Humboldt-Universität zu Berlin – Institut für Informatik

Exposé zur Studienarbeit

Classifying abstracts of biomedical literature for stem cell research

David Asher

June 22, 2012

Supervisors: Prof. Dr. Ulf Leser, Mariana Neves

1 Motivation

Stem cells [12, 13] are biological cells that can differentiate into more specialized cells and have the ability to produce more stem cells through mitosis. It is widely believed, that if they are better understood, stem cell therapies can be developed, which would allow a dramatic improvement of the treatment of human diseases like cancer, Parkinson's, cardiac failures and many more.

Due to their self-renewing and specializing property along with the widespread optimism among the biomedical research community, many scientific research papers on this topic are being published in biomedical journals.

The U.S. National Library of medicine provides the MEDLINE/PubMed[1] database, which contains references to biomedical articles. It accommodates over 19 million references to articles that are published in over 5,5000 journals with biomedical focus, which makes it the largest bibliographic database for biomedical literature.

Because of its sheer size and the not necessarily consistent terminology in biomedical sciences looking for bibliographical references only with keyword-based queries can be a very time consuming activity. In order to make this database more accessible for biologists, more sophisticated search methods than the classical keyword-based search through title and abstract of the articles have been developed. We briefly mention three approaches to facilitate the life of the researcher: searching with MeSH terms and the two projects GoPubMed and Caipirini.

To deal with the problem of ambiguous vocabulary in biomedical sciences, MeSH terms[2] (Medical Subject Headings Terms) have been introduced. They consist of a hierarchically organized, expert-regulated vocabulary of biomedical terms. Thus, we find for example in the hierarchical level, defined by the path '/anatomy/cells/stem cells', terms like 'adult stem cells', 'embryonic stem cells', 'fetal stem cells', etc. Each newly added bibliographical item in MEDLINE/PubMed can be associated with a set of major and minor MeSH terms, hence allowing to perform a search with a uniform vocabulary.

GoPubMed [4] is another project with the aim of simplifying the work of the researcher. It allows the user to narrow the result of his MEDLINE/PubMed query with the help of four categories: the 'who', the 'where' and the 'when' categories let the user specify authors, location and date of publication of his desired references, whereas under the 'what' category the user finds the results of his query grouped according to the hierarchies of MeSH and GO (Gene Ontology, another controlled vocabulary for genes), which enables further curtailing of the result by deselecting unwanted terms.

However, not all of the articles, that are published in journals are necessarily interesting for biologists. Of the 104,332 articles in MEDLINE/PubMed which are associated with the MeSH-Term 'stem cell', we also find papers, that are dealing with funding, business opportunities, patents, moral aspects of stem cell research, legal situation in different countries, etc. MeSH terms do not help to filter these articles out, so it is desirable to have a tool which classifies articles into relevant and non-relevant for stem cell research.

The Caipirini [11] project goes away from rigid and often too coarse-grained hierarchies and tries to adopt itself dynamically according to the researcher's need. It does so by letting the user give examples of relevant and irrelevant documents via PubMed-IDs. Having the samples provided, a support vector regularization model is trained and a third list of PubMed-IDs can be sorted by their estimated probability to belong to the relevant set. The project, though, is optimized for throughput and response time rather than for classification accuracy.

The purpose of this student research project is to check the performance of a text classifier for the cognition of biomedical literature on human embryonic stem cells. Therefore, a classifier will be developed using state of the art natural language processing and machine learning techniques, that attempts to classify articles by analyzing abstracts of articles of biomedical publica-

[1] cf. http://www.nlm.nih.gov/pubs/factsheets/medline.html and http://www.nlm.nih.gov/pubs/factsheets/pubmed.html

[2] cf. http://www.nlm.nih.gov/pubs/factsheets/mesh.html

tions. We give more importance to accuracy of the classification result than to high efficiency. Its performance will be evaluated by common measures of performance.

