Instinctive Mining of Protein Names from Biomedical Text

B.V. Subba Rao[1], K.V. Sambasiva Rao [2]

Abstract

Automated information extraction from biomedical literature is important because a vast amount of biomedical literature has been published. Recognition of the biomedical named entities is the first step in information extraction. With the increasing amount of biomedical text, there is a need for automatic extraction of information to support biomedical researchers. Due to incomplete biomedical information databases, the extraction is not straightforward using dictionaries, and several approaches using contextual rules and machine learning have previously been proposed. Our work is inspired by the previous approaches, but is novel in the sense that it is fully automatic and does not rely on expert tagged corpora. The main ideas are 1) unigram tagging of corpora using known protein names for training examples for the protein name extraction classifier and 2) tight positive and negative examples by having protein-related words as negative examples and protein names/ synonyms as positive examples.

We present preliminary results on Medline abstracts about gastrin, further work will be on testing the approach on BioCreative benchmark data sets.

Key Words: Information Extraction, Biomedical Literature, Machine learning, Unigram tagging, Protein.

1. Introduction

Due to the increasing importance of accurate and up to date protein/gene information databases and ontologies for biomedical research, there is a need to extract protein information from biomedical research literature, e.g. those indexed in Medline. Methodologically these approaches belong to the information extraction field [5,11,12], and in the biomedical domain they range from learning relationships between proteins/genes based on co-occurrences in Medline abstracts [9, 14] to manually developed protein information extraction rules and protein name classifiers trained on manually annotated training corpora [1,2].

The paper is organized as follows. Section 2 describes the materials used, section 3 presents our methodology, section 4 describes related work, section 5 presents empirical results, and finally section 6 contains the conclusion.

2. Materials

Examples of Protien Names in a Textual context are a) "duodenum, a peprtone meal in the ".b)"subtilism plus leucine aminopeptidase plus prolidase followed".

The materials used included biomedical (sample of Medline abstract) and general English (Brown) textual corpora, as well as protein databases, As subject for the expert validation experiments we used the collection of 12,238 gastrin-related Medline abstracts

Examples of Protien Names in a Textual context are a) "duodenum, a peprtone meal in the ".b)"subtilism plus leucine aminopeptidase plus prolidase followed".

[1] Associate Professor, Department of IT, P.V.P Siddhartha Institute of Technology, Vijayawada -520007

[2] Principal, M.V.R. College of Engneering Vijayawada, Krishna Dt., A.P.

that were available in October 2005. Gastrin was selected to fit the field of expertise of the researchers who evaluated the findings.

As a source for finding known protein names we use a web search system called Gsearch, developed at Department of Cancer Research and Molecular Medicine at NTNU. It integrates three common online protein databases, namely Swiss-Prot, LocusLink and UniGene. The Brown repository (corpus) is an excellent resource for training a Part Of Speech (POS) tagger. It consists of 1,014,312 words of running text of edited English.

3. Our Methodology

We used modular approach where every sub module can easily be replaced by other similar modules in order to improve the general Abstract performance of the system. The main modules correlate with the main tasks that have to be solved in an information extraction setting. There are four modules connected to the data gathering phase, namely data selection, tokenization, POS-tagging and Stemming. Then three modules deal with classification, namely Gsearch, feature extraction and Classification. The last three modules are evaluation modules that handle cross-validation, expert evaluation and dataset statistics.

3.1. Data Selection

The data selection module uses PubMed[13] Entrez online system to return a set of PubMed IDs (PMIDs) and abstracts for a given protein, in our case "gastrin" (symbol GAS).

3.2. Tokenization

The text is tokenized to split it into meaningful tokens, or "words". We use the White Space Tokenizer. Words in parentheses were clustered together and tagged as a single token with the special tag Paren.

3.3. POS Tagging

Using a Brill tagger trained on the Brown Corpus. This module acts as an advanced stop-word-list, excluding all the everyday common American words from protein search. Later, the actual POS tags are also used as context features for their neighboring words.

3.4 Porter-Stemming

If the stem of a word can be tagged by the Brill tagger, then the word itself is given the special tag "STEM" and thereby transferred to the common word list.

3.5 Gsearch.org

Tagging is way of automatically creating positive and negative examples for the protein name extraction stage. Classifiers in general follow the rule "garbage in equals garbage out". One way to improve this is to do careful feature selection. Another is in the selection of positive and negative training data which is what we are focusing on.

