A Probabilistic Information Retrieval Approach to Medical Annotation in SWISS-PROT

Pavel B. Dobrokhotova(1) Cyril Goutteb(1), Anne-Lise Veutehay, Eric Gaussierb

aSwiss Institute of Bioinformatics, Switzerland, Pavel.Dobrokhotov@isb-sib.ch
bXerox Research Centre Europe, France, Cyril.Goutte@xre.xerox.com
Both authors contributed equally to this work

Abstract

The goal of medical annotation of human proteins in Swiss-Prot is to add features specifically intended for researchers working on genetic diseases and polymorphisms. For this purpose, it is necessary to search through a vast number of publications containing relevant information. Promising results have been obtained by applying natural language processing and machine learning techniques to solve this problem. By using the Probabilistic Latent Categoriser on representative query sets, 69% recall and 59% precision was achieved for relevant documents. This classifier also rejected irrelevant abstracts with more than 96% precision. Better linguistic pre-processing of source documents can further improve such computer approach.

Keywords: Swiss-Prot, database annotation, document filtering, text mining

1. Introduction

In the last years, there has been a tremendous increase in the amount of human genome data, especially with regard to the molecular basis of genetic diseases. Every week, new discoveries that link one or more genetic diseases to defects in specific genes are made. Swiss-Prot [1] is a curated protein sequence knowledge base that strives to provide a high level of annotation, a minimal level of redundancy and high level of integration with other biomolecular databases. Following the achievement of the human genome project it has been gradually enhanced by the addition of features specifically intended for researchers working on genetic diseases and polymorphisms. This is referred to as medical annotation and deals almost exclusively with human protein entries. The goal is to annotate all the genetic variants of a protein, with the exception of nonsense and frameshift amino-acid changes, because they disrupt the protein sequence completely. Searching for this kind of information is a very tedious task for several reasons:

- Annotation must be complete, which implies retrieving a large number of documents.
- Indirectly related articles are retrieved and read, so as not to miss any description of missense (change of one amino acid into another) and/or in-frame (addition/deletion of an amino acid) sequence changes.
- Annotation must be correct, which implies checking manually the variants’ names, positions and amino-acid changes.

For all these reasons, medical annotation is particularly labour-intensive and can benefit from advanced computing techniques, such as Text Mining.

Building a Medical Annotation Tool

Medical annotation has some distinct advantages for “computerisation”: the search space is small and clearly defined, i.e. only human proteins; sufficient background knowledge is available (the official gene name and synonyms); there is some structured information in
the articles, such as mutation points. The main difficulty is that the collected information should be very comprehensive. This implies that simple data filtering, where papers deemed irrelevant are sifted out, may result in missing some important but misjudged documents. Therefore, we prefer to consider all references returned by PubMed for a given protein and re-rank them in a way that is more relevant to curators, by forcing important documents to the top of the list. Curators can then decide themselves when to stop processing the list. We performed such re-ranking using a probabilistic classifier, as they have been shown to be successful with similar problems[2][3]. Our classifier assigns documents to one of three categories: relevant, irrelevant and unsure documents, and ranks them according to their significance in each category.

2. Materials and methods

Dataset

A total of 32 human genes has been chosen from a list scheduled for medical annotation. The corresponding 2188 abstracts were retrieved from PubMed and classified manually by Swiss-Prot curators: 15% were assigned to the “Good” class (relevant for medical annotation); 70% to “Bad” (irrelevant) and 15% to “Unclear” (does not contain enough information to assess relevance confidently). The number of retrieved documents per gene ranged from 2 to 258. The proportion of “Good” was also highly variable and ranged from 1% to 82%. These figures accurately represent the real data diversity encountered by curators.

Document processing pipeline

![Diagram of document processing pipeline](image)

**Figure 1 - Document processing flow**

The documents from the dataset are processed as follows (Figure 1). Titles and abstracts are morpho-syntactically analysed with Finite State Transducers (FST), up to lemmatisation. Then a simple disambiguation is performed. For words with English and biological meanings, the latter alternative is retained. For other types of ambiguities (e.g. between English words, abbreviations), the highest-ranking meaning is taken. These assumptions, although simplistic, have proven acceptable for the probabilistic classifier. In the normalisation step, all words marked as “biological” are checked against dictionaries compiled for the BioMiRe project [4] so as to replace all gene/protein synonyms by their
official names. In addition, the name of the particular gene used for each query is replaced by a generic token “gene_req” in order to minimise differences between queries. Furthermore, mutations indicated by a pattern AaaNBBb (where Aaa and Bbb are the 1- or 3-letter IUPAC-IUB amino acid codes and N is a natural number) are replaced by the generic token “point mutation”. Finally, all numbers that are not part of a word are replaced by a generic token “NUM”.

