paper "The sequence of the human genome" than in Google, besides BioTextRetriever devalues the number of citations by the paper scientific age, thus the number of citations in MEDLINE is much lower than the one found by Google.

h-index

Hirsch [21] proposed the *h-index*: "A scientist has an index h if h of his or her N papers have at least h citations each, and the other $(N - h)$ papers have less than or equal to h citations each". In other words, the *h-index* bases itself on publications ranked in descending order according to their number of citations. *h-index* is approximately proportional to the square root of the total citation counts [22].

The *h-index* is an index that attempts to measure both the productivity and impact of the published work of a researcher. It is based on the set of the scientist's most cited papers and the number of citations that they have received in other publications. It combines both the number of papers and their quality (impact, or citations to these papers) [23]. The *h-index* is recognized by the ISI Web of Science by Thomson Reuters or Scopus by Elsevier, as an important indicator for assessing research impact [21,24,25].

Like the other bibliometric measures, *h-index* has advantages and limitations. Mathematically it is very simple to compute and it is easy to understand [21,23].

Hirsch claims in [21] that the *h-index* performs better than other single-number criteria commonly used to evaluate the scientific output of a researcher (impact

Table 4 The α value represents the devaluation coefficient in decreasing order of the paper's scientific age

Age of Papers	α
≤ 5 years	1.0
≥ 5 years and ≤ 20 years	0.8
≥ 20 years	0.6

factor, total number of documents, total number of citations, citations per paper rate and number of highly cited papers).

For young researchers the *h-index* is not a very promising measure since they have few publications highly cited and thus will probably have a low *h-index*. One might say that the *h-index* favors the researchers that have many cited publications. A scientist with very few highly cited papers or a scientist with many lowly cited papers will have a weak *h-index* [24], [26]. To address this issue, Hirsch presented the “*m* parameter” in [21] that divides *h* by the scientific age of a scientist (number of years since the author’s first publication) to attenuate this problem.

Besides this, the *h-index* also depends on the database in use, reason which, alongside problems with common names and different spellings, makes its flaws very visible [27]. Hirsch [21] also refers this technical problem in obtaining the complete list of publications of scientists with very common names. To overcome this problem, the authors in [28] recommend that the *h-index* should be calculated with a list of publications authorized by the scientist and found in the Web of Science using a combination of the scientist’s name and address or affiliation.

The *h-index* should not be used to compare scientists from different disciplines [21]. The *h-index* does not take care of self-citations which can increase a scientist’s *h-index* [29]. The *h-index* can also be used to measure the scientific output of institutions and research groups [30].

As was already mentioned, we have obtained using LDB, the number of citations of each paper’s reference inside MEDLINE. We collect and store for each paper author the number of publications. For each publication we count the MEDLINE internal number of references to that particular publication to obtain the number of citations.

Table 5 shows an example of how to calculate the *h-index* of an author inside MEDLINE.

The first line indicates the order of each publication in ascendant order. The second line presents the number of citations in descending order. The author’s first publication has 1988 citations, the second publication has 8 citations, and so on. We know from the literature that the *h-index* of an author is *h* when the number of citations is equal or greater than the number of publications. A researcher has *h-index* *h* if, in the list of articles arranged in decreasing order of the number of citations of these articles, *r=h* is the highest rank such that the papers on rank 1, 2,..., *h* each have at least *h* citations [31]. Thus in

Table 5 Example of *h-index* computation for *h=4* in this case

Rank of publications	1	2	3	4	5
Number of citations	1988	8	7	6	4

the presented example the *h-index* is 4, because the author has four papers with more than four citations each.

As a paper may have more than one author (which is the most common case) we calculate the *h-index* for all the authors of a paper and select the highest *h-index*. If an author has a high *h-index* and the other authors have a smaller *h-index*, it means that at least one author is recognized by the scientific community as having prestige, and a prestigious author has valuable publications.

Journal impact factor

For the ranking we have considered only the journal Impact Factor because MEDLINE only references papers that are published in Journals. The Journal Impact Factor (JIF) [32] is a measure of the frequency with which the average article in a journal has been cited in a particular year or period, thus JIF may change overtime. JIF is based on information obtained from citation indexes. The most widely accepted and used JIF is from the Journal Citation Report (JCR), a product of Thomson Reuters ISI (Institute for Scientific Information) (only considers ISI journals). The Journal Citation Report has been published annually since 1975.

