Citation: Maguitman, A. G., Rechtsteiner, A., Verspoor, K., Strauss, C.E., Rocha, L.M. [2006]. "Large-Scale Testing Of Bibliome Informatics Using Pfam Protein Families". In: *Pacific Symposium on Bioinformatics 2006*. In Press.

LARGE-SCALE TESTING OF BIBLIOME INFORMATICS USING PFAM PROTEIN FAMILIES

ANA G. MAGUITMAN[†], ANDREAS RECHTSTEINER[‡,*]
KARIN VERSPOOR[‡], CHARLIE E. STRAUSS[‡], LUIS M. ROCHA[†]

[†]*School of Informatics, Indiana University*
1900 East Tenth Street, Bloomington, IN 47408
E-mail: anmaguit@indiana.edu, rocha@indiana.edu

[‡]*Los Alamos National Laboratory*
PO Box 1663, Los Alamos, NM 87545
E-mail: arechtsteiner@gmail.com, verspoor@lanl.gov, cems@lanl.gov

Literature mining is expected to help not only with automatically sifting through huge biomedical literature and annotation databases, but also with linking bio-chemical entities to appropriate functional hypotheses. However, there has been very limited success in testing literature mining methods due to the lack of large, objectively validated test sets or "gold standards". To improve this situation we created a large-scale test of literature mining methods and resources. We report on a specific implementation of this test: how well can the Pfam protein family classification be replicated from independently mining different literature/annotation resources? We test and compare different keyterm sets as well as different algorithms for issuing protein family predictions. We find that protein families can indeed be automatically predicted from the literature. Using words from PubMed abstracts, of 3663 proteins tested, over 75% were correctly assigned to one of 618 Pfam families. For 90% of proteins the correct Pfam family was among the top 5 ranked families. We found that protein family prediction is far superior with keywords extracted from PubMed abstracts than with GO annotations or MeSH keyterms, suggesting that the text itself (in combination with the vector space model) is superior to GO and MeSH as a literature mining resources, at least for detecting protein family membership. Finally, we show that Shannon's entropy can be exploited to improve prediction by facilitating the integration of the different literature sources tested.

1. Introduction

Biology was until recently essentially a hypothesis driven science in which experiments were carefully designed to answer one or very few specific questions — e.g. test the function of a specific protein in a specific context. In the last decade, fueled by the widespread use of high-throughput technology, we have witnessed

[*] Address since Oct 2005: Center for Genomics and Bioinformatics, Indiana University, 1001 East 3rd St, Bloomington, IN 47405.

the emergence of a more data-driven paradigm for biological research. Since high-throughput experiments are frequently conducted for the sake of discovery rather than hypothesis testing, and due to the sheer amount of measured variables they entail, it is very difficult to interpret their results. Moreover, since the goal of many experiments is to uncover bio-chemical and functional information about genes and proteins, there is an obvious need to understand the linkages amongst biological entities in literature and databases which allow us to make inferences. Literature mining[18] is expected to help with those inferences; its objective is to automatically sort through huge collections of literature and suggest the most relevant pieces of information for a specific analysis task, e.g. the annotation of proteins[9]. Another application is to uncover similarities of genes according to "publication space", or the more tongue-in-cheek term "bibliome"[8].

Since literature mining hinges on the quality of available sources of literature as well as their linkage to other electronic sources of biological knowledge, it is particularly important to study the quality of the inferences it can provide. Indeed, the *Bibliome* is not just the collection of publications and annotations available; its usefulness ultimately depends on the quality of linking resources that allow us to associate experimental data with publications and annotations. Interestingly, while literature mining is receiving considerable attention in Bioinformatics, it has not been hitherto seriously validated. Towards improving this situation, we present here our large-scale testing and comparison of literature mining algorithms, *paired* with specific bibliome resources.

We present a general method for testing bibliome resources and literature mining algorithms in the context of classification of biological entities. This method formalizes and extends a previous study in which we tested how well is the Pfam protein family classification inferred from PubMed as indexed by the MeSH keyterm vocabulary[16,14]. We expand on these results by testing additional bibliome resources such as GO annotations and text extracted from PubMed abstracts for the same classification problem. We additionally propose a new method based on Shannon's entropy to integrate results from different bibliome resources, and show that it significantly improves protein family predictions.