At this point, the pipeline splits into two branches, which correspond to single-term and compound-term indexes. In the first branch, tokens are filtered according to their part of speech (POS) to remove any empty words (prepositions, auxiliary verbs, cardinals, etc.). Further filtering is then performed by removing tokens smaller than 3 characters or without any letters. Finally, lemmatised forms of remaining words are extracted and counted to produce a single term index (IS) for each document. Furthermore, journal names are added to the index with a token “JournalName”. The second branch extracts consecutive two-token items which match common patterns such as noun+noun, adj+noun, or adv+noun. Extracted terms are counted to produce a compound-term index (IC). Both indexes are instances of the so-called bag-of-word representation of documents, which consists of a list of single/combined terms and their frequency in the document. They may be used either individually or combined as input to the probabilistic classifier in both the training or classification phases.

Probabilistic Classifier

We use the Probabilistic Latent Categoriser (PLC) described in [5], an extension of Probabilistic Latent Semantic Analysis [6] to supervised classification. We model a collection of pre-processed documents using a generative mixture model of co-occurrences of terms \( t \) in documents \( d \):

\[
P(t, d) = \sum_{\alpha} P(\alpha)P(d|\alpha)P(t|\alpha)
\]

(1)

The class variable \( \alpha \) runs over the class labels, e.g. from 1 to \( N \) for \( N \)-class classification. The parameters are: \( P(\alpha) \) the class probability, \( P(d|\alpha) \) the probability that a document belongs to class \( \alpha \) and \( P(t|\alpha) \) the probability of generating term \( t \) in that class. These parameters are estimated from empirical counts using a maximum likelihood (ML) approach [5]. In order to assign a class to a new document \( d_{\text{new}} \), we calculate the posterior \( P(\alpha|d_{\text{new}}) \propto P(d_{\text{new}}|\alpha)P(\alpha) \), where \( P(d_{\text{new}}|\alpha) \) is again estimated by ML, via the expectation maximisation algorithm [5][7].

Documents in the studied dataset belong to one of three classes: “Good”, “Bad” or “Unclear”. Therefore we trained different models, depending on the treatment of the “Unclear” documents: a three-class model: “Good”, “Bad” and “Unclear” classes; two binary models: “Good” vs. “Bad or Unclear” and “Good or Unclear” vs. “Bad”.

Document Re-Ranking versus Document Filtering

Our aim is to provide curators with more relevant documents upfront, in order to speed up the selection of references that should be included in the database. As PLC gives the probability of a document belonging to a given class, we extend our system by re-ranking the documents inside three zones/classes with different levels of relevance:

- “Good” zone, where all documents should be relevant (high precision);
- “Unclear” zone, catching most of the remaining relevant documents (high recall);
- “Bad” zone, where all of the documents should be irrelevant (high precision).

In each zone, documents that we believe are most relevant (or least irrelevant) should appear at the top. The three-class classifier naturally gives such a division. As an alternative, we implemented a two-level classification using a cascade of the “Good or
Unclear” vs. “Bad” and the “Good” vs. “Bad or Unclear” classifiers. Let us define: $P_{GU} = P(\text{“Good or Unclear”}|d)$ and $P_B = P(\text{“Bad”}|d)$ with the first classifier; $P_G = P(\text{“Good”}|d)$ and $P_{UB} = P(\text{“Unclear or Bad”}|d)$ with the second classifier. The cascaded assignment rule becomes:

1. If $P_{GU} < P_B$, assign to “Bad” (with score $P_B$);
2. else, if $P_G > P_{UB}$, assign to “Good” (with score $P_G$);
3. else, assign to “Unclear” (with score $P_G$).

The first two rules aim at ensuring high precision in the “Bad” and “Good” zone, while the remaining default assignment aims at enforcing high recall in the “Unclear” zone.

**Performance Evaluation**

In order to provide an unbiased evaluation, the collection was first split into 5 blocks. All methods are run on four of the blocks and evaluated on the left-out block in a “cross-validation” fashion. The blocks were chosen so as to cover roughly the same number of genes (6 or 7). The high variability in the number of documents per gene led to large differences in the number of documents per blocks, from 364 to 609. The proportion of “Good” and “Bad” documents remained roughly similar over blocks, though, in the range of 8-20% “Good” and 65-77% “Bad”.