Garfield developed the journal’s impact factor metric that is defined by the following formula: $JIF = \frac{C_2}{P_2}$, where C_2 is the number of citations in the current year of any of the items published in a journal in the previous 2 years and P_2 is the number of papers published in the previous 2 years.

For example, the 2012 JIF is calculated by the formula: $\frac{C_2}{P_2}$, where: C_2 is the number of times papers or other items published during 2010-2011 were cited in indexed journals during 2012, and P_2 is the number of items published in 2010 plus 2011.

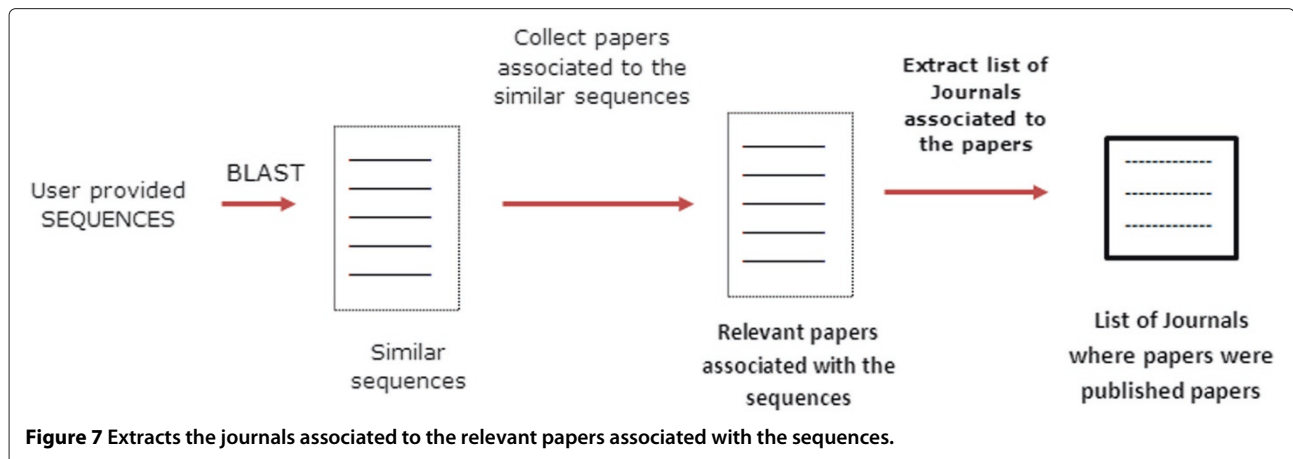
Thomson Reuters released in 2009, the new 5-year journal Impact Factor in addition to the standard 2-year journal Impact Factor. The 5-year journal Impact Factor is the average number of times articles from a journal published in the past five years have been cited in Journal Citation Report year. And it is calculated by dividing the number of citations in the Journal Citation Report year by the total number of articles published in the five previous years.

The Journal Impact Factor is used to compare different journals only within the same field. The ISI Web of Knowledge indexes more than 11,000 science and social science journals.

A journal with a high impact factor is usually considered a high quality journal and high quality journals usually have high quality papers.

The Journal Impact Factor has some limitations stated by [33-35]:

- Journal Impact Factor does not control self-citations;
- Journal Impact Factor varies significantly from field to field;



- The Journal Impact factor depends on the dynamics of the research field;
- Journals databases are not always accessible, i.e., neither all papers are available for free in the Web;
- High citation rates do not always reflect the high quality of a journal/paper;
- The Journal Impact Factor is calculated over a short period of time (the last two or five years);
- A citation in a “low impact” journal is counted equally to a citation in a “high impact” journal, however they should be distinguished, since the second one it is more valuable than the first one;
- A journal score is highly influenced by its total number of citable papers;
- Journal Impact Factor does not assess the quality of individual papers (only a small percentage of Journal papers are highly cited but they have a huge impact in the total number of citations of a Journal).

[34,35] also enumerate the determining factors associated with journals with high impact factors:

- Indexing in most known databases: PubMed/MEDLINE, Scopus and Google Scholar;
- Papers written in the English language;
- Availability of the full-text paper, preferentially for free;
- Availability of the paper abstract;
- Submissions from authors with an higher reputation;
- Publications of an higher number of review papers, because review papers are often more cited;
- To cite papers previously published in the same journal;
- Focus on dynamic “excellence” research fields that generate more citations.

Although the Journal Impact Factor has the above mentioned limitations we included it in the ranking formula.