2. From Text Mining to the Bibliome: Looking for a "Gold Standard"

There exists extensive cross-linkage amongst biomedical databases which can be exploited for bioinformatics analysis. For instance, gene chip identifiers can be linked to protein entries in SWISSPROT which in turn can be linked to PubMed documents. Indeed, in the bibliome, documents are linked to or indexed by various semantic (textual) tags which describe their content; these include Medical Subject Headings (MeSH), Gene Ontology (GO) annotations, PubMed abstract text, HUGO nomenclature for human genes, GenBank accession numbers for gene sequences, etc. Therefore, in order to fully capture the potential of the bibliome for analysis, integration and dissemination of biological knowledge, in addition to research on text mining and natural language processing, literature mining needs more research on the quality of links amongst the resources that make up the bibliome. Text Mining is particularly applicable to the discovery of relevant information inside text — e.g. discovering a portion of text in a document appropriate to annotate a given protein[9]. But given the highly cross-linked nature of the bibliome, in addition to text mining, we need to approach bibliome informatics from an Information Retrieval (IR) perspective.

Several research groups have been exploiting the cross-linked nature of the bibliome, particularly with semantic annotations such as MeSH and GO, for instance the systems developed by Masys et al[13] and Jenssen et al[11] for identifying sets of keyterms associated with sets of genes. Tools that are similar in spirit are PubMatrix[2], MedMiner[22], MeshMap[21] and others. While these systems are potentially very useful, the quality of their results has not been thoroughly validated. For instance, we have applied Latent Semantic Analysis (LSA) to discover functional themes[15,14] from the literature for microarray experiments dealing with the response to human cytomegalovirus infection. Though the functional themes we discovered automatically matched our previously published manual annotation of the same experiments[4], and even uncovered novel functional themes[15,14], such validation by a few expert biologists is done without access to a "*gold standard*".

By "gold standard" we mean a standardized test data which allows us, unambiguously, to decide if a given inference is correct. Homayouni et al were able to build such gold standard for evaluating the performance of LSA, but only by focusing on a very small set of genes[10]. Unfortunately, for data-driven experiments there is no clear expectation of what functional associations are to be found. Therefore, bibliome tools are typically tested by sampling some of their output and presenting it to experts. The problem is that experts typically disagree or cannot be an expert on all the topics involved. Even more systematic approaches such as Biocreative suffer from variability in experts' opinions[5,9,3], leading to potentially unreliable answers.

3. Large-scale standard for bibliome informatics: Methods and Data

3.1. *A general large-scale bibliome informatics test*

The first requirement for our testing methodology is the existence of a biological classification $\mathcal{C}$, accepted as a true standard, and defined on a large set P of biological entities p (e.g. proteins or genes), where each entity p is associated with a single class $C(p)$. Given that the Bibliome is defined not only by publication and annotation resources, but also by their linkage, we also need a high-quality linking resource L_D between P and the documents of some publication or annotation resource D — where $L_D(p)$ denotes the set of documents of D associated with entity p. Given a $\mathcal{C}$ and L_D pair, our *large-scale bibliome informatics test* (LSBIT) can be applied to any pair, $\langle A, K_D \rangle$, of classification algorithm A and keyterm set K_D extracted from D — where $K_D(p)$ denotes the set of keyterms that index documents $L_D(p)$[a]. The objective of the LSBIT is then to **establish how well a given algorithm A can discover a known classification $\mathcal{C}$ of biological entities P, from a publication resource D using an associated keyterm set K_D and a bibliome linking resource L_D between P and D** .

3.2. *Bibliome Resources*

3.2.1. *Defining $\mathcal{C}$ and L_D*

We chose the Pfam protein sequence classification[20] as $\mathcal{C}$ for our tests. Pfam is a manually curated collection of protein families, currently encompassing several thousands of families. Pfam is an ideal classification for objective evaluation and comparison of Bibliome informatics due to it being based on sequence, which is a physical property of proteins that typically leads to functional similarity. Having settled on Pfam for our classification standard $\mathcal{C}$, our biological entities P are proteins. Therefore, a most appropriate linking resource L_D to test various $\langle A, K_D \rangle$ is the SWISSPROT (now UNIPROT[19]) database, which is a protein sequence database curated by experts. Besides the amino acid sequence of a protein it also lists different types of annotations, cross-references to other databases (including the Pfam family of a protein), as well as references to relevant publications for each protein. Therefore, the LSBIT with $\mathcal{C}$ = Pfam and L_D = SWISSPROT, can be applied to classify proteins p under various pairs $\langle A, K_D \rangle$. The expert nature of Pfam and SWISSPROT allows us to use them as a standard for the classification of proteins.