We used traditional Information Retrieval (IR) performance measures: precision, recall and F-score, as well as averaged precision-recall curves. Precision is the ratio of relevant documents correctly identified as relevant (true positives TP) over all documents identified as relevant (TP plus false positives FP); Recall is the ratio of relevant documents correctly identified as relevant (TP) over all relevant documents (TP plus false negatives FN). The $F_\beta$-score is the (weighted) harmonic average of precision and recall ($\beta > 1$ favours recall, $\beta < 1$ favours precision). In short:

$$p = \frac{TP}{TP+FP}, r = \frac{TP}{TP+FN} \text{ and } F_\beta = \frac{(1+\beta^2)p r}{\beta^2 p + r}.$$ 

Like most IR methods, our re-ranking technique outputs a sorted list of documents. We wish to have most relevant documents located at the top of the list, while the irrelevant documents remain at the bottom. To evaluate this, we calculate the precision of relevant documents at various points of recall, i.e. for 10%, 20%, ... 100% relevant documents retrieved. Plotting these points gives a “precision-recall” curve, the higher the curve, the better.

### 3. Results and Discussion

**Table 1 - Overall results for two models**

<table>
<thead>
<tr>
<th>Zone: IS+J</th>
<th>Two-level IS+J</th>
<th>Three-class IS+J</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\beta$</td>
<td>$p$</td>
</tr>
<tr>
<td>Good</td>
<td>½</td>
<td><strong>58.89</strong></td>
</tr>
<tr>
<td>Good+ UNC</td>
<td>2</td>
<td>48.95</td>
</tr>
<tr>
<td>Bad</td>
<td>½</td>
<td><strong>96.26</strong></td>
</tr>
</tbody>
</table>

We first evaluated the individual classifiers. To take into account the requirements for the three zones expressed above, we calculated the precision/recall/$F_\beta$ for the relevant documents 1) in the “Good” zone, 2) in the “Good” and “Unclear” zones, and 3) for the irrelevant documents in the “Bad” zone. The two-level classifier using single term index only and journal names obtains the overall best performance (Table 1, left column). About 84% of all relevant documents are ranked in either “Good” or “Unclear”, and 59% of the “Good” zone are relevant, while 96% of the “Bad” zone are irrelevant.
Although we present detailed results for two models only, we actually performed extensive experiments to compare the influence of different factors. Our conclusions are:

1. The two-level classifier is more efficient than the three-class classifier;
2. Linguistic pre-processing increases recall at the cost of a slight precision penalty, leading to small (1-2%) but consistent overall performance improvements;
3. Adding journal names yields even smaller but still consistent improvements;
4. Adding combined indexes has negative or no effect. This may be due to the inability to tune their weight in the combined IS+IC index.

A comparison between different classifiers is shown with the precision-recall curves (Figure 2, left). Even though the two-level classifier using single term index and journal tokens (2L IS+J) is not superior for all recall points (most notably for the first one, see Figure 2, left), it yields the best overall performance. When every other relevant document has been retrieved (50% recall), over three quarters (77%) of all retrieved documents are relevant. The methods using no linguistic pre-processing (2L baseline) or combined index (2L IS+IC+J) lead to the worst average performance, while the three-class method (3C IS+J) gives intermediate performance.

We also compared the probabilistic classifier to two other techniques (Figure 2, right): the default chronological order as returned by PubMed and a two-level classifier with Support-Vector Machine (SVM), a state-of-the-art text classification technique [8]. We used SVMlight (http://svmlight.joachims.org) with default parameters except for the cost factor, set to the negative/positive dataset distribution ratio, and a linear kernel (polynomial kernel provided no improvement). With identical NLP pre-processing, both PLC and SVM methods yield comparable results and demonstrate a large improvement over the default chronological order. Note that PubMed yields an average precision around 40%, much larger than the ratio of relevant documents in the entire collection. We attribute this fact to a combination of the internal PubMed ranking, and the fact that recently published documents tend to be preferred for Swiss-Prot annotation.

![Figure 2 - Left: Probabilistic classifier with various pre-processing and models; Right: Comparison with the standard PubMed return and SVM classifier](image)

4. Conclusion

The results obtained in this work suggest that the natural language processing techniques combined with Probabilistic Latent Categoriser can be successfully applied to document ranking problems in the biomedical field. With the best classifier 69% recall and 59%
precision on classification of articles relevant to the annotation can be achieved. When including the “Unclear” zone, the recall reaches 84%. The classifier also accomplishes a high level of precision (96%) on detection of irrelevant publications. Overall, after document re-ranking all relevant documents are found in the upper 40% of the list. At present, the work continues along two main directions. First, the current classifier will be integrated into a graphical query interface and field-tested by curators to better assess its performance. Second, investigations will continue on the classifier chain, particularly on the disambiguation and normalisation parts and their impact upon final classification. This includes usage of better biological dictionaries and term recognition, use of more complex mutation patterns, etc. Also, a better weighting scheme of the various components in the final bag-of-words (IS vs. IC; regular word vs. journal name, etc.) will be devised.

5. Acknowledgements

We are grateful to Livia Famiglietti, Raffaella Gatto and Arnaud Gos of the Swiss-Prot medical annotation team for providing the dataset of abstracts. This work was supported in part by a grant from the Sibelius project (INRIA/ISB).

6. References


