BIOINFORMATICS

Vol. 24 ECCB 2008, pages i119–i125
doi:10.1093/bioinformatics/btn291

Comparison of vocabularies, representations and ranking algorithms for gene prioritization by text mining

Shi Yu*, Steven Van Vooren, Leon-Charles Tranchevent, Bart De Moor and Yves Moreau
Bioinformatics group, SCD, Department of Electrical Engineering, Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, B-3001 Heverlee, Belgium

ABSTRACT

Motivation: Computational gene prioritization methods are useful to help identify susceptibility genes potentially being involved in genetic disease. Recently, text mining techniques have been applied to extract prior knowledge from text-based genomic information sources and this knowledge can be used to improve the prioritization process. However, the effect of various vocabularies, representations and ranking algorithms on text mining for gene prioritization is still an issue that requires systematic and comparative studies. Therefore, a benchmark study about the vocabularies, representations and ranking algorithms in gene prioritization by text mining is discussed in this article.

Results: We investigated 5 different domain vocabularies, 2 text representation schemes and 27 linear ranking algorithms for disease gene prioritization by text mining. We indexed 288 177 MEDLINE titles and abstracts with the TXTGate text profiling system and adapted the benchmark dataset of the Endeavour gene prioritization system that consists of 618 disease-causing genes. Textual gene profiles were created and their performance for prioritization were evaluated and discussed in a comparative manner. The results show that inverse document frequency-based representation of gene term vectors performs better than the term-frequency inverse document-frequency representation. The eVOC and MESH domain vocabularies perform better than Gene Ontology, Online Mendelian Inheritance in Man's and London Dysmorphology Database. The ranking algorithms based on 1-SVM, Standard Correlation and Ward linkage method provide the best performance.

Availability: The MATLAB code of the algorithm and benchmark datasets are available by request.

Contact: shi.yu@esat.kuleuven.be

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 INTRODUCTION

Genome-wide experimental methods to identify disease-causing genes, such as linkage analysis and association studies, are often overwhelmed by large sets of candidate genes produced by high throughput techniques for which the low-throughput validation of candidate disease genes is time consuming and expensive (Risch, 2000). Computational prioritization methods can rank candidate disease genes from these gene sets according their likeliness of being involved in a certain disease. Moreover, a systematic gene prioritization approach that integrates multiple genomic datasets provides a comprehensive *in silico* analysis on the basis of multiple sources of existing knowledge. Several computational gene prioritization applications have been previously described.

1.1 Previous approaches

Freudenberg and Propping prioritize disease relevant human genes by measuring similarities among GO annotations and validate the results in OMIM database (Freudenberg and Propping, 2002). GeneSeeker (Van Driel *et al.*, 2005) provides a web interface that filters candidate disease genes on the basis of cytogenetic location, phenotypes and expression patterns. DGP (disease gene prediction) (Lopez-Bigas and Ouzounis, 2004) assigns probabilities to genes based on sequence properties that indicate their likelihood to the patterns of pathogenic mutations of certain monogenetic hereditary disease. PROSPECTR (Adie *et al.*, 2005) also classifies disease genes by sequence information but uses a decision tree model. SUSPECTS (Adie *et al.*, 2006) integrates the results of PROSPECTR with annotation data from Gene Ontology (GO), InterPro and expression libraries to rank genes according to the likelihood that they are involved in a particular disorder. G2D (candidate genes to inherited diseases) (Perez-Iratxeta *et al.*, 2005) scores all concepts in GO according to their relevance to each disease via text mining. Then, candidate genes are scored through a BLASTX search on reference sequence. POCUS (Turner *et al.*, 2003) exploits the tendency for genes to be involved in the same disease by identifiable similarities, such as shared GO annotation, shared InterPro domains or a similar expression profile. eVOC annotation (Tiffin *et al.*, 2005) is a text mining approach that performs candidate gene selection using the eVOC ontology as a controlled vocabulary. It first associates eVOC terms and disease names according to co-occurrence in MEDLINE abstracts, and then ranks the identified terms and selects the genes annotated with the top-ranking terms. In the work of Franke *et al*. (Franke *et al.*, 2006), a functional human genetic network was developed that integrates information from KEGG, BIND, Reactome, human protein reference database, GO, predicted-protein interaction, human yeast two-hybrid interactions and microrray coexpressions. Gene prioritization is performed by assessing whether genes are close together within the connected gene network. Endeavour (Aerts *et al.*, 2006) takes a machine learning approach by building a model on a training set, then that model is used to 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. Endeavour incorporates multiple genomic data sources (microarray, InterPro, BIND, sequence, GO annotation, Motif, Kegg, EST and text mining) and builds a model on each source of individual prioritization results. Finally, these results are

*To whom correspondence should be addressed.

combined through order statistics into a final score that offers an insight on how related a candidate gene is to the training genes on the basis of information from multiple knowledge sources. More recently, CAESAR (Gaulton *et al.*, 2007) has been developed as a text mining-based gene prioritization tool for complex traits. CAESAR 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.