The idea is that if an information extraction classifier should be able to discern between protein names and other entities, it in particular needs to handle entities that are as close to protein names as possible, i.e. protein-related entities.

Table1. Feature approaches

Acronym	Description
F1	3 neighbors w/all
F2	3 neighbors w/text
F3	3 neighbors w/text & POS
F4	3 neighbors w/POS & word-has-bracket

examples (i.e. protein-related entities) by using words describing proteins, and positive examples by using protein names and synonyms. The proteins, synonyms and corresponding descriptions are found using the Gsearch.org search engine. It enables simultaneous searches in the Swiss-Prot, UniGene and Locus Link protein databases. The remaining words are the untagged words that need to be classified (with the classifier trained on the positive and negative data generated in this step).

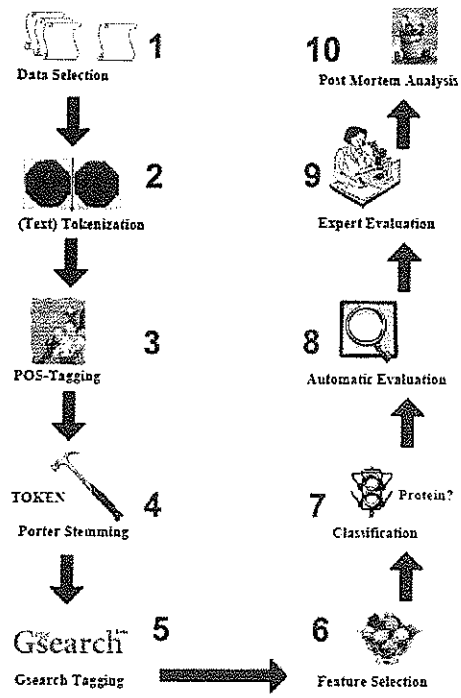

Figure1. Overview of Our Methodology

3.6. Feature Selection

The features we use are the word itself (TEXT), the given tag (POS) from Brill or Gsearch (or None if the word is untagged), and other True/False features like *has bracket ,has first upper, has non alpha num prefix, is lowercase, is numeric, is uppercase.* The features are collected for the word in question, and for the n nearest neighbors (we use n = 3 in our experiments).

3.7. Classifier Performance

The positive and negative examples connected with the features described above are then used as training data for classification of untagged tokens as part of a protein name or not. Selection of classifiers is quite pragmatic due to the no free lunch theorem [7], i.e. "there is no best classifier for all problems".

We used the following classifiers: Support Vector Machines (with lin., pol., sig. and rbf kernels) in the SVM-Light tool [11], Naive Bayes in the Orange tool [6] and a Proximal Support Vector Machine.

3.8. Automatic Evaluation

Table2. Automatic Evaluation results

Classifier	F1	F2	F3	F4
Majority	75.9	75.9	75.9	75.9
SVM Lin. t	75.9	75.9	75.9	75.9
SVM Pol.	**76.4**	75.9	75.9	75.9
SVM RBF	76.1	75.9	75.9	75.9
SVM Sig.	75.7	75.9	75.9	75.9
PSVM($v = 100$)	68.0	N/A	N/A	N/A
PSVM($v = 1$) XV	74.2	N/A	N/A	N/A

In order to efficiently test our extraction approach we first try to classify known data. If this gives extremely poor results there is no reason to pursue in classifying untagged tokens. The methods applied were "train and test" sets of 2500 examples each with various feature set combinations, as well as 10-fold cross validation in order to test whether the "train and test"-set approach was ok.

3.6. Expert Evaluation

The whole purpose of the extraction approach is to find proteins among untagged tokens. In order to do this we

gave a sample of untagged tokens and their surrounding textual context to molecular biologists so they could say if each token was a part of a protein name or not. We then used this as the golden standard to test our classifier performance and to measure true/false positives/negatives and to calculate F Score and classification accuracy.

3.7. Post Mortem Analysis

In order to characterize the size of the untagged protein names problem, we used the expert tagging from the molecular biologists in order to estimate a confidence interval for i) the probability of an untagged token being part of a protein name, and ii) the probability of a token being untagged, given our tagging sources.

4. Related Work

Our specific approach was on using existing databases to automatically annotate information extraction classifiers in biomedical corpora, and at the same time using these databases to create both positive and negative examples. We have not been able to find other work that does this, but there are quite a few approaches on extracting protein names from biomedical literature. Below, a brief overview is given.