The rest of this document is organized as follows: different kinds of features that can be generated from documents and several machine learning algorithms are discussed in the Sections 2.1 and 2.2. In Section 2.3 we present some details about the training data that will serve as input for the machine learning algorithms, followed by a discussion about the evaluation in Section 2.4. The last Section gives an overview about related work.

2 Approach

2.1 Feature Derivation

Representation

Since texts cannot directly be interpreted by machine learning algorithms, we need to map a text t to a document vector $d_t = (w_1, \ldots, w_n)$ in such a way that d_t reflects all important characteristics of the text t. The scalars w_i are called weights. They should describe the contribution of the term associated with the integer i to the characteristic of the document. The process of creating such a mapping is often called document indexing [10]. In its simplest variant the weights are chosen out of the set $\{0, 1\}$, denoting the presence or absence of a term. More sophisticated real-valued weighting schemes are *term frequency* (tf) which counts the number of occurrences of a term in the document, multiplied by "importance factors" such as *inverse document frequency* (idf) or *relevant document frequency* (rf). When using decision tree learning, the binary weighting scheme has the additional advantage that it relieves from us to discretize our obtained document vectors. Machine learning algorithms are particularly sensible to input with highly different scale [5]. To guarantee that every weight falls in the same range, *cosine normalization* can be performed which ensures that every w_i lies in the real interval $[0; 1]$.

Feature Extraction

Another task to be carried out is deciding the set of terms for which the weights should be obtained. The following list describes possible sources for features.

- *Bag of words* [10] – The set of terms consists of every word that appears in the corpus. In order to save save computation time, stopword removal can be carried out. Stopwords are words which usually do not represent useful information for text categorization but have a high frequency, like preposition, articles, etc.

- *Stemming* [10] – Stemming algorithms reduce words to their linguistic stem. Hence words which are inflected differently but share the same stem, will be represented by the same term, resulting in a smaller term set.

- *Bi-grams* [2] – Looking just at isolated words often result in loss of information. Bi-gram approaches do not only consider single words but also the word immediately following a word.

- *MeSH terms* – Many abstracts are associated with minor and major MeSH terms, which serve as an additional features.

- *Named Entity Recognition* – Named entity recognition is the task of identifying biomedical entities in texts. Named entities often consist of several tokens and would thus be ignored if we only consider the bag of words. The extracted entities can be incorporated in the set of terms.

- *Dimensionality Reduction* [10] – If the set of features turns out to be too large to scale well with our machine learning algorithms, further dimensionality reduction can be performed. The most popular methods, among many alternatives, are χ^2-feature selection, feature selection with information gain and odds-ratio. Dimensionality reduction has the additional advantage of preventing the classifier from overfitting.

2.2 Machine Learning Algorithms

In this section we discuss four machine learning algorithms that can be used and address questions concerning the choice of parameters.

Support Vector Machines

The document vectors of the training set are regarded as vectors in n-dimensional vector space. In its simple variant, a support vector machine (SVM) [10] tries to find an optimal hyperplane that separates the two document classes. A hyperplane is considered optimal, when the nearest point of both classes to the hyperplane has maximal distance. If the classes are not linearly separable in the vector space, a kernel can be specified. Kernels that perform non-linear mappings allow the algorithm to look for an optimal hyperplane in a transformed version of the original n-dimensional vector space, with higher dimensionality, in which the training data hopefully appears linearly separable.

k-Nearest Neighbor (k-NN)

In k-NN [8, 9], all document vectors of the training set correspond to points in the n-dimensional vector space. For the classification of a document, at first the k training samples with the minimal distance to the document to classify are determined. The class eventually assigned to the document is the class, of which the majority of the k found samples belong.

The k value and the metric we use to calculate the distances must be chosen very carefully, since both parameters have a great effect on the performance of k-NN. If k is too small the prediction is more susceptible to noise and if k is too big, the prediction will be biased towards the class to which the majority of the samples belong. A simple heuristic for the choice is to increase k successively in a loop while simultaneously testing the achieved performance on a validation set and then fixing k on the best result.

