Combining Data and Text Mining Techniques for Yeast Gene Regulation Prediction: A Case Study

Mark-A. Krogel[1], Marcus Denecke[1], Marco Landwehr[2], and Tobias Scheffer[1]

[1]University of Magdeburg, FIN/IWS
Universitätsplatz 2, 39016 Magdeburg, Germany
{krogel, denecke, scheffer}@iws.cs.uni-magdeburg.de

[2] Leibniz Institute for Neurobiology
Brenneckestr. 6, 39118 Magdeburg, Germany
landwehr@ifn-magdeburg.de

ABSTRACT

In order to solve task 2 of the KDD Cup 2002, we exploited various available information sources. In particular, use of relational information describing the interactions among genes and information automatically extracted from scientific abstracts improves the accuracy of our predictions.

Keywords

KDD Cup, propositionalization, text classification, information extraction, ROC analyses.

1. INTRODUCTION

KDD Cup 2002 task 2 asked for models that predict some specific cellular activity (the AHR signaling pathway) of yeast after the knockout of certain genes. For the proteins encoded by the genes, information about function, localization, and protein classes were given, as well as data about pairwise interactions between them. In addition, several thousand abstracts of research papers on those genes and proteins were provided as a further source of data. More details on the task can be found in an overview article by Craven (this issue).

Our solution is greatly benefiting from an approach to deal with relational information on the interaction of genes by a propositionalization algorithm [1]; we had used this same algorithm successfully for tasks 2 and 3 of the preceding KDD Cup 2001. We could achieve a further improvement by using an information extraction algorithm that allowed us to utilize the scientific abstracts effectively.

2. PROPOSITIONALIZATION

Propositionalization is the process of the transformation of a multi-relational representation of data – as it can be found in relational databases – into the form of a single table. RELAGGS (RELational AGGregationS) [3] computes several joins of the input tables according to their foreign key relationships. These joins are compressed using equivalent functions to SQL avg, count, max, min, and sum, specific to the data types of the table columns, such that there remains a single row for each example, here for each gene. Results of several such join compressions are concatenated example-wise. The result is an appropriate input for conventional data mining algorithms.

For the task at hand, we designed a new schema of a database that could serve as input for RELAGGS. We designed a table "Gene" to contain the names of all genes that were spread over the original tables. Information contained in the names – cf. http://www.uni-frankfurt.de/fb15/mikro/euroscarf/stra_des.html – as well as the class labels were included in this table, see Figure 1.

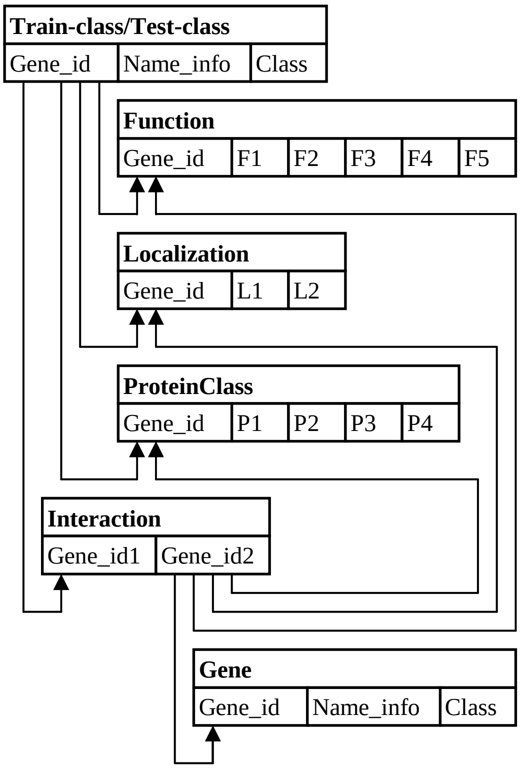

Figure 1. Data set representation as 6 linked tables

Tables "Train-class" and "Test-instances" (with class information given after the Cup as "Test-class") are in fact materialized views of table "Gene", containing just the training and test examples, respectively.

For information about function (5 levels of hierarchy), localization (2 levels), and protein class (4 levels), we introduced columns per level. These columns contain the appropriate values won from a split of the original representation of this information. In the original variant, values of different hierarchy level where concatenated in a special way.

For interactions, we made symmetry explicit. We included a line to state that gene B interacts with gene A if there was the fact that A interacts with B contained in the original table. We also included rows for certain transitivity assumptions. For instance, for second level interactions, we included rows that express that gene A interacts with gene C, if there are entries for interactions between A and B and between B and C in the original table.

RELAGGS produced joins of those tables along foreign links [4] (indicated by the arrows in Figure 1) and compressed these mainly by just counting the different possible values per training and test gene/protein, respectively. It concatenated these results and thus finally output a table with about 1,000 columns for further analysis using Joachim's SVM^{light} [2].

