Finding Gene Function using LitMiner

Berry de Bruijn and Joel Martin

Institute for Information Technology
National Research Council of Canada
berry.debruijn@nrc.gc.ca, joel.martin@nrc.gc.ca

Abstract
NRC (National Research Council, Canada) submitted 2 sets of results for the primary task in the TREC Genome track. The systems that generated these results were tuned primarily to achieve very high recall (above 90%) and secondarily to minimize the number of documents retrieved. Both submitted sets were the outputs of automatic systems (non-interactive, non-supervised) with a modular architecture.

The TREC evaluation confirmed that recall for both submissions was extremely high: 543 out of 566 target documents (0.9594) were returned. In addition, these systems returned far fewer documents than were allowed by the genomic track rules. They returned an average of 196 documents per query across the 50 queries, with a median value of only 100 documents.

For the first submission, the system was entirely based on Information Retrieval techniques, tuned to achieve very high recall and fair precision. Averaged precision was 0.3941 for the first submission. This first submission ranked third out of 49 runs submitted by all participants.

For the second submission, reranking was done based on the outcome of an information extraction module, tuned towards the task of identifying gene function papers. This module identified 539 documents as highly promising; 121 of these turned out to be target documents, 418 weren't. All in all this caused the averaged precision to drop slightly to 0.3771 - contrary to our expectations. This second submission ranked fifth out of all 49 runs.

1. Introduction

Scientists reviewing literature in their field often hope for exhaustive searches that return all the relevant documents. The cost of missing an important document is high, so less than perfect precision is accepted from real-world (less than perfect) systems if close to full recall is still guaranteed. Of course, since the scientist would have to scan every article returned by the search, a system returning 100 results is far better than one returning 1000. This is the type of system we envisioned when taking up the TREC task.

After a genomics 'pre-track' in 2002, a full genomics track was added to the TREC setup for the 2003 edition. This development is in full agreement with the strong attention that Information Extraction from biomedical literature has recently received (see for instance De Bruijn and Martin, 2003). The genomics track presents interesting issues to the text retrieval research community because of the combination of working with large-scale document collections and the specifics of the application field with its esoteric jargon.

The National Research Council (NRC) is the Government of Canada's premier organization for research and development. For a number of years, researchers at its Institute for Information

Technology have been working on language processing technologies, including methods and tools to process biomedical literature. Multiple technologies are now being bundled into an integrated toolbox for literature access and management, named LitMiner.

This paper gives an overview of the architecture that we used for our TREC-genomics submissions, including a discussion on how such a high recall was achieved. It includes a further analysis of those queries where performance was disappointing or under par. Design and performance of the information extraction module are discussed.

2. System

2.1 system architecture

For storage, a MySQL database was used under Linux/Intel586. The documents were stored in one table, and an index on terms to document identifiers (PMIDs) was created in a separate index table. The index contained lists of PMIDs for all non-stop words, along the position of the word in that text. A title word index was included in a separate column in the same table; other columns contain the word frequencies and document frequencies. Also stored in the index table were entries for the complete RN and MeSH terms which can be multiple word terms.

The NRC system finally consisted of 7 modules, working in sequence. The modular architecture is sketched in figure 1. The functions of the various modules is as follows:

[1] *basic retrieval:* database/index lookup of all articles containing one or more of the query gene names verbatim

[2] *morphological term expansion:* based on a number of rules, variations of the original gene names are generated and articles containing these expanded query terms are added to the retrieved set

[3] *acronym disambiguation:* this module removes those articles from the result set where the letter combination clearly has an incorrect meaning, for instance, "MVP" should mean "major vault protein" and not "mitral valve prolapse".

[4] *relevance feedback:* this is done on the RN field of the Medline records; records are added to the retrieved set if they contain RN terms that are relatively 'overrepresented' in the set so far. Such a relevance feedback (RF) loop is done twice.