However, before the LSBIT may be performed, some preprocessing of the set of proteins to be tested is necessary. We extracted all the SWISSPROT protein IDs which contained a single Pfam classification. Multiple Pfam family assignments occur for 15% of all SWISSPROT proteins, possibly because some proteins have more than one classified domain. Because we are interested in constructing a large, unambiguous data set for validating bibliome methods, we removed multi-classification proteins. We do not consider those to be erroneous in any way, but they simply do not serve the purposes of out testing standard, which needs to be unambiguous. After pre-processing (details in[14,16]), we obtained a dataset with

[a] We use keyterm to refer to both keywords and keyphrases depending on available resources.

$|P| = 15,217$ proteins from $\mathcal{C} = 1611$ Pfam families. Each protein p is associated with a unique Pfam family $C(p)$.

3.2.2. *Defining publication/annotation resources D*

Since SWISSPROT lists PubMed IDs, a very natural publication resource is PubMed; let us denote it as D_{PM}. Via SWISSPROT, our linking resource L_D, we retrieve different keyterm sets K_D from PubMed, detailed in the next subsection. Another annotation resource we used was GO, which we denote as D_{GO}, derived from the GOA/UNIPROT dataset provided by the GOA project, run by the European Bioinformatics Institute (EBI). Because we needed to compare and integrate the tests using D_{PM} and D_{GO}, we looked at a reduced set of proteins for which links to both PubMed publications and GO annotations were found, that is $P^r = \{p : L_{D_{PM}}(p) \bigcap L_{D_{GO}}(p) \neq \emptyset\}$. We also restricted our study to Pfam families with at least 3 proteins. This reduced dataset P^r contains 3663 proteins from 618 distinct Pfam families, where 179 of these families contain only 3 proteins and the largest 3 families contain 17 proteins. Mean and median family size is 5.9 and 5 proteins, respectively; standard deviation is 3.3.

3.3. *Keyterm Sets K_D to Test*

We have adapted the IR vector space model[1] to represent proteins as vectors in a keyterm space. Four different keyterm sets were used in our analysis. Three of these sets contain keyterms extracted from PubMed (D_{PM}) publications associated with proteins, while the fourth was based on term annotations in the Gene Ontology (D_{GO}). The first keyterm set $K^{MeSH}_{D_{PM}}$ contains MeSH terms. MeSH (Medical Subject Headings) is a hierarchically organized vocabulary produced by the National Library of Medicine to index MEDLINE/PubMed. $K^{MeSH}_{D_{PM}}$ contains all MeSH terms occurring in the $L_{D_{PM}}(p)$ set of PubMed records associated with all proteins $p \in P^r$.

For the second keyterm set, $K^{Words}_{D_{PM}}$, we used all words (after stop-word filtering) extracted from PubMed abstracts associated with all proteins $p \in P^r$. To build the third keyterm set, $K^{Stems}_{D_{PM}}$, we reduced the words in $K^{Words}_{D_{PM}}$ to their linguistic stems, using a morphological normalization tool, called BioMorpher, which we have used previously[23]. Finally, the fourth keyterm set $K^{Terms}_{D_{GO}}$ contains terms from the $L_{D_{GO}}(p)$ set of GO annotations associated with all proteins $p \in P^r$. Notice that many of the annotations in GOA are electronically inferred (e.g. they are based on hits from sequence similarity searches or are transferred from database records). To avoid circularity in our argument we used the GOA evidence code to filter out term annotations inferred from electronic annotations (IEA), limiting our selection to those annotations assigned due to experimental evidence or published literature.