1.2 Gene prioritization in imbalanced datasets

The performance of the training–testing approach of gene prioritization can be evaluated by checking the positions of real relevant genes in the ranking of a test set. A perfect prioritization should rank the gene with the strongest causal link to the biomedical concept, represented by the training set, at the highest position (at the top). The interval between the real position of that gene with the top is regarded as the *error*. For a prioritization model, minimizing this *error* is equal to improving the ranking position of the most relevant gene and in turn it reduces the number of irrelevant genes to be investigated in biological experimental validation. So a model with smaller *error* is more efficient and accurate to find disease relevant genes and that *error* is also used as a performance indicator for model comparison.

A potential problem for this training–testing approach is that ranking candidate genes in the whole genome is a *class-imbalanced problem* because the majority of genes are not related to the biomedical concept represented by the training set. In a class imbalanced dataset, standard discriminant algorithms are often biased towards the majority class. Hence, they are more likely to cause a high false positive rate when the majority is labeled as negative samples. For this imbalance problem, a strategy of *one-class classification* is often proposed to reduce the error rate on the majority class (Estabrooks *et al.*, 2004; Tax, 2002). The problem of *one-class classification* can be easily transformed to *one-class prioritization* as an information retrieval problem since classification is often based on ranking of distances to the density of class samples. A simple one-class prioritization model is to rank the candidate genes by their distances to the center of training genes, which is equal to the similarity value obtained by standard correlation on data with the same norm. Another complex model looks for a small coherent subset of genes, which can be achieved by finding a small-radius ball that covers as many training genes as possible (Tax and Duin, 1999). Obviously, the genes lying within the ball are more likely to be relevant than those lying outside. Thus, prioritization is performed by ranking the distance of candidate genes to the center of the ball. In a similar problem, one class Support Vector Machines (De Bie *et al.*, 2007; Scholkopf *et al.*, 2001) is applied to separate most of the training genes from the origin using a hyperplane and prioritization can be achieved by ranking the distance to the hyperplane. The prioritization model can also be extended by clustering methods and vary by different criteria of clustering and distance measures. Most of these formulations are similar in the way assigning a convex score function on the basis of Euclidean distance. The global minimum of this score function is at the center of the training samples (or the ball), then it increases linearly towards the outside. If the number of training genes is large, the score function can be further regularized by penalizing outliers among the training genes. After regularization, some outliers in the training set are regarded as irrelevant samples. Hence, a ball with smaller radius is obtained and it might improve the precision of prioritization. In this article, we will regard gene prioritization as an imbalanced learning problem and employ several *one-class prioritization* algorithms and compare their performance.

1.3 Gene prioritization in high dimensional datasets

Current genomic datasets are usually high dimensional. As known, high-dimensional data is a double-edged sword for statistical analysis (Donoho, 2000). For the task of gene prioritization the high dimensionality of the dataset influences two aspects: First, discriminating relevant genes from irrelevant ones is more likely to be a linear problem because it is often easier to find a separating hyperplane in higher dimension. Second, processing high-dimensional data with parametric methods is difficult because these methods require an appropriate ratio of samples and variables. Moreover, the complexity of estimation, optimization and integration of these methods grows exponentially with the dimension. The second problem is also known as *the curse of dimensionality* (Bellman, 1961). For these reasons, in this aricle we will focus on several non-parametric ranking methods for high-dimensional data.

1.4 Approach and motivation

We adopted a high-dimensional benchmarking dataset generated by the biomedical literature mining system TXTGate (Glenisson *et al.*, 2004). TXTGate indexes titles and abstracts of MEDLINE with different vocabularies and weighting schemes. Then, the *documents* × *terms* matrix is transformed into *genes* × *terms* matrix according to the curated gene-to-doc mapping in EntrezGene. These gene-by-term vectors, denoted as *textual profiles*, represent existing expert knowledge about genes from free text and have been successfully applied in text-based gene clustering (Glenisson *et al.*, 2004) and gene prioritization (Aerts *et al.*, 2006) applications. We could also use other non-textual profiles, such as microarray data. In Endeavour (Aerts *et al.*, 2006), the similarity of genes is measured by standard correlation and the prioritization performance on textual gene profiles is higher than for other data sources [Supplementary Fig. 1 of (Aerts *et al.*, 2006)]. This is partly because results on textual profiles are biased towards existing knowledge, since evaluation of prioritization is obtained by benchmarking disease related genes that are already known. On the other hand, the low performance on some other datasets might be caused by several factors, for example, the pre-processing methods of original data, the influence of normalization methods, etc., so they are not suitable for benchmark datasets in our problem. In text mining approaches, the effect of different vocabularies and representations is still an open question and they have been mostly selected empirically in previous approaches. The importance of text mining in gene prioritization makes its optimization an important issue. In this article, we will focus on these implied problems: (1) choice of vocabularies in text mining, (2) choice of representations for text-based data vectors and (3) comparison of different linear ranking algorithms in unbalanced datasets.

