Improving Search Results in Life Science by Recommendations based on Semantic Information

Christian Colmsee[1], Jinbo Chen[1], Kerstin Schneider[2], Uwe Scholz[1], Matthias Lange[1]

[1]Department of Cytogenetics and Genome Analysis
Leibniz Institute of Plant Genetics and Crop Plant Research (IPK) Gatersleben
Corrensstr. 3
06466 Stadt Seeland, Germany
{colmsee,chenj,scholz,lange}@ipk-gatersleben.de

[2]Department Automation / Computer Science
Harz University of Applied Sciences
Friedrichstr. 57-59
38855 Wernigerode, Germany
{kschneider}@hs-harz.de

Abstract: The management and handling of big data is a major challenge in the area of life science. Beside the data storage, information retrieval methods have to be adapted to huge data amounts as well. Therefore we present an approach to improve search results in life science by recommendations based on semantic information. In detail we determine relationships between documents by searching for shared database IDs as well as ontology identifiers. We have established a pipeline based on Hadoop allowing a distributed computation of large amounts of textual data. A comparison with the widely used cosine similarity has been performed. Its results are presented in this work as well.

1 Introduction

Nowadays the management and handling of big data is a major challenge in the field of informatics. Larger datasets are produced in less time. In particular, this aspect is intensively discussed in life science. At a technology level, new concepts and algorithms have to be developed to enable a seamless processing of huge data amounts. In the area of data storage new database concepts are implemented, such as column based storage or in memory databases. In respect to data processing, distributed data storage and computation have been made available by new frameworks such as the Hadoop framework (http://hadoop.apache.org). Hadoop is using the MapReduce approach [DG08] allowing the distribution of tasks in the map phase over different clusters and to reduce the amount of data in the reduce phase. Furthermore Hadoop is able to integrate extensions such as the column oriented database HBase. So the framework combines the distributed computation architecture with the advantages of a NoSQL database system. Hadoop has already been used in life science applications such as Hadoop-BAM [NKS$^+$12] and Crossbow [LSL$^+$09].

Beside these technological aspects, information retrieval (IR) plays an important role as well. In this context search engines play a pivotal role for an integrative IR over widely spread and heterogeneous biological data. Search engines are complex software systems and have to fulfil various qualitative requirements to get accepted by the scientific community. Its major components are discussed in [LHM$^+$14]:

- Linguistic (text and data decomposition, e.g. tokenization; language processing, e.g. stop words and synonyms)
- Indexing (efficient search, e.g. inverse text index)
- Query processing (fuzzy matching and query expansion, e.g. phonetic search, query suggestion, spelling correction)
- Presentation (intuitive user interface, e.g. faceted search)
- Relevance estimation (feature extraction and ranking, e.g. text statistics, text feature scoring and user pertinence)
- Recommender systems (semantic links between related documents, e.g. "page like this" and "did you mean"

The implementation of those components is part of the research project LAILAPS [ECC$^+$14]. LAILAPS is an information retrieval system to link plant genomic data in the context of phenotypic attributes for a detailed forward genetic research. The underlying search engine allows fuzzy querying for candidate genes linked to specific traits over a loosely integrated system of indexed and interlinked genome databases. Query assistance and an evidence based annotation system enable a time efficient and comprehensive information retrieval. The results are sorted by relevance using an artificial neural network incorporating user feedback and behaviour tracking.

While the ranking algorithm of LAILAPS provides user specific results, the user might be interested in links to other relevant database entries to an entry of his interest. Such a recommender system is still a missing LAILAPS feature but would have enormous impact for the quality of search results. A scientist may search for a specific gene to retrieve all relevant information to this gene without a dedicated search in different databases. To realise such a goal, recommendation systems are a widely used method in information retrieval. This concept is already used in several life science applications. For example, EB-eye as IR system for all databases that are hosted at the European Bioinformatics Institute (EBI) provide suggestions to alternative database records [VSG$^+$10]. Another example are PubMed based IR systems for searching in biomedical abstracts [Lu11]. In this work we will describe a concept of providing recommendations in LAILAPS based on semantic information.

2 Results

When users are searching in LAILAPS for specific terms, the result is a list of relevant database entries. Beside the particular search result, the user would benefit from a list of related database entries that are potentially of interest to him. To implement this feature it is necessary to measure the similarity between database entries. A widely used concept is the expression of a document as a vector of words (tokens) and the computation of its distance by cosine similarity. Within this approach the tokens of each document will be compared, meaning documents using similar words have a higher similarity to each other. But to get a more useful result especially in the context of life science it would be necessary to integrate semantic information in the comparison of documents. Here we present a method allowing the estimation of semantic relationships of documents.