[5] *organism matching:* since the target organism is given for each query, only articles are returned that actually mention the organism name.

[6] *reranking:* articles are reranked based on the occurrence of more gene aliases or more significant ones; aliases occurring in the title give an extra boost.

[7] *function phrase finding:* (second submission only) - documents that score high on a high-precision 'function phrase finding' test get an additional boost up.

Modules 1,2 and 4 are designed to achieve optimal recall, while modules 3, 5, 6 and 7 are designed to improve precision. The various modules are described in detail in the next section.

2.2 Technical description of each of the modules

2.2.1 query term retrieval

Basic document retrieval / index lookup: retrieves the document identifiers for those documents that contain that word (for single-word terms) or that phrase (for multi-word terms). Matching was case-independent and on full, untruncated terms. Since stop words were excluded from indexing, query terms that double as common English stop words (such as AND) would not pose a problem. Stop words were allowed to be part of query phrases.

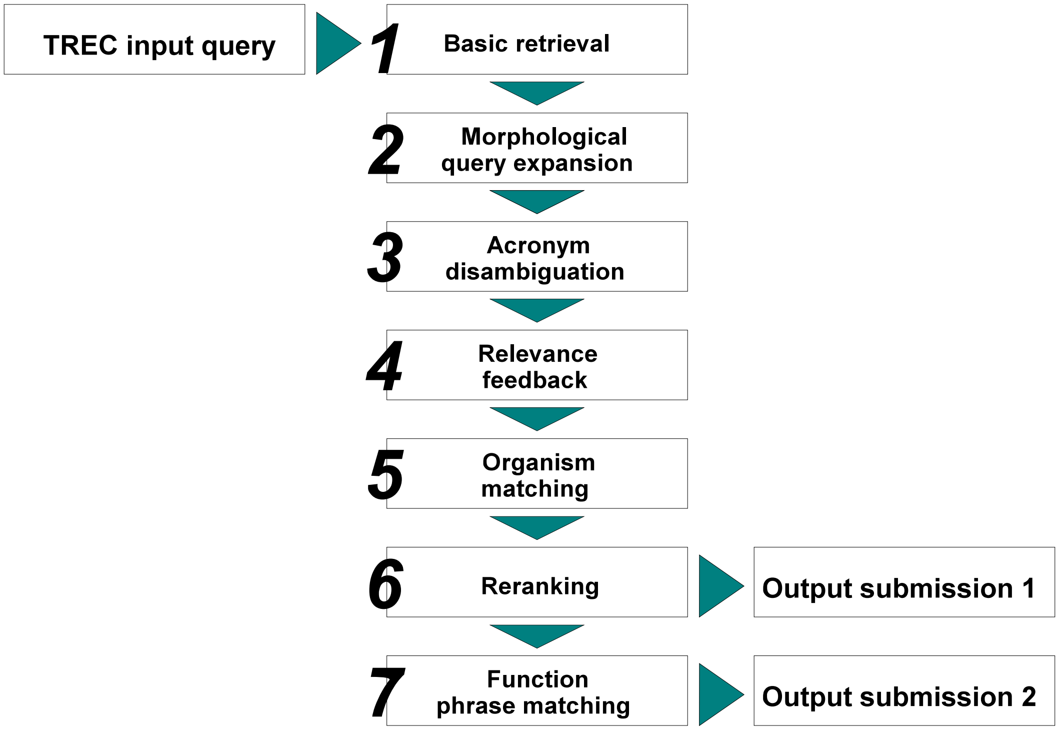

Figure 1: architecture of the NRC retrieval system

2.2.2 morphological term expansion

It was observed by us and our domain experts that authors do not always use the preferred molecule name, or in other words, the list of aliases is not complete in listing all possible molecule names that authors decide to use. Some of the used names are minute variations on one of the official symbol/alias names and can be found by creating a limited number of morphological variations on the official names. This module includes handcrafted rules that make these variations. A complete list of the rules is available from the authors; some examples of morphological variations that this module deals with, are:

'ATF2' could be referred to as 'ATF-2', 'ATF 2', 'ATF-II'
'PPARG' could be referred to as 'PPAR Gamma'
'FGFR' could be referred to as 'FGF receptor'.