2 DATASETS AND METHODS

2.1 Datasets

2.1.1 Textual profiles of genes We created 10 groups of textual gene profiles on the text mining platform TXTGate. Various literature indices were created based on title text and abstract text of MEDLINE publications and linked MEDLINE information presented in EntrezGene. Five vocabularies (Table 1) derived from public resources act as perspective on the textual information with different levels of detail. The first vocabulary is derived from GO. The names of all GO terms are retrieved from the online repository, then processed by different kind of filters. Through these filters, the terms are stemmed (Porter, 1980), the stopwords and punctuations are removed. After this treatment, we obtained a GO domain vocabulary of 23 857 terms.

The second vocabulary is based on the Medical Subject Headings (MeSH), the National Library of Medicine's controlled vocabulary thesaurus. After the same pre-processing procedures as for the GO vocabulary, we obtained 30 136 terms for MeSH vocabulary.

The third vocabulary is retrieved from the online mendelian inheritance in man's (OMIM) morbid map, the cytogenetic location of all disease genes present in OMIM and their associated diseases and consists of 5576 terms.

The fourth vocabulary is based on the London Dysmorphology Database (LDDB), which contains information on dysmorphic and neurogenetic syndromes. We extracted dysmorphology concepts as vocabulary terms and 935 terms were obtained after pre-processing.

The fifth domain vocabulary is drawn from eVOC, an ontology consisting of four orthogonal controlled vocabularies (anatomical system, cell type, pathology and developmental stage) subsuming the domain of human gene expression data. After filtering, we obtained 1788 eVOC terms.

Among these vocabularies, four of them are also used in TXTGate system. Using these controlled vocabularies, we indexed 288 177 MEDLINE titles and abstracts with reference to the mapping of EntrezGene. The terms from the domain vocabulary are regarded as a *bag-of-words* hence the indexed documents are represented as vectors in the space spanned by these terms. Based on the gene-to-doc mappings in EntrezGene, multiple linked documents of a same gene were combined as a single averaged gene profile and all gene profiles are normalized to obtain gene vectors on a unit space. For each domain vocabulary we investigated representation schemes to calculate the value of terms in vectors: inverse document frequency (IDF) and term-frequency × inverse document frequency (TFIDF). Apart from these, we had also implemented a binary scheme as a simplest baseline of representation. However, the performance of binary scheme is not comparable with IDF and TFIDF ones so it is not presented in this article. After combining different vocabularies and representations, we obtained 10 groups of textual profiles. The overview of the size and overlapping terms of vocabularies after indexing is presented in Table 1. In Table 2 some highest ranking terms and lowest ranking terms are listed as examples. To compare the effect of vocabularies in text-based gene prioritization, we also created a group of special profiles that uses no controlled vocabulary in the text mining procedure, denoted as no-voc profile. When no vocabulary is used, all the terms once appearing once in the referenced MEDLINE titles and abstracts in EntrezGene are regarded as useful annotations for text mining. The conceptual overview of obtaining textual gene profiles and the formulations for computing IDF and TFIDF representations are available in the Supplementary Material. The details of profiling genes using textual information is presented in the TXTGate paper (Glenisson *et al*., 2004).

2.1.2 Benchmark dataset of disease relevant genes We used the benchmark dataset of Endeavour (Aerts *et al*., 2006), which consists of 618 relevant genes from 29 diseases. Genes from the same disease were constructed as a disease-specific training set used to benchmark the prioritization performance. The name of diseases and the number of genes related to the diseases are shown in Table 1 of the Supplementary Material.

2.2 Prioritization algorithms

We implemented 27 models of non-parametric prioritization algorithms categorized in three different types: regularized one-class Support Vector Machines, k-nearest neighbor method and clustering method, which is implemented as k means clustering and hierarchical clustering.

2.2.1 One-class SVM The one-class SVM method, suggested by Scholkopf (Scholkopf *et al*., 2001), extends the binary SVM classification scheme into one-class learning by mapping the training data that contains just one class into a high-dimensional Hilbert space via a kernel function. The algorithm iteratively finds the maximal margin hyperplane that best separates the training data from the origin. In the present article, we only use linear kernels because the dimensionality of the data is very high. In prioritization task, the decision function of one-class SVM in (Scholkopf *et al*., 2001) is extended to a prioritization function by dropping the sign function and the constant value ρ solved by one-class optimization.