We have obtained the 2-year Journal Impact Factor for the papers that have been published in the Web of Knowledge website. We have downloaded the complete list of Journals Impact Factors available in October 2010^h. For each paper references BioTextRetriever retrieves, we gather the Journal Impact Factor of the publication (Thomson Reuters).

Journal similarity factor

The Journal Similarity Factor, highlights the journals with more papers published that are associated to the sequences introduced by the user. A paper can be published in one and only one Journal. The key idea is that the papers that are associated to the sequences introduced by the user should have a higher impact in the formula. Figure 7 illustrates this procedure.

Table 6 shows an example of four papers from the set of papers associated with the input sequences. Suppose that a paper we collect from MEDLINE, was published in Journal 1. As this particular journal has published three papers associated with the sequences, the formula should emphasize this fact by assigning the weight 2 to this factor in detriment of a journal that has, for example zero papers published.

Some of the items in the ranking formula, namely the number of MeSH terms in common with the papers

Table 6 Example of four publication journals of four papers associated to the input sequences

Papers	J1	J2	J3	J4
Paper1	1	0	0	0
Paper2	0	1	0	0
Paper3	0	0	1	0
Paper4	1	0	0	0
Total	2	1	1	0

J1 stands for Journal1, J2 for Journal and so on.

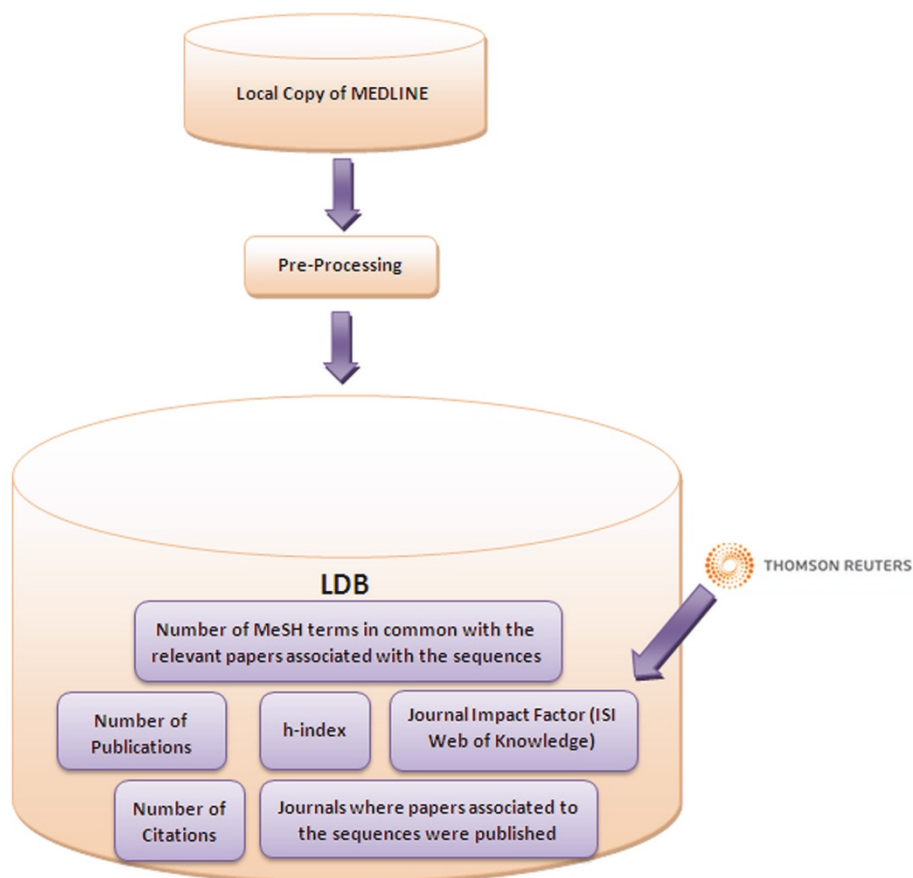


Figure 8 Information items stored in the LDB to be used in the ranking function. The Journal Impact Factor is the only one not calculated through the local copy of MEDLINE's available information, but is downloaded from the Web of Knowledge website (Thomson Reuters) and is saved in LDB.

associated with the sequences, the author's h-index, the number of publications, the number of citations and the Journal Similarity Factor are calculated and stored jointly to the authors. The number of citations and the number of publications are independent of the other items. However the h-index relates the number of publications and the number of citations.

Figure 8 summarizes the six issues mentioned that are part of the ranking function developed. The Journal Impact Factor is an item that is not calculated through the LDB but is obtained through an external source (Thomson Reuters).