2.1 Get semantics with database identifiers

A widely used concept to provide semantic annotation in life science databases are ontologies such as the Gene Ontology (GO) [HCI$^+$04]. Each GO term has a specific ID allowing the exact identification of a term. Beside the use of ontologies, annotation targets are repositories of gene functions. This wide range of databases such as Uniprot [BAW$^+$05], have in common that they can be referenced by a unique identifier for each database entry.

With the help of such unique identifiers for ontologies and database entries the documents in LAILAPS can be compared on a semantic level. If for example two documents share a GO identifier, this could be interpreted as a semantic connection between these documents. The final goal therefore would be to design a recommendation system, which is determining these information and to recommend the end user database entries based on these unique identifier.

For the extraction of above mentioned identifiers, different methods could be applied. One method is the usage of regular expressions, where specific patterns are used, such as a token beginning with the letters GO, is likely a GO term [BSL$^+$11]. Another method is described in Mehlhorn et al. [MLSS12], where predictions are made with the support of a neural network. Feature extractions were focused on positions, symbols as well as word statistics to predict a database entry identifier. To include a very high number of database identifiers, we decided to use the neural network based approach, allowing the identification of IDs based on known ID patterns.

2.2 Determine document relations

We applied Hadoop to identify IDs in a high throughput manner. The Hadoop pipeline has two MapReduce components (see Figure 1). The first MapReduce job has a database as an input file. Each database entry consists of a unique document ID as well as document content. The mapper will then analyse each document and detect tokens that might be an ID with the IDPredictor tool from Mehlhorn et al. [MLSS12]. The reducer will then generate a list of pairs with a token and documents including this token. The second MapReduce job will then determine the document relations. Here the mapper will build pairs of documents having an ID in common. The reducer will finally count the number of shared IDs for each document pair. A high count of shared IDs means a high similarity between two documents. The source code of the pipeline is available at: http://dx.doi.org/10.5447/IPK/2014/18.

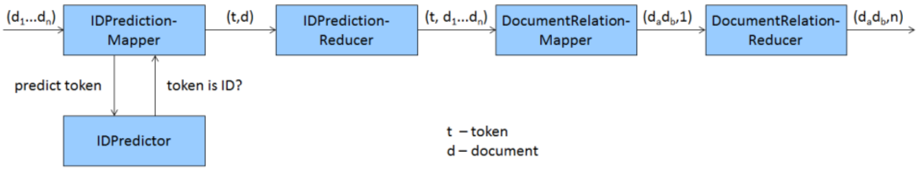

Figure 1: Hadoop pipeline including two MapReduce components as well as the ID prediction component

2.3 Cosine similarity versus ID prediction

As a benchmark we computed documents from the Swissprot database and compared the ranking results with the cosine similarity mentioned in section 2.1. While the cosine similarity score between two documents is built upon word frequencies and results in a value between zero and one, the ID prediction score is an integer value based on shared IDs. To make both values comparable, we calculated z-scores for both ranking scores. To detect deviations in the ranking, the results were plotted on a scatterplot (see Figure 2). The plot illustrates, that in most cases there are only small differences in the ranking. But there are some cases of large differences in the relative ranking, indicating, that for specific document relations the semantic component leads to a completely different ranking in contrast to the simple approach of comparing words.

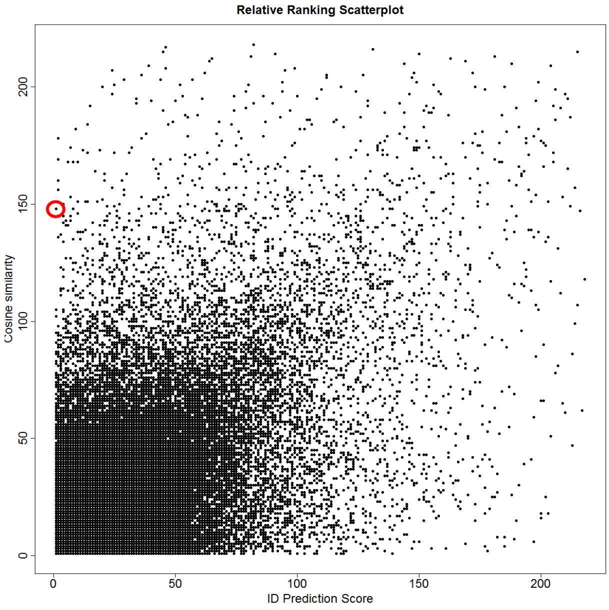

Figure 2: Scatterplot illustrating the different ranking results between cosine similarity and ID prediction score

When looking into specific results with strongly different rankings, semantically similarities could be detected. Picking up one example from Figure 2 (marked with a red circle) a document pair was ranked at place 1 in IDPrediction and at place 148 in cosine similarity. When looking into these documents we could determine that they are sharing a lot of IDs like EC (Enzyme Commission) numbers and GO terms. Both documents are dealing with fatty acid synthase in fungal species. A protein BLAST against Swissprot of the protein sequence of document A listed document B in the fourth position with a score of 1801 and an identity of 44%.