For each of the keyterm sets, we computed a protein-keyterm co-occurrence matrix where each positive entry denotes that the respective keyterm occurs in a document or annotation linked to the respective protein. The rows of the Matrix define the protein vectors for each protein $p \in P^r$ in the respective keyterm space. Table 1 shows the number of non-zero entries for each matrix and the average number of keyterms per protein in each of the four keyterm sets.

Table 1. A comparison of the four keyterm sets.

	$K_{D_{PM}}^{MeSH}$	$K_{D_{PM}}^{Words}$	$K_{D_{PM}}^{Stems}$	$K_{D_{GO}}^{Terms}$
total protein-keyterm associations	98707	560639	484072	14583
avg. keyterms per protein	27	153	132	4

3.4. *Protein Vectors and Protein Similarity*

The entry for a given protein-keyterm pair in the protein-keyterm co-occurrence matrix is a weight representing the relative importance of the keyterm for that protein. This weight is defined by multiplying a local and a global weight for the protein-keyterm pair. The local weight is the *term frequency* tf_{ik}, defined as the number of documents or annotations cited for protein p_i in SWISSPROT that are also indexed by keyterm k in publication resource D being tested.

The coefficients of the protein vectors are then scaled by a global weight to capture the relative importance of each keyterm in the space. The global weight we applied is related to the *Inverse Document Frequency* (IDF) in IR[7]. We named it *inverse protein family frequency* (IPFF) and defined it as $ipff_k = \log(\frac{N^{PF}}{n_k^{PF}})$ where N^{PF} is the total number of Pfam families in $\mathcal{C}$ and n_k^{PF} is the number of Pfam families that contain a protein with at least a document/annotation indexed by keyterm k. Finally, the protein-keyterm co-occurrence matrix W is defined by $w_{ik} = tf_{ik} \cdot ipff_k$ where row i denotes protein vector i in keyterm dimension/column k. Figure 1 depicts this process.

To measure protein similarity in keyterm space, we used the IR cosine measure[1]: given protein vectors $\mathbf{p}_i$ and $\mathbf{p}_j$ in a n-dimensional keyterm space, the cosine similarity $\sigma_{\cos}$ between them is their normalized dot product:

$$\sigma_{\cos}(\mathbf{p}_i, \mathbf{p}_j) = \frac{\mathbf{p}_i \cdot \mathbf{p}_j}{\|\mathbf{p}_i\|\|\mathbf{p}_j\|}$$

3.5. *Prediction Algorithms A*

Our first LSBIT experiments, designed to establish how well we can predict the Pfam family of proteins using the bibliome resources described above, tested two classification algorithms closely related to the k-nearest neighbor algorithm[6]. Given a protein keyterm vector $\mathbf{p}_i$ and an angle α, the first algorithm, A_α, assigns a score to each Pfam family j based on the number of proteins of that family

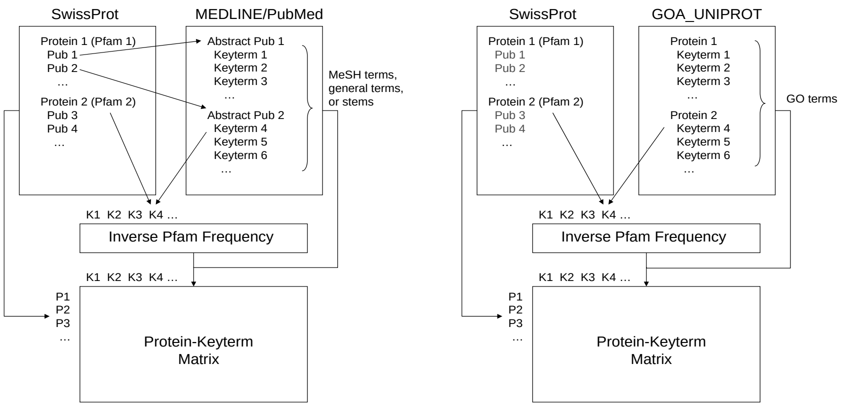

Figure 1. The process of building a protein-keyterm matrix using different linkage information sources: MEDLINE/PubMed (left) and GOA_UNIPROT (right).