2.2.3 acronym disambiguation

Acronyms are fairly likely to be ambiguous. For instance, using the query term "MVP" with the intention retrieve documents on major vault protein will also return irrelevant documents where the same acronym stands for 'mitral valve prolapse', 'microvascular pressure', or ‘Midwifery Ventouse Practitioners’. This module disambiguates the meaning by first creating a list of possible meanings for that acronym from Medline documents, in a routine analogous to e.g. Pustejovsky (2001), or Schwartz and Hearst (2003). For each of the meanings in the list, it is determined whether the meaning is correct or incorrect in the context of the query. Then documents under consideration are tested for the presence of any of the possible acronym meanings, and if only an incorrect meaning is found in a document then it is excluded from the retrieved set.

2.2.4 relevance feedback

After retrieval with the query terms, and after disambiguation, an unsupervised relevance feedback step is done (twice). For all documents thus far retrieved, the values for the RN field are retrieved. The RN field in the Medline data lists Registration Numbers to chemicals and substances involved in the study, including Enzyme Commission numbers and names. Then other (so far unretrieved) documents are added to the retrieved set if they contain RN terms that are strongly represented in the retrieved set. Per feedback iteration, the number of terms used was at most 10 and at least five provided that each added term occurred in at least 20% of the previously retrieved documents. The number of feedback iterations was set to two. These parameters were set based on runs with the training material.

2.2.5 organism matching

This step rejects articles where the correct gene might be discussed but where it relates to another organism. For that, a synonym shortlist is used, containing the synonyms for the most frequently researched organisms (human, drosophila, mouse, rat, cow, yeast, zebrafish, e-coli, c. elegans and xenopus). If the organism from the query is not referred to in the retrieved article then that article is discarded.

2.2.6 reranking

A TF*IDF weighting scheme is used to rerank the retrieved articles, so that articles that contain characteristic query terms will end up higher in the results list. Articles that mention multiple query terms get an additional boost. Articles where the title contains query terms get an additional boost as well.

2.2.7 phrase matching and boosting

A phrase matching module was used to identify sentences likely to include descriptions of gene function and boost those documents up in the rankings. The module was trained with a supervised machine learning method on training data, while the test phase took place without supervision. In this method, seed sentences known to be GeneRIFs from the training material were automatically generalized to include similar phrases that appeared in other abstracts. The expansion included identifying words that could be substituted in the sentence and the gradual shortening of the phrase to increase the number of abstracts in which it appeared. The specific algorithm used produced sentence identification with very high precision in the training set. When a sentence or phrase was identified in a title or an abstract, that abstract could be promoted in the results ranking. This act of promotion could lead to large errors. To prevent that, the system further restricted promotion to ensure high precision. This promotion was only used when the phrase was in the title of the article AND clearly contained a reference to the gene.

3. Experiment

3.1 Task

The aim of the competition in the Primary Task of the Genomics track was to retrieve documents that contain a description of the function of a gene, given the gene names and the organism under consideration. Result lists should have the (more) relevant documents at the top and any less/not relevant documents near the tail. Result lists can contain at most 1000 documents per query.

3.2 Material

The document set consisted of 525,938 article abstracts, or one year's worth (2002), from

Medline/PubMed. These are article citations from every kind of biomedical discipline. The abstracts or citations or records contain various fields including title, abstract, author names, affiliation, publication info (journal name, issue), category terms from the Medical Subjects Headings (MeSH) hierarchy, and RN entries. The RN field contains Registration Numbers and official names of chemical substances, as well as official Enzyme Commision names and numbers of molecules involved in the study.