2.2.2 k-nearest neighbor The nearest neighbor methods we used in this article are proposed by (Tax, 2002). In the present article, we tried three different k values ($k=1,2,3$). When $k\geq 2$, three varieties of nearest neighbor algorithms were implemented, denoted as κ, δ and γ, according to the differences of averaging the distance of test data to the k nearest neighbors.

2.2.3 K-means clustering The objective function of K-means is

$$\min_{\vec{c}_k} \sum_i (\|\vec{x}_i - \vec{c}_k\|^2). \tag{1}$$

The prioritization is achieved by ranking the distance of the test gene to the centroid(s). In this article we tried three different K values ($K=1,2,3$). Notice that when $K=1$ and if all data have the same norm, the K-means algorithms is equivalent to the standard correlation (Pearson correlation) method, which directly measures the angular separation of candidate gene between averaged vectors of training genes around the origin. If the data is clustered into more than one clusters, there is a choice to select the maximum, minimum or average distance of a test gene to multiple centroids as the prioritization score.

Table 1. Overview of the sizes of domain vocabularies, the number of overlapping terms among vocabularies and the number of indexed human genes through textual profiling

Domain vocabulary	Number of terms	Number of overlapping terms					Number of indexed human genes
		GO	MeSH	OMIM	eVOC	LDDB	
GO	10 249	–					23 875
MeSH	17 201	2812	–				23 875
OMIM	3462	526	1587	–			23 875
eVOC	1496	277	772	339	–		23 865
LDDB	933	65	331	206	103	–	16 212

Table 2. Examples of the most frequent terms and the least frequent terms in different vocabularies

	GO	MeSH	OMIM	LDDB	eVOC
Highest Rank					
1	Cell	Protein	Cell	Growth	Cell
2	Protein	Express	Protein	Brain	Human
3	Express	Cell	Express	Liver	Associ
4	Gene	Gene	Gene	Muscl	Induc
5	Activ	Activ	Activ	Kidnei	Factor
6	Function	Result	Function	Lung	Type
7	Regul	Suggest	Specif	Heart	Depend
8	Specif	Function	Bind	Calcium	Develop
9	Sequenc	Studi	Factor	Skelet	Famili
10	Induc	Human	Associ	Lipid	Site
Lowest Rank (freq = 1)					
1	Coniferin	Abelmoschu	Meleda diseas	Arpal bone fusion	Spermatozoid
2	Protein autoubiquitin	Tyrpcidin	Mast syndrom	Muscular build	66 yr
3	Acid ammonia	Intern agenc	Lindau	Enchondromata	Myofibrobast
4	Prenol	Brain injuri chronic	Leydig cell adenoma	Absent parathyroid	Toddler
5	Phenylserin	Integrin alphaxbeta2	Kina	Flat face	Superior vestibular nuclei
6	Adenin metabol	Mytilida	Kappa light chain defici	Enlarg lymph gland	Hensen cell
7	Class iii pi3k	Myofasci	Bradyopsia	Abnorm scar format	Ag 86
8	Nutrient import	Enoxaprain	Woud	Cowlick	Peptic cell
9	Ey antenn disc develop	Nasal provoc test	Zlotogora	Septum pellucidum	Endoth
10	Liga activ	Celliprolol	Anisomastia	Point chin	Medial accessori

2.2.4 Hierarchical clustering Similarly, the data can also be clustered by linkage methods. In this article, we tried four different linkage methods (Single linkage, complete linkage, average linkage and Ward linkage) to cluster training genes into two clusters and ranked the candidate gene according to its distance to the clustering centroids either by max, min or average function. In total 12 different hierarchical clustering methods are used in this article.

Details about the prioritization algorithms used in this article are available in the Supplementary Material.

2.3 Evaluation of prioritization

2.3.1 Leave one out (LOO) validation The performance of algorithms was evaluated by LOO prioritization. In each experimental test on a disease gene set, which contains K genes, one gene, termed the 'defector' gene, was deleted from a set of training genes and added to M randomly selected test genes, denoted as the test set. We used the remaining $K-1$ genes, denoted as the training set, to train our prioritization model. Then, we prioritized the test set, which contains $M+1$ genes by the trained model and determined the ranking of that defector gene in test data. The prioritization performance was evaluated by the error between the perfect ranking and the combined ranking position of all defector genes in the disease set with the following equation

$$Error = 1 - \frac{M}{M-1}\left[1 - \frac{1}{K}\sum_{i=1}^{K}\frac{r_i}{M}\right], \quad (2)$$

where r_i is the ranking position of the i-st gene in the disease set, K is the number of genes in the disease set, $\frac{M}{M-1}$ is a normalization term to make the perfect ranking equal to 1 and leads the Error to 0. In order to benchmark algorithms in a class imbalanced dataset, we set the number of random genes M to 9999.