3 Discussion and Conclusion

In this work we developed a system providing recommendations based on semantic information. By the support of a neural network, IDs were predicted. With this information, documents can be compared on a semantic level. To support big data in life science we implemented the documents distance computation as a Hadoop pipeline. The results of our approach have shown differences to cosine similarity in case of rankings. The ID prediction based approach is able to detect semantic similarities between documents and recommend this information to the users. However to get a precise idea about the quality improvement, the new method should be applied to the LAILAPS frontend system to determine if the users are more interested in this new information. To implement the presented pipeline into LAILAPS powerful systems such as ORACLE Big Data [Dj13] could be a solution. It allows supporting multiple data source including Hadoop, NoSQL as well as the ORACLE database itself. Although Hadoop is a powerful system, LAILAPS would also benefit from a more integrative approach like using in memory technology. Users who would like to install their own LAILAPS instance might be not able to set up their own Hadoop cluster. In memory systems might be able to allow as the just-in-time computation of the available data as well. LAILAPS would benefit from further investigations in this field.

Acknowledgements

The expertise and support of Steffen Flemming regarding the establishment and maintenance of the Hadoop cluster is gratefully acknowledged.

References

[BAW+05] Bairoch, A.; Apweiler, R.; Wu, C.H.; Barker, W.C.;Boeckmann, B.; Ferro, S.; Gasteiger, E.; Huang, H.; Lopez, R.; Magrane, M. et al. The universal protein resource (Uniprot). Nucleic acids research, 33 (suppl 1):D154-D159, 2005.

[BSL+11] Bachmann, A.; Schult, R.; Lange, M.; Spiliopoulou, M. Extracting Cross References from Life Science Databases for Search Result

Ranking. In Proceedings of the 20th ACM Conference on Information and Knowledge Management, 2011.

[DG08] Dean, J.; Ghemawat, S.: Mapreduce: simpified data processing on large clusters. Communications of the ACM, 51(1):107-113, 2008.

[Dj13] Djicks, J.: Oracle: Big Data for the Enterprise. Oracle White Paper, 2013.

[ECC+14] Esch, M.; Chen, J.; Colmsee, C.; Klapperstück, M.; Grafahrend-Belau, E.; Scholz, U.; Lange, M.: LAILAPS – The Plant Science Search Engine. Plant and Cell Physiology, Epub ahead of print, 2014.

[HCI+04] Harris, M.A.; Clark, J.; Ireland, A.; Lomax, J.; Ashburner, M.; Foulger, R.; Eilbeck, K.; Lewis, S.; Marshall, B.; Mungall, C. et al.: The gene ontology (GO) database and informatics resource. Nucleic acids research, 32 (Database issue):D258, 2004.

[LHM+14] Lange, M; Henkel, R; Müller, W; Waltemath, D; Weise, S: Information Retrieval in Life Sciences: A Programmatic Survey. In M. Chen, R. Hofestädt (editors) Approaches in Integrative Bioinformatics. Springer, 2014, pp 73-109.

[LSL+09] Langmead, B.; Schatz, M.C.; Lin, J.; Pop, M.; Salzberg, S.L.: Searching for SNPs with cloud computing. Genome Biology, 10(11):R134, 2009.

[Lu11] Lu, Z.: PubMed and beyond: a survey of web tools for searching biomedical literature. Database. Oxford University Press, 2011

[MLSS12] Mehlhorn, H.; Lange, M.; Scholz, U.; Schreiber, F.: IDPredictor: predict database links in biomedical database. J. Integrative Bioinformatics, 9, 2012.

[NKS+12] Niemenmaa, M.; Kallio, A.; Schumacher, A.; Klemelä, P.; Korpelainen, E.; Heljanko, K.: Hadoop-BAM: directly manipulating next generation sequencing data in the cloud. Bioinformatics, 28(6):876-877, 2012.

[VSG+10] Valentin, F.; Squizzato, S.; Goujon, M.; McWilliam, H.; Paern, J.; Lopez, R.: Fast and efficient searching of biological data resources - using EB-eye. Briefings in bioinformatics, 11(4):375-384, 2010.