Choosing the ranking function coefficients

As described in Section 'From sequences to papers', the ranking function combines the following six components:

1. The number of MeSH terms associated with the papers connected to the sequences introduced by the user;
2. Number of PubMed publications;
3. Number of citations;

4. Author h-index;
5. Journal Impact Factor;
6. Journal Similarity Factor.

In order to assure the usefulness of these coefficients to the relevance of the retrieved papers and also to propose default values for the formula coefficients, we undertook a set of experiments that are next described.

Experimental settings

Data description

We have used 14 data sets, each one composed by more than 90 relevant papers. These data sets resulted from using sequences from 7 different domains with the following distribution:

- Rnases: 1 sequence
- Alzheimer: 1 sequence
- Blood Pressure: 1 sequence
- Erythrocytes: 2 sequences
- Hypertension: 2 sequences
- Blood Glucose: 4 sequences
- Lung Disease: 3 sequences

The data sets used are characterized in Table 7 concerning the number of attributes, and in Table 8 BioTextRetriever as relevant. Table 8 show high variability in the number of relevant papers collected in MEDLINE. One possible reason is that some biological problems are more popular than others. In that case it is likely that we will find more papers from the more wide studied domains. It is also frequent that the sequences used (and the papers related to them) are a new/recent research field and in that case there will be unlikely for find a lot of available papers in MEDLINE The papers from which the classifier selects the relevant ones is constructed using MeSH terms common to the “similar sequence’s papers. Some of the MeSH term may be general enough to include quite diverse types of papers. In that case the relevant papers would be a small percentage of the set.

In [14] we showed, empirically, that the best alternative was to use the Ensemble algorithms (Alternative 2) for the classification problem. Consequently we have used the results provided from this alternative in the experiments.

Experimental procedure

Since we do not have access to an expert to evaluate the results of the application of the ranking function, and also because the sorted set of paper is still very large we have adopted the following procedure. For each data set we performed the following actions:

1. Run step 1 through step 5 of the tool to get a set of potentially relevant papers;

2. Add to the extracted set of papers classified as relevant in the previous step of the tool, 50 papers extracted randomly from the relevant papers associated to the input sequences. Since these papers are guaranteed to be relevant (by the owners of the original sequences) we use them to alternate the fact of not having access to an expert.
3. Count how many of the guaranteed relevant papers (obtained in 2.) will appear in high positions of the ranked set.
4. For each data set of the Table 7:
 - (a) Create 10 new sub data sets, each of them with 50 randomly examples added from the relevant one’s
5. The average of the combinations for each of the 10 sub data sets is obtained and represents the value achieved for each data set.

In these experiments we have tested the five coefficients with values from the set {0, 25, 50, 75, 100} with the restriction that the sum of all coefficients must be 100%. The combination of all these values for the five coefficients gives a total of one hundred and twenty six possible combinations.

The ranking function is evaluated by analyzing the first 20 papers that are presented to the user in descendant order of relevance and counting the number of papers from the 50 relevant ones inserted in the data set that appear in this 20 first.