Bunescu et al. present a method similar to ours, except that they train their classifiers on manually created corpora [2, 3, 4]. Ginter et al. describe a method weighting words by positions for resolving gene/protein name disambiguation, but they use a manually developed corpus for training [8].

Bickel et al. describe an approach using Support Vector Machine classifiers for gene name recognition, but it is also trained using a manually generated biomedical corpus [1]. Mukherjea et al. describe a method that combines manually generated rules with rules learned using UMLS to do biomedical information extraction . Torii and Vijay-Shanker use an unsupervised bootstrapping technique from Word Sense Disambiguation. This resembles approach in the sense that it is fully automatic, but differs in the sense that they use an unsupervised bootstrapping technique on names found using the manually developed rules described a supervised method using comprehensive domain knowledge and dictionaries together with classifiers for biological term extraction [10].

Table3. Protein Classification – untagged words

Classifier	TP/TN	FP/ FN	Prec/Rec/F	CA
N. Bayes	6/120	67/7	8/46/27	63
Majority	0/187	0/13	NA/0/NA	94
SVM Lin	0/187	0/13	NA/0/NA	94
SVM Pol	6/159	28/7	18/46/32	83
SVM rbf	3/174	13/10	19/23/21	89
SVM Sig	0/186	1/13	0/0/NA	93

5. Empirical Results

Since our motivation is to test the feasibility of 1) automatic creation of training data for protein name classifiers and 2) selection of appropriate negative examples in the training data, we did not put much emphasis on the optimal selection of features for the information extraction classifiers. That is a natural next step, but outside the scope of this paper. The different feature sets we used are described in table 1, and more details about the features are given in our methodology.

5.1 Automatic Evaluation

In order to get an overview of which classifier performance to expect, we first tested them on already tagged data, using protein names and symbols found in Gsearch as positive examples and other words from Gsearch (assumed to be protein-related) as negative examples (results in table 2). The data was first divided into a training and test set with 2500 examples each, and later we did a 10-fold cross-validation (XV) on all 5000 examples (train+test set) to verify the train and test approach.

5.2 EXPERT EVALUATION

The main purpose of our extraction approach is to detect which untagged words that are part of protein names. In order to do (and test) this, we first tagged using the Brown Corpus (regular English words) and Gsearch (protein names and protein related words) and then we selected a sample of 200 words that had not been tagged. These words and their corresponding textual contexts were classified using the classifier, and compared to manual annotations done by biologists (table 3).

5.3 POST MORTEM ANALYSIS

In order to say something more general about the number of protein names that cannot be tagged with LocusLink, Swiss-Prot and UniGene, we used the results after stage 5 (Gsearch tagging) and the expert's classifications of untagged words. We created confidence intervals for the probability of a word being untagged after stage 5, and for the probability that an untagged word is a part of a protein name. The total number of unique tokens in the 12000 abstracts covering gastrin is N = 76359, and 26885 of them were untagged. This gives an estimated probability of an untagged token P_u = 26885/76359 = 35.21% and σ_u = "$P_u(1-P_u)/N \approx 0.0017$. The 95% confidence interval is [0.3521-1.96×0.0017, 0.3521+1.96× 0.0017] [34.88%, 35.54%] The expert found 13 protein names among a random sample of n = 200 untagged tokens (random sample from 26885 unique untagged tokens in total), this gives an estimated probability that an untagged word is a part of protein name P_p = 13/200 = 6.5% and σ_p ="$P_p(1-P_p)/n$ 0.0173. The 95% confidence interval of [6.5 - 1.96 × 1.73, 6.5 + 1.96 × 1.73] = [3.11%, 9.89%]

6 Conclusion

This paper presents a novel method for automatically creating both positive and negative training data for protein name extraction classifiers. Since we focused on the automatization of creating training data and relevant negative examples, we only used relatively simple domain modeling and feature extraction/selection approaches. This leads to promising, though not yet highly accurate, empirical results. So in the next round we need additional work on i) feature extraction and selection, and ii) incorporating domain knowledge. The approaches presented in [10, 12] seems to be complementary to ours and might increase accuracy in future versions.