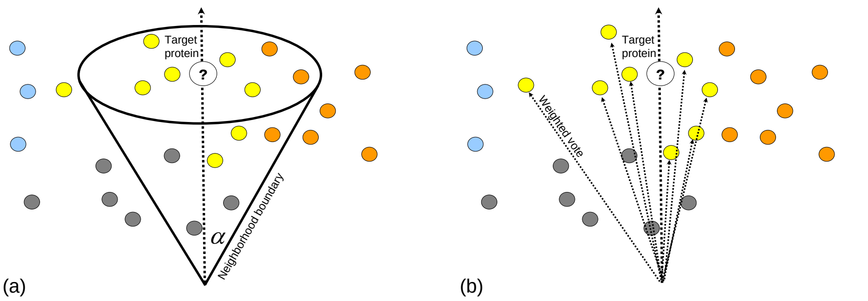

Figure 2. (a) A_α prediction algorithm: target protein neighborhood defined by the hypercone with opening angle α and centered around the target protein vector. (b) A_{WV} prediction algorithm: proteins voting in proportion to their cosine similarity to the target protein.

found in a hypercone defined by the angle α and centered around $\mathbf{p}_i$, as illustrated in figure 2(a). Thus, A_α returns a ranking of Pfam families based on this score:

$$A_\alpha : \mathbf{Pfam}_j(\mathbf{p}_i, \alpha) = |\{\mathbf{p}_k \in \mathit{pfam}_j : \sigma_{\cos}(\mathbf{p}_i, \mathbf{p}_k) \geq \cos(\alpha)\}|$$

The family with most proteins in the neighborhood is ranked first, and so forth. This algorithm is described in detail in[14,16].

A problem with the A_α algorithm is that it depends on an angle α. If α is large, unrelated proteins may be included in the neighborhood; if α is small the neighborhood may contain very few proteins or may be empty, in which case no prediction can be made. A second problem is that it is biased towards ranking

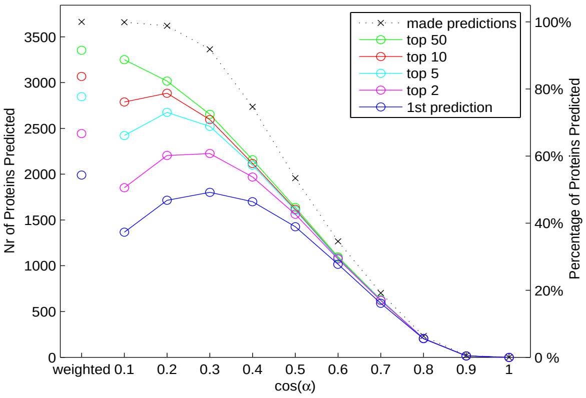

Figure 3. Prediction success using algorithms A_{WV} and A_α, and keyterm set K_{DPM}^{MeSH}.

larger families first. We have adapted A_α to deal with both these issues. In the new algorithm, A_{WV}, every protein in the space issues a "weighted vote" for its Pfam family (not just those inside a neighborhood hypercone):

$$A_{WV} : \mathbf{Pfam}_j(\mathbf{p}_i) = \frac{\sum_{\mathbf{p}_k \in \mathit{pfam}_j} \sigma_{\cos}(\mathbf{p}_i, \mathbf{p}_k)}{\sqrt{|\mathit{pfam}_j|}}$$

The weight of each protein's vote is given by the cosine of the angle between its vector and the vector of the protein being classified. In order to weaken the bias towards larger families, the family score is normalized with a division by the square root of family size. Figure 2(b) illustrates this process. A_{WV} improves on our first algorithm because it does not require a neighborhood angle to be defined in advance and it always issues a prediction for any protein vector in the space. Additionally, as we will see next, it has a higher prediction success than A_α.

4. Results: Testing $\langle A, K_D \rangle$

The two algorithms A_α and A_{WV} were tested using the four keyterm sets K_{DPM}^{MeSH}, K_{DPM}^{Words}, K_{DPM}^{Stems} and K_{DGO}^{Terms}. Figure 3 shows the prediction success of our algorithms using K_{DPM}^{MeSH} in terms of *true-positives*, i.e. the number of proteins for which the Pfam family was predicted correctly. The first entry on the x-axis (labelled weighted) corresponds to the weighted-voting algorithm A_{WV}. The remaining entries on the x-axis (labelled 0.1, 0.2, etc.) indicate the cosine of α for the A_α algorithm. The y-axis shows the number of proteins correctly predicted out of a total of $3663 \in P^r$. The black, dashed curve shows the number of proteins for which a prediction was made using A_α for various angle α. As the cosine threshold increases, the number of predictions made by A_α decreases.