2.3.2 Similarity of prioritization We used Spearman's rank correlation to compare the ranking order of two prioritization results P_1 and P_2 obtained on identical n genes,

$$\rho = 1 - \frac{6\sum d_i^2}{n(n^2-1)}, \quad (3)$$

where d_i is the difference between rankings in P_1 and P_2 on corresponding genes. For each disease set, we randomly selected 99 genes and calculated a Spearman correlation matrix when each defector gene is left out. Then, we averaged the Spearman matrices for all the genes in one disease set. For all disease sets, 29 Spearman matrices were further averaged and the final matrix was used to compare the similarity of all algorithms on ranking results.

3 RESULTS AND DISCUSSION

We compared the performance of the prioritization algorithms and textual gene profiles by LOO cross-validation on 9999 random genes. Some significant results are shown in Figures 2 and 3. The complete table of overall benchmark result is shown in the Supplementary Material (Table 2). The performance obtained on IDF profiles is significantly better than for TFIDF profiles. When IDF profiles are used, eVOC and MeSH domain vocabularies are significantly better than GO, LDDB and OMIM. Generally, the errors of ranking algorithms based on 1-SVM, standard correlation and average ward linkage are smaller than other algorithms.

3.1 Representation schemes: IDF performs better than TFIDF

The comparison of errors on the textual representation schemes of terms shows that IDF is generally better than TFIDF in text mining-based gene prioritization. In Figure 1, we compared the errors of two representation schemes on all domain vocabularies and three best ranking algorithms. The minimal error obtained by IDF profile is (eVOC, 1SVM, 0.0477) while the minimal one by TFIDF is (GO,

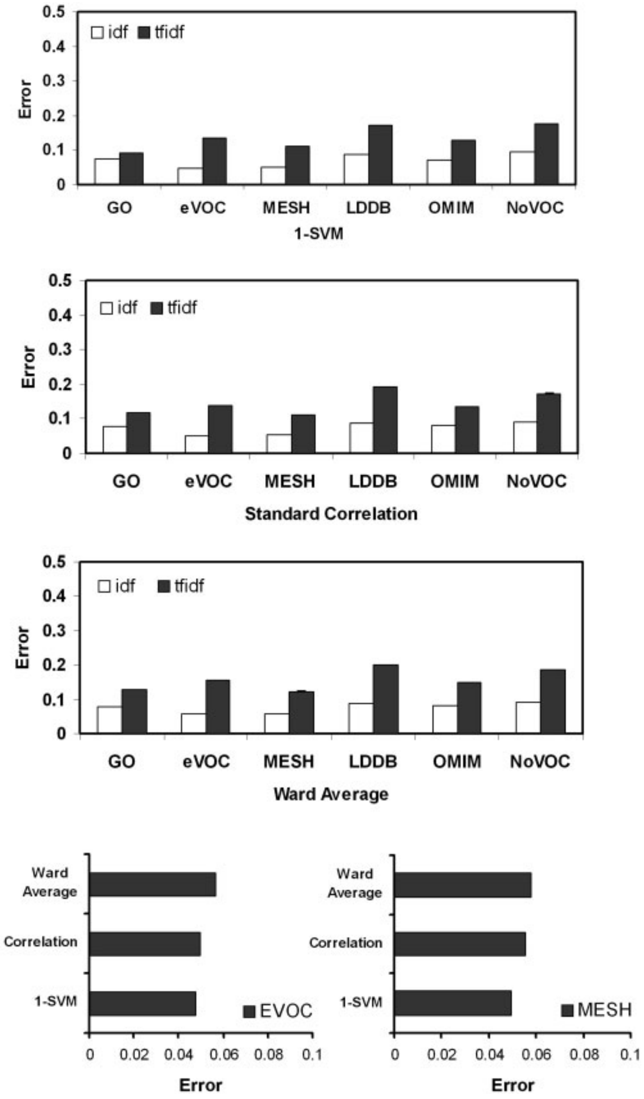

Fig. 1. Errors of LOO prioritization results on different vocabularies, representations and ranking algorithms. The figure shows the prioritization results obtained by three best ranking algorithms. The top three figures compare the performance of different vocabularies and representation schemes. The figure on the fourth row compares three ranking algorithms. Since the validations use 9999 random genes, the deviations of all prioritization errors are smaller than 0.0001 so they are not mentioned explicitly in the figures.

1SVM, 0.0916), which means the error of best IDF profile is almost 50% less than the TFIDF one. Moreover, when the same domain vocabulary and same ranking algorithm is used, error with IDF is always smaller than with TFIDF. This is mainly because some rare terms play an important role in distinguishing the term vectors of genes from disease to disease, and through IDF representation, these rare discriminative terms get large values and dominate the prioritization results. In contrast, TFIDF tries to balance the effects of IDF and TF by multiplying them, which in fact weakens the discriminative effect for gene prioritization.