References

[1] S. Bickel, U. Brefeld, L. Faulstich, J. Hakenberg,U. Leser, C. Plake, and T. Scheffer. A Support Vector Machine classifier for gene name recognition. *In Proceedings of the EMBO Workshop: A Critical Assessment of Text Mining Methods in Molecular Biology,* March 2004.

[2] R. Bunescu, R. Ge, R. J. Kate, E. M. Marcotte, R. J.Mooney, A. K. Ramani, and Y. W. Wong. Comparative Experiments on Learning Information Extractors for Proteins and their Interactions. *Journal Artificial Intelligence in Medicine: Special*

Issue on Summarization and Information Extraction from Medical Documents

[3] R. Bunescu, R. Ge, R. J. Kate, R. J. Mooney, Y. W.Wong, E. M. Marcotte, and A. K. Ramani. Learning to Extract Proteins and their Interactions from Medline Abstracts. *In Proceedings of the ICML-2003 Workshop on Machine Learning in Bioinformatics,* Pages 46–53, August 2003.

[4] R. Bunescu, R. Ge,R. J.Mooney, E.Marcotte, and A.K.Ramani. Extracting Gene and Protein Names from Biomedical Abstracts. Unpublished Technical Note, Machine Learning Research Group, University of Texas at Austin, USA, March 2002.

[5] J.CowieandW.Lehnert. Information Extraction. *Communications of the ACM,* Pages 80–91, January 1996.

[6] J. Demsar and B. Zupan. Orange: From Experimental Machine Learning to Interactive Data Mining. White Paper, Faculty of Computer and Information Science, University of Ljubljana, 2004.

[7] R. O. Duda, P. E. Hart, and D. G. Stork. *Pattern Classification.* Wiley-Inter science, 2nd edition, 2001.

[8] F.Ginter, J. Boberg, J. Jarvinen, andT. Salakoski. New Techniques for Disambiguation in Natural Language and Their Application to Biological Texts. *Journal of Machine Learning Research,* Pages 605–621, June 2004.

[9] T.K. Jenssen, J. Komorowski, and E. Hovig. A literature network of human genes for high throughput analysis of gene expression. *Nature Genetics,* Pages 21–28, May 2001.

[10] S. Jiampojamarn. Biological term extraction using classification methods. Presentation at Dalhousie Natural Language Processing Meeting, June 2004.

[11] B.V Subba Rao, Dr. K. Sambasiva Rao. "*Automatic information Mining from Biomedical literature using NLU*", In International Journal on Computer Engineering and Information technology (IJCEIT), Volume-06, No.10, Pages 30-35, 2009.

[12] B.V Subba Rao, Dr, K Sambasiva Rao. "*Semantic Explanation of Biomedical text using Google*". In Proceedings of the International Conference on Web Sciences-09, Pages 452-457 January 2009.

[13] PubMed Medline. URL: www.pubmed.gov

[14] Medline Abstracts. URL: http://medline.cos.com

Author's Biography

B.V.Subba Rao, presently working as Associate Professor in P.V.P Siddhartha Institute of Technology Vijayawada, affiliated to Jawaharlal Nehru Technological University. He received his M.Tech degree with distinction in Computer Science and Engineering from Acharya Nagarjuna University. He is pursuing Ph.D in Computer Science and Engineering at Acharya Nagarjuna University, Guntur. He has guided 30 post Graduated and 40 graduate projects. He has published 2 papers (National / Conference Proceedings) and has Academic participation in 24 International / National Seminars / workshops and Conferences. He is a member of Computer Society of India (CSI), Association for Computing Machinery (ACM), and Indian Society for Technical Education (ISTE). His current research interests are in the areas of Artificial Intelligence, Natural Language Processing and Information Retrieval systems.

Dr.K.V.Sambasiva Rao, presently working as a Principal of MVR College of Engineering and Technology, Paritala. He pursued his M.E from BITS, Pilani and Doctorate from IIT Delhi. He has a total of 21 years of rich experience comprising teaching, research and industry. He has published 4 books, 18 papers in international and national journals. He has conducted numerous national conferences, workshops with the support of AICTE, DST and other government bodies. He has given more than 50 seminar talks at various technical institutions. He has guided 25 Masters level projects and is research director for 11 Ph.D candidates. His biography was included in MARQUI'S INTERNATIONAL, New Jersey, USA "Who is who in the World" in the year 1999 and was awarded "Outstanding achievement award" by International Biography Centre, Cambridge, UK. He is the life member of 3 professional bodies.