The standard k-NN approach has the problem of assuming, that far distant neighbors are equally important as very near neighbors for the determination of the class. An easy improvement of the the standard k-NN rule is to scale neighbors proportional to their distance. Far distant neighbors should be given lower weight and closer neighbors should be weighed higher. Another refinement is using a more sophisticated distance metric than the euclidean one. In the euclidean metric every feature contributes equally much to the distance, though, certainly more predictive features should be valued higher. A candidate for such an improved metric uses information gain as scale (cf. also Section 3).

Decision Trees

Decision trees [8] work, in contrast to the above discussed algorithms, on a discrete rather than a real valued domain. Each internal node of a decision tree consists of a test on a feature. The edges departing from an internal nodes are labeled with every possible value the tested feature can attain. Every leaf is labeled with a class. For classification, we start at the root node and then recursively branch to the next level, according to the outcome of the internal node tests. When we eventually reach a leaf, its class label determines the document class. Algorithms for the creation of decision tree algorithms attempt to place tests on more predictive features in a higher position of the tree. The most common measurement for predictiveness is information gain, though other measurements are conceivable. Many algorithms perform a pruning on the constructed tree to avoid overfitting.

Decision trees can also work with real valued vectors by testing on ranges rather than equality. However, this results in larger trees because for each range an internal node test has to be performed.

An advantage of decision trees over k-NN is their ability to predict a class by polling only a subset of all available features, thus creating more general rules for classification.

Classifier Committee

For a task which requires expert knowledge, more then one expert often get better results than a single expert. This is the motivation of constructing classifier committees [10, 8], i.e. combining many classifiers (the committee) to a single one. There are many ways to merge classifiers. Besides majority voting, there have been proposed solution, that give different amount of credit to individual classifiers of the committee. A common way to do this is by calculating the weighted linear combination of the classifier, in which each weight reflect the credibility we give to the corresponding classifier.

Dealing with imbalanced Training Data

Machine learning algorithms are designed to optimize for accuracy rather than precision and recall. With highly imbalanced training data sets there is always the risk, that classifiers are biased towards the majority class. Text classification problems often have more negative than positive examples. A way to deal with this issue are resampling techniques [14]. One can either downsample negative documents or oversample positive documents.

2.3 Training Data

Löser et al. [7] examined 990 biomedical publication, published until the end of 2008, in order to find out how many publicly disclosed hESC-lines (human embryonic stem cell) are available and to examine the impact of these publications. The abstracts of these articles will be used as positive examples. However, we lack a set of negatively labeled abstracts. Such a set can be constructed by performing a search on the MEDLINE/PubMed database using the keyword 'human embryonic stem cell' on all articles published until the end of 2008. From the result we should dismiss all abstracts that also appear in our positive list.

2.4 Evaluation

For each machine learning algorithm, we will construct a baseline classifier and compare its precision, recall and F_1-measure using leave-one-out cross-validation against more sophisticated variants of the same algorithm. Later we will check which approach worked best and if classifier committees are able to improve performance.

3 Related Work

General overviews about text classification can be found in Sebastiani [10] and Joachims [6]. Sebastiani describes methods for obtaining document vectors, dimension reduction of the feature set and depicts several machine-learning algorithms. Joachims focuses on text categorization with SVMs and performs several experiments with different parameter sets.

Cohen [3] describes a general purpose, SVM-based approach for automatic classification of biomedical documents. The approach uses binary document vectors, indicating the presence or absence of terms in abstract and title, all assigned MeSH-Terms as well as normalized biological entity identifies. The set of features is

reduced by χ^2-feature-selection. To address the issue with unequally distributed document classes, he proposes a cost-sensitive learning method by downsampling the set of negatives with *cost-proportionate rejection sampling* before applying a linear kernel SVM on the training data. He calls his classifier SVM-PCR. The approach is tested on all four training sets of the TREC 2005[3] task, where it significantly outperforms the baseline SVM-only approach. Without χ^2-feature-selection, inferior results are achieved and the stronger the skew of negatives and positives in the training set, the stronger is the positive effect of downsampling.