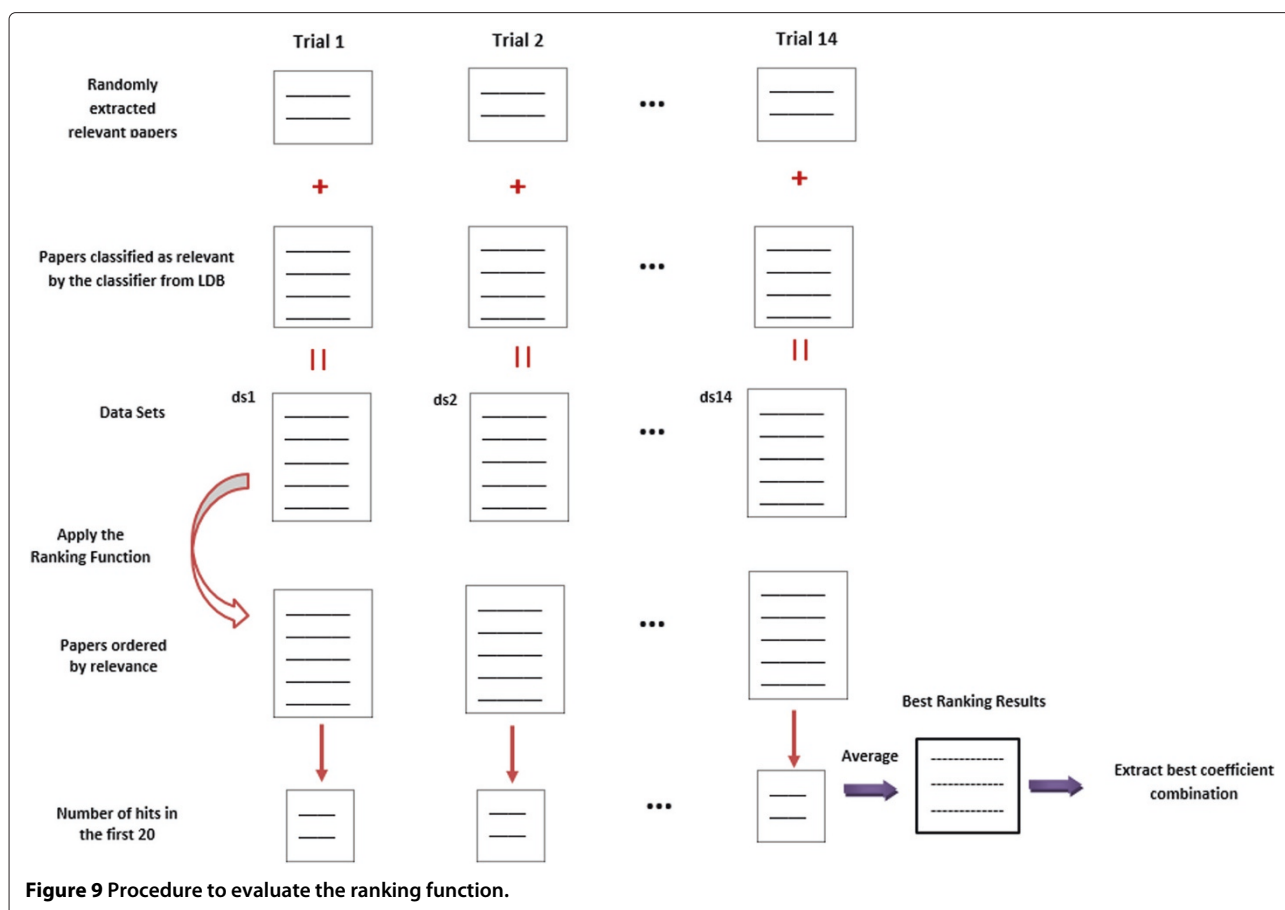
Table 7 Characterization of data sets regarding the number of attributes and the number of positive and negative examples

Data sets	NA	Positive examples	Negative examples	Total examples
S12	1602	128	128	156
BP25	441	63	31	94
ALZ31	1485	114	114	228
ERYT11	1505	99	99	198
ERYT21	1592	118	118	236
HYP11	1706	130	130	260
HYP21	1944	194	194	388
BG11	1546	97	97	194
BG21	1631	115	115	230
BG31	1859	149	149	298
BG41	1812	161	161	322
LUNG11	1553	124	124	248
LUNG21	1535	120	120	240
LUNG31	1054	74	74	148

NA stands for number of attributes.

Table 8 Characterization of data sets used to tune the coefficients of the ranking function

Data sets	Total relevant papers classified by BioTextRetriever	% Relevant papers classified by BioTextRetriever
S12	3947	78.9%
BP25	2071	41.4%
ALZ31	1751	35.0%
ERYT11	2498	50.0%
ERYT21	4397	87.9%
HYP11	4235	84.7%
HYP21	4638	92.8%
BG11	4423	88.5%
BG21	5	0.1%
BG31	2301	46.0%
BG41	3288	65.8%
LUNG11	4103	82.1%
LUNG21	4182	83.6%
LUNG31	1644	32.9%



The combination that returns the higher number of relevant papers associated with the references constitute the best coefficient combinations for the proposed ranking function. Figure 9 summarizes the procedure.

The combination that has more hits in average for all the data sets is considered the best combination for the default ranking formula.

Results and discussion

The columns C1 to C6 of Table 9 represent the best ranking coefficient combinations for the presented methodology.

Table 9 The three best combinations for the fourteen data sets described in Table 7

Combination	C1	C2	C3	C4	C5	C6	Average
comb1	0	75	25	0	0	0	3.8 (3.8)
comb2	0	100	0	0	0	0	3.7 (4.1)
comb3	0	50	50	0	0	0	3.6 (4.1)

The C_i s are the following coefficient weights: C1 - number of MeSH terms; C2 - number of citations; C3 - author h-index; C4 - the impact factor; C5 - number of publications and C6 - Journal Similarity Factor.