Fifty queries were supplied in a training set and another fifty in a test set. The test set was kept in isolation away from all system development and all developers (conforming to the competition rules). The training set was used to tune system parameters and make other performance decisions. No other data source was used (such as LocusLink, which was specifically off-limits, or Gene Ontology), with the exception of a background collection of older Medline abstracts for the Acronym Disambiguation module (more about that in section 3.3). The training material as well as the test material originated from NCBI's LocusLink database, specifically the GeneRIF field (RIF = Reference in Function) and the nomenclature field (for the query terms). Each query contained a (TREC) query number, a LocusLink identifier (not used), the organism, and a number of gene identifiers, including the official symbol name, alias names, and product names.

For the training queries, the answers were available, for the test queries, answers were made available after the submission deadline. The number of target documents per query ranged between few (0) to many (53) in the training set, median=4, average=5.83. In the test set, the range turned out to be 2 to 66, median=7, average=11.32.

3.3 Evaluation metrics

Systems were evaluated with the common Information Retrieval metrics, as well as a metric named Averaged Precision: at each point in the results list where a target document is positioned, the precision is calculated and averaged over the number of target documents. For unretrieved target documents, precision=0.

4. Results and discussion

4.1 Evaluation of the entire system

Submission 1, overall performance: this was the submission as produced with the system without module 7 (the phrase searching / document boosting module). This system returned 9824 documents over the 50 queries; recall was 543 target documents found out of 566 target documents max, or 95.9% recall. Average precision for all relevant documents (averaged over queries) was .3941. The average number of documents returned per query was 196.5, but the distribution over queries was very skewed and the median number was 100 documents per query.

Submission 2, overall performance: this system included the phrase searching / document boosting module. The module identified 539 documents as highly promising, 121 of these were indeed target documents while 418 were not. For this system, the submission-1 results were only re-ranked so the returned set remained 9824 documents with 95.9% recall. Since quite some non-target documents got boosted, the average precision score dipped to .3771.

The results for submissions 1 and 2 are summarized in table 1.

With a median number of 100 documents per query and a recall of nearly 96%, exhaustive (not exhausting) searches are realistic with this system. Scientists often want to find all the relevant documents without having to discard too many false positives. The NRC system successfully restricts the number of documents returned while maintaining a high recall. The NRC system achieved a very competitive score in the TREC competition, falling just short of matching the highest scores from one other participant.

One explanation of the lower score for our second submission, is the sparsity of GeneRIF's. The module used for that submission tried to capture characteristics of function descriptions. But not every statement of a gene's function is currently a GeneRIF. Different systems that find the same document will likely favour different valid statements of a gene's function and will produce different measures of precision based on GeneRIFs. Unlike the estimated measure for recall, an estimate of precision based on a small sample of the true documents will not necessarily be a reliable estimate of the precision across the entire population. Only one in four of the documents that were promoted by the Function Phrase finding module in our second system, were in fact GeneRIF's. That does not mean that three in four are false positives, only that we do not know their true status.

The performance of this phrase matching module was strikingly different between the training and the test data sets. Besides the problem of a sparse gold-standard, it is also possible that the learning method overfitted the training data. Only the GeneRIFs in the training set were used to infer the structure of phrases indicating function. As a result, those structures may have worked differentially well when applied to reranking training documents.

In addition to a solution to the sparsity of the GeneRIFs, we would like to propose another metric to assess the recall. For one example, the average precision could be averaged over the reading cost to give a benefit per unit of effort. Another measure would divide the recall (a measure of benefit), by the number of documents returned (a measure of the cost). Although, the NRC systems would do well on such measures, other systems might also do well by deleting the documents beyond the first 100. This modification would only produce a fair comparison if the system itself is able to determine that the number of documents should be 100.