3. TEXT MINING

In order to exploit the abstracts provided for analysis, we experimented with two different approaches: text classification and information extraction. Since there were many missing values in the tables, the latter approach was especially intended to find more values for function, localization and protein classes.

For text classification, we put together abstracts per gene, applied a stemming algorithm, and formed a TFIDF representation as an input to SVM^{light}. The decision function values output by this learner served as an additional attribute for the corresponding RELAGGS results.

For information extraction, we again merged the abstracts per gene and implemented a tool to efficiently find search terms. These were produced from the hierarchy files of possible values for function, localization, and protein classes according to a few simple rules, such as the addition of plural forms to the original lists, e.g. "nuclei" in addition to "nucleus". On finding values from our search term list, the corresponding original values were included in the appropriate input tables for RELAGGS.

4. RESULTS

As a solution for KDD Cup 2002 task 2, we handed in the results of a model for the so-called "narrow class problem" as one of the two subtasks of task 2 that included the additional name information, interaction information up to the second level, and results of information extraction. With these predictions, we could achieve the best result on this subtask, and with the very same predictions, the result on the "broad class problem" was still good enough for a good overall result.

With class information for test examples available now, we tried to find out the influence of the different experimental conditions here. For the additional information from gene names as well as text classification information, we can not observe relevant differences. However, using interaction information and data from information extraction improved the predictive power of the models, cf. Fig. 2 and 3.

5. CONCLUSION

The approach of propositionalization in combination with text mining techniques seems promising as indicated by our experimental results. As an addition, we plan to perform experiments with a co-learning algorithm.

6. REFERENCES

[1] Cheng, J., Hatzis, C., Hayashi, H., Krogel, M.-A., Morishita, S., Page, D., and Sese, J. KDD Cup 2001 Report. *ACM SIGKDD Explorations*, 3(2), pp.47–64.

[2] Joachims, T. *Learning to Classify Text using Support Vector Machines*. Kluwer, 2002.

[3] Krogel, M.-A. and Wrobel, S. Transformation-Based Learning Using Multirelational Aggregation. In *Proceedings of the 11th International Conference on Inductive Logic Programming (ILP),* Springer-Verlag, 2001.

[4] Wrobel, S. An algorithm for multi-relational discovery of subgroups. In Proceedings of the 1st European Symposium on Principles of Data Mining and Knowledge Discovery (PKDD), Springer-Verlag, 1997.

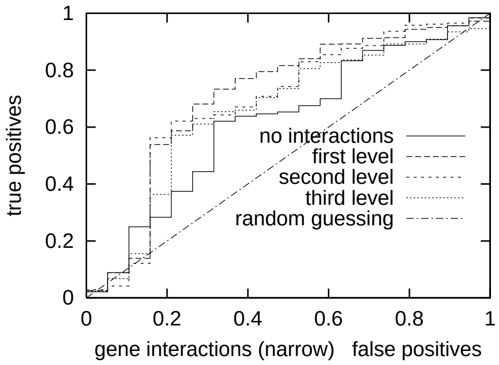

Figure 2. Influence of interaction information on ROCs.

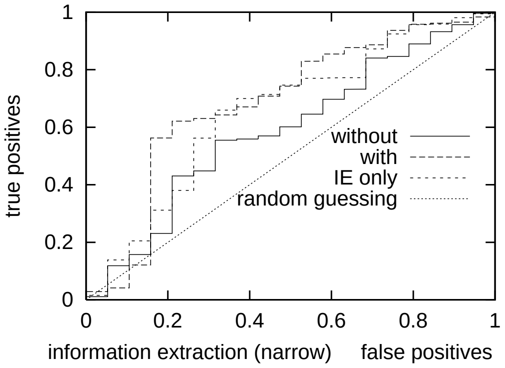

Figure 3. Influence of information extraction on ROCs.

About the authors:

Mark-A. Krogel received his first degree in computer science from the University of Magdeburg, Germany, and an M.Sc. in cognitive science from the University of Edinburgh, Scotland. He is now a Ph.D. student in the research group for Machine Learning and Knowledge Discovery at the University of Magdeburg.

Marcus Denecke received his B.Sc. in computer science and economics at the University of Magdeburg where he also works as a research assistant.

Marco Landwehr studied chemistry with a specialization in biophysical chemistry at the University and Medical School of Hanover. He is now a Ph.D. student in the research group for Molecular Plasticity at the Leibniz Institute for Neurobiology.

Tobias Scheffer received his Ph.D. from the Technische Universität Berlin. After working at Technische Universität Berlin, Siemens Corporate Research, the University of New South Wales and SemanticEdge, he co-founded Tonxx and is now teaching at the University of Magdeburg.