A_{WV} outperformed A_α in all our tests, therefore for the other three keyterm sets, we only display results for A_{WV} summarized in Table 2. Noticeably, the three keyterm sets extracted from PubMed records performed better than the one extracted from GO annotations. This might be due to fewer GO than PubMed keyterms per protein (see table 1). Among the three keyterm sets based on PubMed, the two obtained from abstract words significantly outperform the one containing MeSH terms; the stem-based keyterms provided slightly better results than plain words.

Table 2. Prediction success for A_{WV}.

	$K_{D_{PM}}^{MeSH}$	$K_{D_{PM}}^{Words}$	$K_{D_{PM}}^{Stems}$	$K_{D_{GO}}^{Terms}$
1st prediction	54.35%	75.27%	75.89%	38.08%
top 2	66.72%	84.17%	84.22%	45.65%
top 5	77.70%	88.83%	89.30%	55.53%
top 10	83.76%	91.13%	91.48%	61.86%
top 50	91.54%	94.02%	94.40%	75.59%

5. Integrating Predictions from Different Keyterm Sets

We noticed that the sets of correctly predicted proteins using different keyterm sets do not completely overlap. Therefore, using Shannon's measure of entropy[12] we can select the lower-uncertainty class predictions from the different keyterm sets, leading to a more successful algorithm that efficiently integrates information from those distinct sources.

Let $\rho_K(\mathbf{p}_i, \mathit{pfam}_j, \alpha)$ be the probability of selecting pfam_j as the protein family predicted for protein $\mathbf{p}_i$ using keyterm set K and a neighborhood bounded by angle α. We estimate this probability as follows:

$$\rho_K(\mathbf{p}_i, \mathit{pfam}_j, \alpha) = \frac{|\{\mathbf{p}_k \in \mathit{pfam}_j : \sigma_{\cos}(\mathbf{p}_i, \mathbf{p}_k) \geq \cos(\alpha)\}|}{|\{\mathbf{p}_k : \sigma_{\cos}(\mathbf{p}_i, \mathbf{p}_k) \geq \cos(\alpha)\}|}.$$

Then, we compute the entropy of a prediction for protein i as follows:

$$H_K(\mathbf{p}_i, \alpha) = \begin{cases} \infty & \text{if } |\{\mathbf{p}_k : \sigma_{\cos}(\mathbf{p}_i, \mathbf{p}_k) \geq \cos(\alpha)\}| = 0 \\ -\sum_j \rho_K(\mathbf{p}_i, \mathit{pfam}_j, \alpha) \log \rho_K(\mathbf{p}_i, \mathit{pfam}_j, \alpha) & \text{otherwise.} \end{cases}$$

Finally, we compute the prediction uncertainty of protein i using keyterm set K, $U_K(\mathbf{p}_i)$, as the average entropy on a finite set of angle thresholds T:

$$U_K(\mathbf{p}_i) = \begin{cases} \infty & \text{if } \forall\, \alpha \in T, H_K(\mathbf{p}_i, \alpha) = \infty \\ \langle H_K(\mathbf{p}_i, \alpha) \rangle & : \alpha \in T \wedge H_K(\mathbf{p}_i, \alpha) \neq \infty. \end{cases}$$

Using the uncertainty measure, we implemented and tested a novel algorithm that integrates protein family predictions issued by each keyterm set, by selecting the lower uncertainty predictions. Let $\mathcal{K}$ be a set of keyterm sets. For $K \in \mathcal{K}$, let $\mathbf{Pfam}_j^K(\mathbf{p}_i)$ be the score assigned to protein family j when predicting protein i using keyterm set K. Then, our integration algorithm based on uncertainty, A_U is implemented as follows:

$$A_U : \mathbf{Pfam}_j^{U_{\mathcal{K}}}(\mathbf{p}_i) = \mathbf{Pfam}_j^K(\mathbf{p}_i) \text{ where } K = \operatorname*{argmin}_{K' \in \mathcal{K}} U_{K'}(\mathbf{p}_i).$$