Ambert and Cohen [1] describe a classifier based on k-NN to determine articles related to protein-protein-interaction. Besides the set of words in the abstract, they also adopt biological entities identified by an NER component and a set of regular expressions. They apply their approach on the BioCreative II.5[4] Article Categorization sub-Task. For the evaluation, they compare their approach with a baseline linear kernel SVM and a SVM-PCR (v.s.) classifier. The k-IGNN approach outperforms both the SVM-only and the SVM-PCR approach and achieves good results even with sparse samples. Their idea is to scale individual features by their information gain (IG).

References

[1] Kyle H. Ambert and Aaron M. Cohen. k-information gain scaled nearest neighbors: A novel approach to classifying protein-protein interaction-related documents. *IEEE/ACM Trans. Comput. Biol. Bioinformatics*, 9(1):305–310, January 2012.

[2] R. Bekkerman and J. Allan. Using bigrams in text categorization. Technical Report IR-408, Center of Intelligent Information Retrieval, UMass Amherst, 2004.

[3] Aaron M. Cohen. An effective general purpose approach for automated biomedical document classification,. *Proc. Am. Medical Informatics Assoc. (AMIA) Ann. Symp.*, pages 161–165, 2006.

[4] Heiko Dietze, Dimitra Alexopoulou, Michael R. Alvers, Bill Barrio-Alvers, Andreas Doms, Jörg Hakenberg, Jan Mönnich, Conrad Plake, Andreas Reischuk, Loic Royer, Thomas Wächter, Matthias Zschunke, and Michael Schroeder. Gopubmed: Exploring pubmed with ontological background knowledge. In Michael Ashburner, Ulf Leser, and Dietrich Rebholz-Schuhmann, editors, *Ontologies and Text Mining for Life Sciences : Current Status and Future Perspectives*, number 08131 in Dagstuhl Seminar Proceedings, Dagstuhl, Germany, 2008. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany.

[5] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2 edition, November 2001.

[6] Thorsten Joachims. Text categorization with support vector machines: learning with many relevant features. In Claire Nédellec and Céline Rouveirol, editors, *Proceedings of ECML-98, 10th European Conference on Machine Learning*, pages 137–142, Heidelberg et al., 1998. Springer.

[7] Peter Löser, Jacqueline Schirm, Anke Guhr, Anna M. Wobus, and Andreas Kurtz. Human embryonic stem cell lines and their use in international research. *STEM CELLS*, 28(2):240–246, 2010.

[8] Tom Mitchell. *Machine Learning*. McGraw Hill, 1 edition, March 1997.

[9] Euihong (sam Han, George Karypis, Vipin Kumar, and Vipin Kumar. Text categorization using weight adjusted k-nearest neighbor classification. Technical report, 1999.

[10] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Comput. Surv.*, 34(1):1–47, March 2002.

[11] Theodoros G. Soldatos, Sean I. O'Donoghue, Venkata P. Satagopam, Adriano Barbosa-Silva, Georgios A. Pavlopoulos, Ana Carolina C. Wanderley-Nogueira, Nina Mota M. Soares-Cavalcanti, and Reinhard Schneider. Caipirini: using gene sets to rank literature. *BioData mining*, 5(1):1+, February 2012.

[12] James A. Thomson, Joseph Itskovitz-Eldor, Sander S. Shapiro, Michelle A. Waknitz, Jennifer J. Swiergiel, Vivienne S. Marshall, and Jeffrey M. Jones. Embryonic stem cell lines derived from human blastocysts. *Science*, 282(5391):1145–1147, 1998.

[13] Bernard E. Tuch. Stem cells–a clinical update. *Australian Family Physician*, pages 719–721, 2006.

[14] Michael Wasikowski. Combating the Class Imbalance Problem in Small Sample Data Sets. Master's thesis, Kansas School of Engineering, 2009.

[3] cf. http://trec.nist.gov/pubs/trec14/t14_proceedings.html
[4] cf. http://www.biocreative.org/events/biocreative-ii5/