Table 10 Individual combination results for the data sets described in Table 7 for the three combinations presented in Table 9

Data set	Comb1	Comb2	Comb3
S12	1.9 (1.2)	1.9 (1.2)	1.6 (1.3)
BP25	3.4 (1.2)	3.4 (1.2)	3.4 (1.2)
ALZ31	3.1 (1.8)	3.3 (1.7)	2.8 (1.7)
ERYT11	1.0 (1.2)	1.3 (1.2)	0.9 (0.9)
ERYT21	4.1 (1.8)	3.8 (1.8)	3.7 (1.9)
HYP11	0.9 (0.9)	0.9 (0.9)	0.7 (0.8)
HYP21	1.9 (1.6)	1.7 (1.2)	1.8 (1.5)
BG11	3.1 (1.3)	3.2 (1.7)	3.6 (1.4)
BG21	17.6 (0.7)	16.4 (0.7)	17.5 (0.9)
BG31	2.8 (1.7)	2.8 (1.7)	2.6 (1.6)
BG41	2.0 (1.3)	2.1 (1.5)	2.0 (1.3)
LUNG11	3.8 (2.2)	3.6 (1.9)	3.7 (2.2)
LUNG21	2.7 (1.6)	2.8 (1.7)	2.1 (1.1)
LUNG31	4.4 (1.2)	4.1 (0.9)	3.7 (0.9)

The columns C1 to C6 of Table 9 represent the values of the six items coefficients: where C1 is the coefficient weight for the number of MeSH terms; C2 is the coefficient weight for the number of citations; C3 is the coefficient weight for the author h-index, C4 is the coefficient weight for the impact factor, C5 is the coefficient weight for the number of publications and C6 is the coefficient weight for the Journal Similarity Factor. The last column represents the hits average for each combination for the fourteen data sets used. Table 10 shows the individual combination results for the fourteen data sets described in Table 9 for each line of this Table.

The best results highlight the number of citations and the h-index factors. We have applied the t-test to analyze these three best results. The t-test ($\alpha = 0.05$) gave no statistical significance between the three best results presented.

From the presented best combinations, BioTextRetriever was configured with the combination presented in the first line of Table 9. Although BioTextRetriever was configured with the aforementioned weights, the user may introduce the weights.

Conclusions

We have developed a new methodology based on Machine Learning techniques to construct a classifier in real time for classifying MEDLINE papers. We have devised and assessed several ways of partitioning the data and combining the Machine Learning algorithms in order to achieve a good performance in the classification process. From this study we were able to conclude that the best Machine Learning algorithms to achieve a good performance are the Ensemble of Classifiers (a method that combines the individual decisions of a set of classifiers through majority or voting). In terms of the accuracy of the results, the Ensemble of algorithms achieved an accuracy of 95.3% and the stand alone classifiers achieved an accuracy of 92.7%. The results show that the use of Machine Learning is extremely valuable to automate the Information Retrieval process with good performance results.

In this paper we have proposed a new methodology that enables the automation of the assessment process of a multi-criteria ranking function.

BioTextRetriever's last procedure is to organize the papers selected as relevant by the classifier. In fact, this set of papers classified as relevant is quite large and it is not advisable to present such a huge number of papers to the user. We proposed an integrated ranking function that combines MeSH terms, PubMed number of citations, author PubMed h-index, journals impact factor, authors number of PubMed publications and journal similarity factorⁱ.

Since we do not have access to an expert to evaluate the results of the ranking function, we have adopted a

procedure where the relevant papers associated to the original sequences are the ones that maximize the presented ranking function if they appear in the first 20 results. Since these papers are guaranteed to be relevant (because they are associated to the original sequences) we use them as an alternative to the fact that we do not have access to an expert. The ranking function is evaluated by analyzing the first 20 papers that are presented to the user in descendant order of relevance by the ranking function, and counting the number of papers from the relevant papers associated with the introduced sequences that maximize the ranking function. The best combinations maximize the number of citations and the h-index. BioTextRetriever was configured, by default, with this coefficients combination, however the user can introduce other weights for each factor.

Endnotes

^aWe have used MEDLINE 2010.

^bThe Medical Subject Headings (MeSH) [36] is a controlled vocabulary thesaurus maintained by the National Library of Medicine (NLM).