As a baseline for comparison, we implemented a simple prediction algorithm, $A_{\langle K \rangle}$, that also integrates the predictions issued by the four keyterm systems by computing the average score $\langle \mathbf{Pfam}_j^K(\mathbf{p}_i) \rangle$ over all $K \in \mathcal{K}$. Table 3 summarizes the results obtained by these algorithms, highlighting the usefulness of an uncertainty-based method for the top predictions. Indeed, in addition to clearly outperforming $A_{\langle K \rangle}$, A_U outperforms the best results of A_{WV} with a single keyterm set (K_{PM}^{Stems}) (see table 2) for correct first and top 2 predictions.

Table 3. Prediction with combined keyterm sets.

	$A_{\langle K \rangle}$	A_U
1st prediction	70.84%	77.15%
top 2	80.02%	84.77%
top 5	87.50%	88.86%
top 10	91.35%	90.88%
top 50	95.93%	93.80%

6. Discussion and Conclusions

Our experiments show that the Pfam classification of SWISSPROT proteins is quite well inferred, independently, from the publication resources and associated keyterm sets (MeSH, GO, PubMed abstracts), we tested with the LSBIT. The publication space with associated keyterms largely captures the functional information structure represented by the Pfam classification. Moreover, we have shown that Shannon's measure of entropy can be used to integrate the predictions from various keyterm sets, resulting in an improved protein Pfam prediction algorithm. Algorithm A_{WV} always issues a prediction, which is desirable when we want to maximize the number of true-positives. However, for certain tasks it may be desirable to minimize the number of false-positives, or to use a certainty factor to express how reliable we judge a prediction to be. We are exploring the use of Shannon's measure of entropy to implement this scheme.

An interesting finding for us was that for all tested algorithms, protein family prediction is far superior with keywords extracted from PubMed abstracts than with terms extracted from GO annotations. This suggests that although GO is becoming the standard annotation resource for gene and protein annotation, PubMed abstracts, and even MeSH keyterms, are far superior as resources for literature mining. Given our results, it is fair to conclude that PubMed abstracts and MeSH terms contain more semantic and functional information to classify proteins. In future work, we will investigate what specific information is missing in the GO annotations which causes the lower performance.

Our results also show that the simple vector space model from IR is capable of well representing the semantics entailed in PubMed abstracts for protein family prediction: e.g. for 90% of proteins the correct Pfam family was among the top 5 ranked families (see table 2). In preliminary tests, we have observed that LSA improves the results only when using PubMed abstract words, and not with the other keyword sets. These results suggest that abstract keyterms have more synonymy and polysemy than MeSH and GO, but the details of that analysis are forthcoming. In future work we intend to produce working bibliome informatics tools that build up on the knowledge and algorithms of this study. We will also extend this study with additional algorithms and resources. This includes extending our algorithms by exploiting the Ontology nature of MeSH and GO with similarity measures, testing additional uncertainty-based methods, and methods based on our network analysis methodology[23,17].

Acknowledgements

We are grateful to IU's Research and Technical Services (especially Steve Simms and George Turner) for technical support. The AVIDD Linux Clusters used in our analysis are funded in part by NSF Grant CDA-9601632. This work was also supported by the Department of Energy under contract W-7405-ENG-36 to the University of California. We particularly thank Tom Terwilliger at the Los Alamos National Laboratory for the motivation to conduct this study.

References

1. R. Baeza-Yates and B. Ribiero-Neto. *Modern Information Retrieval*. Pearson Education, 1999.
2. K. G. Becker, D. A. Hosack, G. Dennis, R. A. Lempicki, T. J. Bright, C. Cheadle, and J. Engel. PubMatrix: a tool for multiplex literature mining. *BMC Bioinformatics*, 4(1):61, Dec 2003.
3. E. B. Camon, D. G. Barrell, E. C. Dimmer, V. Lee, M. Magrane, J. Maslen, D. Binns, and R. Apweiler. An evaluation of go annotation retrieval for biocreative and goa. *BMC Bioinformatics*, 6 Suppl 1:S15, 2005.
4. J. Challacombe, A. Rechtsteiner, R. Gottardo, L. M. Rocha, E. P. Brown, T. Shenk, M. Altherr, and T. Brettin. Evaluation of the host transcriptional response to human cytomegalovirus infection. *Physiological Genomics.*, 18(1):51–62, 2004.
5. M. E. Colosimo, A. A. Morgan, A. S. Yeh, J. B. Colombe, and L. Hirschman. Data preparation and interannotator agreement: Biocreative task 1b. *BMC Bioinformatics*, 6 Suppl 1:S12, 2005.