3.2 Domain vocabularies: eVOC and MESH perform better than LDDB, GO and OMIM

When the same algorithm and representation are applied, the errors obtained on eVOC and MESH vocabulary are much smaller than other vocabularies. For example, using 1-SVM and IDF, the errors on eVOC (0.0477) and MESH (0.0497) are much better than LDDB (0.0877), GO (0.0758) and OMIM (0.0714). The same situation happens for other algorithms as well (see Supplementary Material Table 1). This result is interesting since the size of the MESH vocabulary is almost 10 times larger than that of eVOC. The actual reason of why they outperform others is an issue requiring further investigation. According to our experimental results obtained from a random vocabulary, the size of the random vocabulary directly determines the error of prioritization result (Fig. 2). The larger the random vocabulary the smaller the error in prioritization. However, the size of domain vocabulary does not impact the performance directly, it is the semantic content of the vocabulary that matters. This also raises an open question about the existence of an optimal vocabulary for the problem of gene prioritization. Discussion about this topic would also be important but it is beyond the scope of this article.

3.3 Prioritization algorithms

In the beginning of the article, we proposed the strategy of *one-class prioritization* and the effect of regularization with respect to the issue of class imbalancing. According to the benchmark result of 27 different linear non-parametric ranking algorithms, 1-SVM, correlation and ward average linkage are the three best algorithms.

These three ranking algorithms are similar in the sense that their ranking scores are almost equal to the distance toward the density center of the training genes. In standard correlation, the ranking score is equal to the distance from the candidate gene to the center of all training genes. In 1-SVM, the score is the distance to the center of the ball that covering the training genes by regularization. During regularization, some training genes that are far from the original center are removed and the new center is recalculated. The ward linkage method is also a well-known agglomerative hierarchical clustering method and it is reported with good results in many information retrieval and pattern detection applications. In the implementation of *ward average linkage* in this article, the number of clusters is set to 2 and the average distance towards the 2 ward linkage clustering centroids is used as ranking score.

3.4 Clustering of prioritization algorithms

We used the Spearman correlation to measure the similarity of gene prioritization results obtained by two different algorithms. Similar to LOO cross-validation, in each disease benchmark set, a ‘defector’ gene was left out and mixed with 99 random genes. To compare the results on different algorithms, the random gene list was kept identical when the same gene was left out. Then the average correlation of the disease benchmark set is computed, furthermore, the final average correlation of all 29 disease sets is obtained and regarded as the correlation of the prioritization algorithms. Based on the pairwise Spearman correlation matrix of all 27 prioritization models presented in the Supplementary Material (Table 3), we clustered these 27 models in the dendrogram by complete linkage (Fig. 3). Standard correlation is highly similar to ward average linkage in ranking (Spearman correlation = 0.9915).