^cThe journal similarity factor highlights the journals with more papers published associated with the original sequences.

^de-value is a statistic to estimate the significance of a "match" between two sequences [37].

^eWe have established that this would be 10% of the number of "not similar" sequences associated with the introduced sequence.

^fIn all of the algorithms uses a wrapper was used to find the best algorithm's parameter combination.

^gat most 5000.

^hAt this date there were 7347 journal classifications available.

ⁱThe journal similarity factor highlights the journals with more papers published that are associated to the original sequences.

Competing interests

The authors declare that they have no competing interests.

Authors' contributions

All authors have contributed to the different methodological and experimental aspects of the research. All authors read and approved the final manuscript.

Author details

¹Instituto Superior de Contabilidade e Administração do Porto (ISCAP)– Instituto Politécnico do Porto (IPP) LIACC & Faculdade de Engenharia da Universidade do Porto (FEUP), Portugal. ²LIAAD & Faculdade de Engenharia da Universidade do Porto (FEUP), Rua Dr. Roberto Frias, 4200-465 Porto, Portugal. ³LIACC & Faculdade de Engenharia da Universidade do Porto (FEUP), Rua Dr. Roberto Frias, 4200-465 Porto, Portugal.

Received: 17 May 2013 Accepted: 9 March 2014

Published: 19 June 2014

References

1. Cohen AM, Hersh WR (2005) A survey of current work in biomedical text mining. *Brief Bioinform* 6(1): 57–71. doi:10.1093/bib/6.1.57.
2. Ananiadou S, Pyysalo S, Tsujii J, Kell DB (2010) Event extraction for systems biology by text mining the literature. *Trends Biotechnol* 28(7): 381–390. doi:10.1016/j.tibtech.2010.04.005
3. Luscombe NM, Greenbaum D, Gerstein M (2001) What is bioinformatics? An introduction and overview. Tech. rep., Department of Molecular Biophysics and Biochemistry Yale University New Haven, USA
4. Smith L, Wilbur W (2011) The popularity of articles in pubmed. *The Open Information Systems Journal*, National Center for Biotechnology Information, Bethesda, Maryland, USA
5. Deng ZH, Lai BY, Wang Z, Fang GD (2012) Pav: A novel model for ranking heterogeneous objects in bibliographic information networks. *Expert Syst Appl* 39(10): 9788–9796
6. Zhang M, Feng S, Tang J, Ojokoh BA, Liu G (2011) Co-ranking multiple entities in a heterogeneous network: integrating temporal factor and users' bookmarks. In: Xing C, Crestani F, Rauber A (eds) *ICADL*. Springer, pp 202–211
7. Ratprasartporn N, Bani-Ahmad S, Cakmak A, Po J, Özsoyoglu G (2007) Evaluating different ranking functions for context-based literature search. In: *ICDE Workshops*, IEEE Computer Society, pp 261–268
8. Zhou YB, Lü L, Li M (2012) Quantifying the influence of scientists and their publications: distinguishing between prestige and popularity. *New Journal of Physics* 14(3): 033033
9. Bernstam EV, Herskovic JR, Aphinyanaphongs Y, Aliferis CF, Sriram MG, Hersh WR (2006) Research paper: using citation data to improve retrieval from medline. *JAMIA* 13(1): 96–105
10. Lin Y, Li W, Chen K, Liu Y (2007) Model formulation: a document clustering and ranking system for exploring medline citations. *JAMIA* 14(5): 651–661
11. Robertson S, Walker S, Beaulieu M, Gatford M, Payne A (1996) Okapi at trec-4. In: *Proceedings of the 4th Text REtrieval Conference (TREC-4)*, pp 73–96
12. Frakes WB, Baeza-Yates R (1992) *Information retrieval: data structures and Algorithms*. Prentice Hall, Upper Saddle River, New Jersey, USA
13. Gonçalves CA, Gonçalves CT, Camacho R, Oliveira EC (2010) The impact of pre-processing on the classification of medline documents. In: Fred A. L. N. (ed) *Pattern Recognition in Information Systems, Proceedings of the 10th International Workshop on Pattern Recognition in Information Systems, PRIS 2010, In conjunction with ICEIS 2010, Funchal, Madeira, Portugal, June 2010*, pp 53–61