6. R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification*. Wiley, New York, NY, 2nd edition, 2000.
7. S. Dumais. Enhancing performance in latent semantic indexing, 1990.
8. W. Hersh, R. T. Bhupatiraju, and S. Corley. Enhancing access to the Bibliome: the TREC Genomics Track. *Medinfo*, 11(Pt 2):773–777, 2004.
9. L. Hirschman, A. Yeh, C. Blaschke, and A. Valencia. Overview of biocreative: critical assessment of information extraction for biology. *BMC Bioinformatics*, 6 Suppl 1:S1, 2005.
10. R. Homayouni, K. Heinrich, L. Wei, and M. W. Berry. Gene clustering by Latent Semantic Indexing of MEDLINE Abstracts. *Bioinformatics*, 21(1):104–115, 2005.
11. T. K. Jenssen, A. Laegreid, J. Komorowski, and E. Hovig. A literature network of human genes for high-throughput analysis of gene expression. *Nat. Genet.*, 28(1):21–28, 2001.
12. G. J. Klir and M. J. Wierman. *Uncertainty-Based Information : Elements of Generalized Information Theory*. Studies in Fuzziness and Soft Computing. Physica-Verlag, 1999.
13. D. R. Masys, J. B. Welsh, J. Lynn Fink, M. Gribskov, I. Klacansky, and J. Corbeil. Use of keyword hierarchies to interpret gene expression patterns. *Bioinformatics*, 17(4):319–26, 2001.
14. A. Rechtsteiner. *Multivariate Analysis Of Gene Expression Data And Functional Information: Automated Methods For Functional Genomics*. PhD thesis, Portland State University, 2005.
15. A. Rechtsteiner and L. M. Rocha. MeSH key terms for validation and annotation of gene expression clusters. In *Currents in Computational Molecular Biology. RECOMB 2004*, pages 212–213, 2004.
16. A. Rechtsteiner, L. M. Rocha, and C. E. Strauss. Clustering of protein families in literature keyword space. In *Currents in Computational Molecular Biology. RECOMB 2005*, Boston, MA, 2005.
17. L. M. Rocha, T. Simas, A. Rechtsteiner, M. DiGiacomo, and R. Luce. Mylibrary@lanl: Proximity and semi-metric networks for a collaborative and recommender web service. In *The 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2005)*, pages 565–571, Compiegne, France, Sep 2005.
18. H. Shatkay and R. Feldman. Mining the biomedical literature in the genomic era: An overview. *Journal of Computational Biology*, 10(6):821–856, 2003.
19. SIB/EBI. UniProt/Swiss-Prot. http://www.ebi.ac.uk/swissprot/, 2004.
20. E. L. Sonnhammer, S. R. Eddy, and R. Durbin. Pfam: a comprehensive database of protein domain families based on seed alignments. *Proteins*, 28(3):405–420, Jul 1997.
21. P. Srinivasan. MeSHmap: a text mining tool for MEDLINE. *Proc AMIA Symp*, pages 642–646, 2001.
22. L. Tanabe, U. Scherf, L. H. Smith, J. K. Lee, L. Hunter, and J. N. Weinstein. MedMiner: an Internet text-mining tool for biomedical information, with application to gene expression profiling. *Biotechniques*, 27(6):1210–1214, Dec 1999.
23. K. Verspoor, J. Cohn, C. Joslyn, S. Mniszewski, A. Rechtsteiner, L. M. Rocha, and T. Simas. Protein annotation as term categorization in the gene ontology using word proximity networks. *BMC Bioinformatics*, 6 Suppl 1:S20, 2005.