Table 1: performance of the system and its separate components

System (or modules included)	**Retrieved (before capping at 1000)**	**Recall**	**averaged precision**
nrc1 (modules 1-6)	9824 (9837)	543/566 = 95.9%	.3941
nrc2 (modules 1-7)	9824 (9837)	543/566 = 95.9%	.3771
module 1 +6	13975 (25737)	433/566 = 76.5%	.2051
modules 1+2 +6 *(morph. query expansion)*	17466 (34188)	511/566 = 90.3%	.2355
modules 1+4 +6 1 loop relevance feedback 2 loops rf 3 loops rf	 16086 (28612) 17370 (32616) 18540 (34263)	 479/566 = 84.6% 495/566 = 87.5% 498/566 = 88.0%	 .2198 .2220 .2198
modules 1+2+4 +6 (mqe + 2 loops rf)	20813 (40692)	546/566 = 96.5%	.2397
modules 1+2+3+4 +6 (+ acronym disambiguation)	20636 (40185)	546/566 = 96.5%	.2399

4.2 Evaluation of each of the modules

Additional runs were done with some modules switched on and others switched off, to determine the separate contributions of the modules. The results of these runs can also be found in table 1.

- *Basic retrieval* with most modules switched off: only query term retrieval is included, and reranking is applied. This retrieves 25737 articles, but if the retrieved set is capped at 1000 articles per query ('cap-1000', as per TREC guidelines), 13,975 articles remain. Recall is 76.5% (433 target documents out of 566), with .2051 average precision. These measures confirm that basic retrieval alone does not cut it.

- *Morphological query expansion* increases the size of the retrieved set to 34188 articles or 17466 after cap-1000. Recall improves to 90.3% (511 documents), average precision is .2355. The strong increase in documents retrieved indicates that many irrelevant documents are also added. In this stage of the process, where high recall is the main objective, that is less important.

- *Relevance feedback* (RF) applied as an alternative to morphological query expansion comes close to the same scores but not quite. One RF loop retrieves 28612 documents (16086 after cap-1000); recall is 84.6% (479 documents); average precision is .2198. A second loop increases recall to 87.5% (495 documents) with 17370 retrieved (32616 before cap-1000); average precision .2220. A third RF loop does very liitle more: 88.0% recall (498 documents) with 18540 retrieved (34263 before cap-1000) and average precision of .2198. This confirms the decision that was based on the training material that two RF iterations would be the best setting.

- Relevance feedback did complement morphological query expansion. With both modules in place, the recall was near to perfect with 96.5% (546 documents) with 20813 retrieved (40692 before cap-1000); average precision here was .2397.

- *Acronym disambiguation* did not help much on the test material. While this module did oust 507 documents compared to the previous setting, with no incorrect discardings so no loss in recall, most of these ousted documents were at the tail of the document rankings anyway and wouldn't have survived the cap-1000 step, or wouldn't harm the average precision. While the overall improvement was small, this module did prove very useful in a few cases - including some cases in the training material. Since it this module is intuitively appealing and wasn't hurting performance we decided to leave it in.

- *Organism matching* proved a powerful method to drastically limit the number of retrieved items while doing very little harm to recall. This module cut the size of the retrieved set in four - from 40185 to 9837 documents. Among the discarded abstracts were a mere 10 target documents. After cap-1000, the final result set was 9824 with no further loss of precision. This step gives the average precision a terrific boost: from .2399 without the module to .3941 with the module. Of the ten discarded target documents, eight mention a different organism than the target organism and make a dubious or poor GeneRIF. One more document lists a broader organism category ('mammalian') rather than the target document (human). Finally, one article did not list an organism, but the Medline entry for that article has been updated after the TREC data was collected, and currently does list 'human' in the MeSH field.

4.3 Failure analysis

In the following section, we analyse a number of queries where our system returned poor results.