14. Gonçalves CT, Camacho R, Oliveira E (2013) Biotextretriever: yet another information retrieval system. In: Correia L, Reis L, Cascalho J, Gomes L, Guerra H, Cardoso P (eds) *In Proceedings of the Workshop Text Mining and Applications (TEMA) of the 16th Portuguese Conference on Artificial Intelligence (EPIA 2013)*, pp 522–533
15. Dietterich TG (2000) Ensemble methods in machine learning. In: *Proceedings of the First International Workshop on Multiple Classifier Systems, MCS '00*. Springer-Verlag, London, UK, pp 1–15
16. Johnson M, Zaretskaya I, Raytselis Y, Yuri Merezuk SM, Madden TL (2008) Ncbi blast: a better web interface. *Nucleic Acids Res* 36(2): W5–W9. doi:10.1093/nar/gkn201.
17. Van Leeuwen TN, Visser MS, Moed HF, Nederhof TJ, Van Raan A (2003) The holy grail of science policy: exploring and combining bibliometric tools in search of scientific excellence. *Scientometrics* 57(2): 257–280
18. Bradshaw S (2003) Reference Directed Indexing: Redeeming Relevance For Subject Search in Citation Indexes. In: Koch T, SÅyilvberg IT (eds) *In Proc. of the 7th conference in the series of European Digital Library conferences (ECDL)*. Springer-Verlag, Trondheim, Norway, pp 499–510
19. Garfield E (1964) "Science citation index"—a new dimension in indexing. *Science* 144(3619): 649–654. doi:10.1126/science.144.3619.649.
20. Shadbolt N, Brody T, Carr L, Harnad S (2006) The open research web: a preview of the optimal and the inevitable
21. Hirsch JE (2010) An index to quantify an individual's scientific research output that takes into account the effect of multiple coauthorship. *Scientometrics* 85(3): 741–754. doi:10.1007/s11192-010-0193-9.
22. Franceschini F, Maisano DA (2010) Analysis of the hirsch index's operational properties. *Eur J Oper Res* 203(2): 494–504
23. Glänzel W (2006) On the opportunities and limitations of the h-index. *Sci Focus* 1(1): 10–11
24. Cronin B, Meho L (2006) Using the h-index to rank influential information scientists: brief communication. *J Am Soc Inf Sci Technol* 57(9): 1275–1278. doi:10.1002/asi.v57:9.
25. Abramo G, D'Angelo CA, Di Costa F (2010) Citations versus journal impact factor as proxy of quality: could the latter ever be preferable? *Scientometrics* 84(3): 821–833
26. Egghe L (2006) An improvement of the h-index: The g-index. *ISSI NewsL* 2(1): 1–4
27. Hasan DSA, Subhani DMI, Osman MA (2012) H-index: The key to research output assessment. MPRA Paper 39097, University Library of Munich, Germany
28. Bornmann L, Daniel HD (2007) What do we know about the *h_i* index? *JASIST* 58(9): 1381–1385
29. Van Raan AFJ (2005) Comparison of the hirsch-index with standard bibliometric indicators and with peer judgment for 147 chemistry research groups. *Scientometrics* 67(3): 12
30. Egghe L, Rao IKR (2008) Study of different h-indices for groups of authors. *J Am Soc Inf Sci Technol* 59(8): 1276–1281. doi:10.1002/asi.v59:8.
31. Egghe L (2012) Averages of ratios compared to ratios of averages: Mathematical results. *J Informetrics* 6(2): 307–317
32. Garfield E (1999) Journal impact factor: a brief review. *CMAJ Can Med Assoc J* 161(8): 979–980
33. Seglen PO (1997) Why the impact factor of journals should not be used for evaluating research. *BMJ (Clin Res Ed)* 314(7079): 498–502
34. Lippi G (2009) The impact factor for evaluating scientists: the good, the bad and the ugly. *Clin Chem Lab Med* 47(12): 1585–6
35. Smith DR (2007) Historical development of the journal impact factor and its relevance for occupational health. *Ind Health* 45(6): 730–42
36. Sewell W (1964) Medical subject headings in medlars. *Bull Med Libr Assoc* 52: 164–170
37. Hulsen T, de Vlieg J, Leunissen JAM, Groenen PMA (2006) Testing statistical significance scores of sequence comparison methods with structure similarity. *BMC Bioinformatics* 7(444)

doi:10.1186/1678-4804-20-13

Cite this article as: Gonçalves et al.: Ranking MEDLINE documents. *Journal of the Brazilian Computer Society* 2014 **20**:13.

Submit your manuscript to a SpringerOpen® journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Immediate publication on acceptance
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► springeropen.com