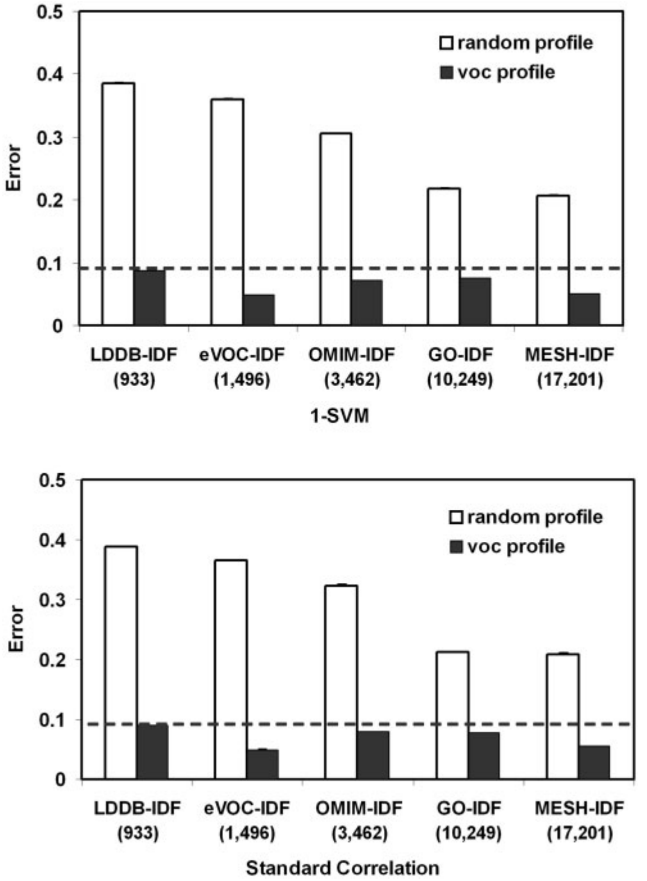

Fig. 2. Comparison of prioritization performance using text profiles based on random vocabularies, domain vocabularies and no vocabulary. The horizontal line is the error of prioritization obtained by *no-voc* profile, which contains 259 815 terms resulted from text mining process without using any vocabulary. Based on this *no-voc* profile, we randomly selected several subsets of terms and created five *random-voc* profiles as the comparison sets to the domain vocabulary profiles. The performance obtained by domain vocabulary profiles is compared with the *random-voc* profiles that have the same number of terms. On the X-axis, the profiles are sorted from smaller size to larger size. As it shows, the performances of *random-voc* profiles increase monotonically with the vocabularies size. On the contrary, the performance of domain vocabulary profile does not solely depend on the size of vocabulary but is mainly determined by its semantic content.

1-SVM is similar to several minimal distance methods. Nearest neighbor methods and maximal distance methods are quite different from the forementioned methods.

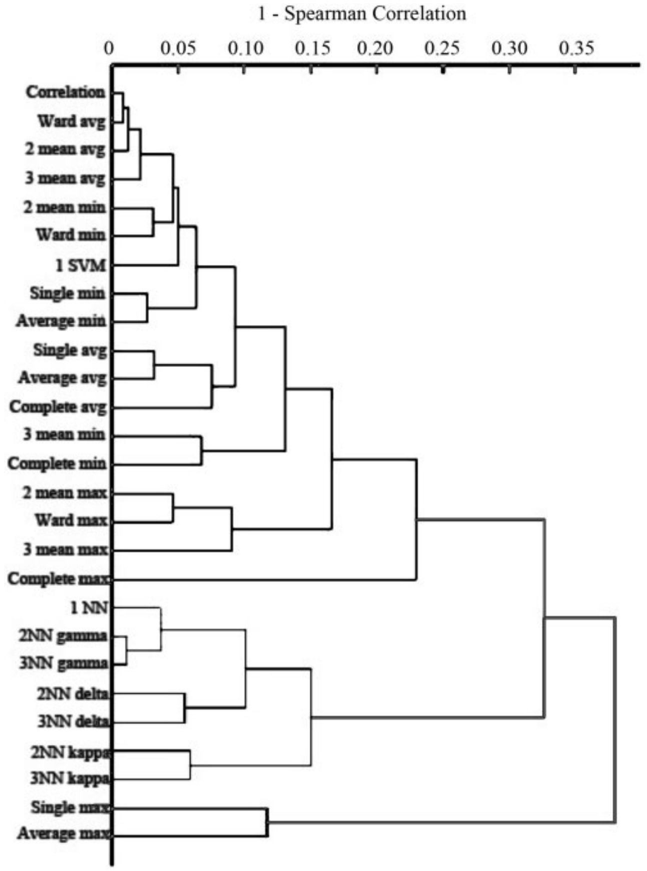

Fig. 3. A dendrogram of clustering 27 prioritization models through Spearman correlation analysis.

3.5 Selecting the best configuration in text mining-based gene prioritization

From now on, for conciseness, we use the term *configuration* to denote the triplet choice of domain vocabulary, representation scheme and ranking algorithm. On the basis of the experimental results and previous discussion, the *configuration* has a strong impact on the quality of prioritization model. According to the result of full benchmark experiments, the improperly selected *configuration* could lead to a large error (no-voc, single max, TFIDF, 0.3757) on prioritization, which is > 7 times larger than the error of a carefully selected *configuration* (eVOC, 1-SVM, IDF, 0.0477). If the prioritization result is used as the reference list in biological validation, the efficiency gained from a good *configuration* will be remarkable.

3.6 Results of profile integration

According to the results on domain vocabulary-based profiles, we picked the best two IDF profiles (eVOC and MESH), the best two TFIDF profiles (GO and MESH) and the best of each of them (eVOC-IDF and MESH-TFIDF) and integrated them by three integration functions (*min*, *max* and *average*). Although there are some consistent improvement by integrating text profiles, however, the improvements are too small to be relevant so we do not discuss it in this article. The explanation of integration methods and results are available in the Supplementary Material (Table 4).

4 CONCLUSION

In this article, we presented an approach of comparing different *configurations* to create and rank textual profiles for gene prioritization. By integrating the TXTGate text mining profiling system and prioritization framework from the Endeavour system, we investigated 5 domain vocabularies, 2 text mining weighting schemes and 27 ranking algorithms (270 *configurations*). Our discussion can be mainly concluded as the following: controlled domain vocabulary provides an effective view to conduct text mining for gene prioritization, moreover, the impact of the selection of *configurations* on prioritization performance is significant. For the representation of vector-based data, IDF representation of terms causes less error than TFIDF representation. eVOC and MESH domain vocabularies give smaller errors than the GO, OMIM and LDDB vocabularies. Among the 27 models we benchmarked, 1-SVM, standard correlation and ward linkage method are the better candidates for ranking algorithm. In short, the selection of configurations is an important factor of the quality of disease-oriented prioritization model by text mining.