Query 4: *"guanine nucleotide binding protein (G protein), alpha activating activity polypeptide, olfactory type".* The system retrieved 984 documents with both target documents retrieved, in ranks

18 and 95 (submission 1), or 54 and 125 (submission 2). The results set after module 1 had 2 documents, including one of the two target documents. The morphological query expansion module included a rule that would cause the phrase “G-protein” to be used as a query term in itself, while this would be too general a query term to be appropriate in this case. A system with module 2 disabled would have retrieved five documents with total recall and .5 average precision on this query. Disabling the single rule that caused the over-generalization showed harmful to the performance on other queries.

Query 22: *“arginine vasopressin”.* This is a fairly frequently researched molecule and many top retrieved articles seem on-topic and relevant, albeit not strictly target documents. This might be a case where the evaluation metrics undervalue the results.

Query 41: *“CASP8 and FADD-like apoptosis regulator”.* This query includes three query terms that could double as common English words: ‘FLAME’, ‘FLIP’ and ‘CASH’. These terms contaminate the result set and even though 100% recall was achieved, precision suffered dramatically. An added rule that would prohibit common English words from being used as query terms would not help: on that query, recall dropped to an unacceptable 0.1 (1 retrieved, 10 targets). Allowing English words as long as capitalization of that word is uncommon (a capital letter somewhere after the first letter of the word) did reduce the number of retrieved documents from 323 to 101, kept the recall on 10/10 and raised the average precision from .1502 to .2972. That rule, however, caused a severe degradation on two other queries from the test set (26 and 27) and on one in the training material (14). With no major improvements on other queries, it would give an overall worse performance.

Query 49: *“T-cell receptor alpha chain”.* This query gave a failure of a different kind: even though a limited number of articles was retrieved (80) and two target documents were returned in positions 4 and 12 of the result list, five more target documents were not retrieved. That made this query the only one where recall for our system was <50% and it accounted for 5 out of 23 missed target documents over the entire task.

These failure analyses and other observations gave us no reason to drastically revise the overall architecture or details of the design of the system.

5. Conclusions

The LitMiner system used for the primary TREC-genomics task, was almost entirely based on established Information Retrieval techniques. It performed very well in absolute terms as well as in comparison with other systems in this TREC track. The two submissions from LitMiner ranked third and fifth out of 49 submitted runs by all participants.

We designed our system in such a way that an early stage would return a limited document set with high recall and fair precision. The second stage would be allowed to be more computationally expensive if it would be capable of increasing precision while retaining the high recall. We must admit that the second stage performed less well than we expected. This has to do with the slightly vaguer definition of what a GeneRIF would stand for. Future work will explore the use of additional training examples and hopefully more effective information extraction.

On the other hand, the first stages succeeded very well in getting the high-recall results while limiting the size of the result set. In fact, searching literature with a >95% recall in a typical result set size of a mere 100 documents provides a very practical tool for biologists.

While interactive systems are planned to be part of future TREC-genomics tasks, the setting of this year's task was a static environment. It differs from the regular, interactive search environment that characterizes our search toolbox LitMiner. Supervised relevance feedback techniques and result navigation tools are likely to add power to the current design and allow the user to quickly home in on the desired results.

Acknowledgments

We would like to thank Brandon Smith, Roy Walker, Hung Fang, Maria Moreno, and Annie Law from the NRC Institute for Biological Sciences, as well as Lynn Wei and Jeffrey Demaine (NRC-IIT), for their input during our joint meetings and on-line.

References

De Bruijn B and Martin J: Getting to the (c)ore of knowledge: mining biomedical literature. Int J Med Inf 2002 Dec;67(1-3):7-18.

Pustejovsky J, Castano J, Cochran B, Kotecki M, Morrell M: Automatic extraction of acronym-meaning pairs from MEDLINE databases. Medinfo 2001;10(Pt 1):371-5.

Schwartz AS, and Hearst MA: A simple algorithm for identifying abbreviation definitions in biomedical text. Pac. Symp. Biocomput. 2003; 451-462.