ACKNOWLEDGEMENTS

Funding: This work was supported by Research Council K.U.Leuven: GOA /2005/11, EF/05/007, PhD grants; GOA-AMBio RICS, CoE-EF/05/007 SymBioSys, FWO-G.0241.04, FWO-G. 0499.04, FWO-G.0232.05, FWO-G.0318.05, FWO-G.0553.06, FWO-G. 0302.07, FWO-ICCoS, FWO-ANMMM, FWO-MLDM, IWT-McKnow-E, IWT-GBOU-ANA, IWT-TAD-BioScope-IT, IWT-Silicos, IWT-SBO-BioFrame, IAP-P6/25, EU-ERNSI, FP6-NoE-Biopattern, FP6-IP-e-Tumours, FP6-MC-EST-Bioptrain, EC-FP6-STREP-strokemap.

Conflict of Interest: none declared.

REFERENCES

Adie,E.A. *et al*. (2005) Speeding disease gene discovery by sequence based candidate prioritization. *BMC Bioinformatics*, **6**, 55.

Adie,E.A. *et al*. (2006) SUSPECTS: enabling fast and effective prioritization of positional candidates. *Bioinformatics*, **22**, 773–774.

Aerts,S. *et al*. (2006) Gene prioritization through genomic data fusion. *Nat. Biotechnol.*, **24**, 537–544.

Bellman,R.E. (1961) *Adaptive Control Processes: A Guided Tour*. Princeton University Press, Princeton, New Jersey.

De Bie,T. *et al*. (2007) Kernel-based data fusion for gene prioritization, *Proc. ISMB 2007*, **23**, 125–132.

Donoho,D.L. (2000) High-dimensional data analysis: the curses and blessings of dimensionality. *Neural Comput*, Aide-Memoire of a Lecture at AMS Conference on Math Challenges of the 21st Century. Available at http://www-stat.stanford.edu/~donoho/Lectures/AMS2000/AMS2000.html.

Estabrooks,A. *et al*. (2004) A multiple resampling method for learning from imbalanced data sets. *Comput. Int.*, **20**, 18–36.

Franke,L. *et al*. (2006) Reconstruction of a functional human gene network, with an application for prioritizing positional candidate genes. *Am. J. Hum. Genet.*, **78**, 1011–1025.

Freudenberg,J. and Propping,P. (2002) A similarity-based method for genome-wide prediction of disease-relevant human genes. *Bioinformatics*, **18** (Suppl 2), 110–115.

Gaulton,K.J. *et al*. (2007) A computational system to select candidate genes for complex human traits. *Bioinformatics*, **23**, 1132–1140.

Glenisson,P. (2004) Integrating scientific literature with large scale gene expression analysis. Ph.D thesis, K.U.Leuven.

Glenisson,P. *et al*. (2004) TXTGate: profiling gene groups with text-based information. *Genome Biol.*, **5**, R43.

Lopez-Bigas,N. and Ouzounis,C.A. (2004) Genome-wide indentification of genes likely to be involved in human genetic disease. *Nucleic Acids Res.*, **32**, 3108–3114.

Perez-Iratxeta,C. *et al*. (2005) G2D: a tool for mining genes associated with disease. *BMC Genet.*, **6**, 45.

Porter,M.F. (1980) An algorithm for suffix stripping. *Program*, **14**, 130–137.

Risch,N.J. (2000) Searching for genetic determinants in the new millennium. *Nature*, **405**, 847–856.

Scholkopf,B. *et al*. (2001) Estimating the support of a high-dimensional distribution. *Neural Comput.*, **13**, 1443–1471.

Tax,D.M.J. (2002) One-class classification: concept-learning in the absence of counter-examples. Ph.D thesis, Delft University of Technology.

Tax,D.M.J. and Duin,R.P.W. (1999) Support vector domain description. *Pattern Recogn. Lett.*, **20**, 1191–1199.

Tiffin,N. *et al*. (2005) Integration of text- and data-mining using ontologies successfully selects disease gene candidates. *Nucleic Acids Res.*, **33**, 1544–1552.

Turner,F.S. *et al*. (2003) POCUS: mining genomic sequence annotation to predict disease genes. *Genome Biol.*, **4**, R75.

Van Driel,M.A. *et al*. (2005) GeneSeeker: extraction and integration of human disease-related information from web-based genetic databases. *Nucleic Acids Res.*, **33** (Web Server issue), 758–761.