Evaluating Protein Name Recognition: An Automatic Approach

Alexander K. Seewald[1]

Austrian Research Institute for Artificial Intelligence, Freyung 6/VI/7,
A-1010 Vienna, Austria alexsee@oefai.at

Abstract. In some domains, named entity recognition might be considered a solved problem. This does not hold for biological text mining, where protein and gene name recognition are still open research problems [4, 6]. In this paper, we compare two current approaches to the problem of protein name recognition, KeX [5] and Yapex [4]. Unlike manual evaluation which relies on domain experts' judgement concerning position and extent of all relevant names and entails a high workload, our comparison methodology is fully automatic. Our results agree with previous manual evaluations of KeX and Yapex which validates our approach.

1 Introduction

While for some tasks named entity recognition may be considered a solved problem, this is not the case for protein name recognition in biological text mining. Although useful results can be achieved with a fixed static dictionary [1, 8] thousands of new papers appear daily and no fixed dictionary is expected to be accurate for long. Current implementations of protein name recognizers rely on heuristics and proprietary text processing, and do not yet learn, although they are quite able to recognize previously unseen protein names. Machine learning approaches to this problem are also upcoming[1]. In this paper, we focus on two approaches which are already available, KeX[2] [5] and Yapex[3] [4].

A reasonable approach to compare protein name recognizers is to measure them against the gold standard – a domain expert which marks up all protein names in a set of test papers. While this is not completely unproblematic – domain experts may sometimes disagree to the extent of protein names or what kinds of names are considered valid [4] – it entails a high workload which is better put to more productive use. Therefore, we propose a fully automatic approach to comparison, using the full SWISS-PROT [7, 2] database and associated MEDLINE[4] publications as approximation to the gold standard. Our approach allows

[1] GAPSCORE, http://bionlp.standford.edu, which we look forward to investigate.

[2] The KeX source code has been made freely available by the authors at http://www.hgc.ims.u-tokyo.ac.jp/service/tooldoc/KeX/intro.html

[3] Yapex is available via web form http://www.sics.se/humle/projects/prothalt/yapex.cgi and utilizes a commercial tagger which is subject to licensing.

[4] MEDLINE is a bibliographic database owned by the U.S. National Library of Medicine and can be searched via PubMed: http://www.ncbi.nlm.nih.gov/entrez/query.fcgi

to utilize several orders of magnitude more text for evaluation than manual approaches.[5] While we cannot offer specific answer keys, and our approach is only based on positive examples of protein names, our results agree with an earlier comparison of KeX and Yapex by classical means.

2 Databases

SWISS-PROT is one of the largest proteomics databases. All its entries are created by biologists; continually updated, extended and corrected.

For our experiments, we obtained a recent snapshot of the SWISS-PROT database, consisting of 121,745 entries. We also obtained all referenced MEDLINE entries, yielding 83,044 documents. For our experiments, we focus on the DE field which encodes protein names and synonyms; and use it as standard against which different protein name recognizers are measured, using the referenced MEDLINE entries as input. Since MEDLINE publications are expected to reference other proteins as well, we also measured each recognizer against a full list of proteins names generated from the whole SWISS-PROT database.

All in all, there are 95,982 unique protein names and synonyms referenced in our SWISS-PROT snapshot. This is less than the total number of entries, so not all proteins have an unique name. Non-specific entries such as 111 kDa protein, which refers to any protein with a specific molecular weight, explain this discrepancy. On average, each SWISS-PROT entry includes 2.34 ± 1.52 protein names and synonyms.

From each MEDLINE publication, we chose title and abstract. For KeX, we compiled KeX and executed it locally; for Yapex we are obliged to Kristofer Franzen, who supervised the processing at their site. Both KeX and Yapex are quite fast; parsing all 83,044 MEDLINE documents took about a working day, i.e. 8-10h. Thus, parsing all new MEDLINE entries in real-time seems feasible.

3 Experiments

Extracting the protein names from KeX and Yapex output is quite simple – both use html-like tags which enclose protein names and parts. While KeX outputs a single level of tags, those by Yapex are sometimes recursively nested. For the latter, we chose to use the full word sequence within the outermost level of tags.

Matching recognized protein names to SWISS-PROT protein names is less obvious. Inspired by [4], we have considered three matching schemes:

- *Strict*, i.e. matching of the whole string. We observed slight differences in capitalization and therefore chose to use case-insensitive matching.
- *Protein Name Parts* (*PNP*), i.e. a degree of match between 0 and 1 as ratio of those words within the recognized protein which are matched to any words in

[5] We *do* indirectly rely on SWISS-PROT curators painstakingly collecting protein names – so even our approach heavily relies on reusing human expertise.

Table 1. This table shows results for KeX and Yapex with different comparison methodologies, as well as vs. all of SPROTs protein names or vs. only those which are associated to a given MEDLINE publication. Better values are shown in **bold**.

	all SPROT		only SPROT refs	
	Yapex	KeX	Yapex	KeX
Strict	**0.202**±0.401	0.097±0.296	**0.077**±0.267	0.038±0.192
PNP	**0.606**±0.423	0.529±0.374	**0.328**±0.426	0.221±0.346
Sloppy	0.732±0.443	**0.775**±0.420	**0.415**±0.493	0.354±0.478

a given SWISS-PROT template. Word matching is done both with lowercase and uppercase first letter to account for differences in capitalization. We rely on the protein recognizer to emphasize word boundaries, which we consider to be whitespace, or parentheses wrapped in whitespace.

- *Sloppy*, i.e. a match is counted even if only a single word matches. This is equivalent to $PNP>0$.

The *Left* and *Right* schemes from [4] are not applicable since we have no information on the position of a probable match. Furthermore, our protein list is not exhaustive, so we have no complete data on non-matches.

Along an orthogonal dimension, we compared each protein name recognizer against a full list of all unique SWISS-PROT protein names (*all SPROT*); or only against those SWISS-PROT names which are associated with the given MEDLINE publication (*only SPROT refs*). The latter is considered to be more specific – it is expected that at least one entry from this list appears in every MEDLINE publication. This is almost the case for Yapex with 0.81±1.93 entries per publication; and less so for KeX with 0.46±1.33. Therefore, not all synonyms are recorded in SWISS-PROT, which agrees with our domain experts stated opinion that the protein lists from SWISS-PROT are not exhaustive.

Table 1 gives the results. We give averages and standard deviations over the match values from all recognized proteins, based on the three matching schemes. A match is considered 1 and no match 0 for *Strict* and *Sloppy*; and a ratio between 0 and 1 for *PNP*. We see that *Sloppy* is the only matching scheme where KeX performs comparable to Yapex, which was also found in [4]. Under the other matching schemes, Yapex performs better. If we consider *only SPROT refs*, Yapex always performs better than KeX.

We take a closer look at the distributions of *PNP* match values in Figure 1. Alas, the most obvious patterns are just artefacts of ratio scores – short protein names mean a smaller number of possible scores, since each word can only match or not. Since there are many short protein names, the values tend to cluster near simple ratios, such as $\frac{1}{2}$, $\frac{1}{3}$, $\frac{2}{3}$ and so on. For longer protein names, the number of possible scores increases, so longer names tend to distribute over a wider area and are more susceptible to disappear into the background.

There *are* some patterns, though. The match values for KeX are biased towards smaller values, indicating that its protein names contains superfluous parts and are thus on average too long, which was also found by [4]. The former is

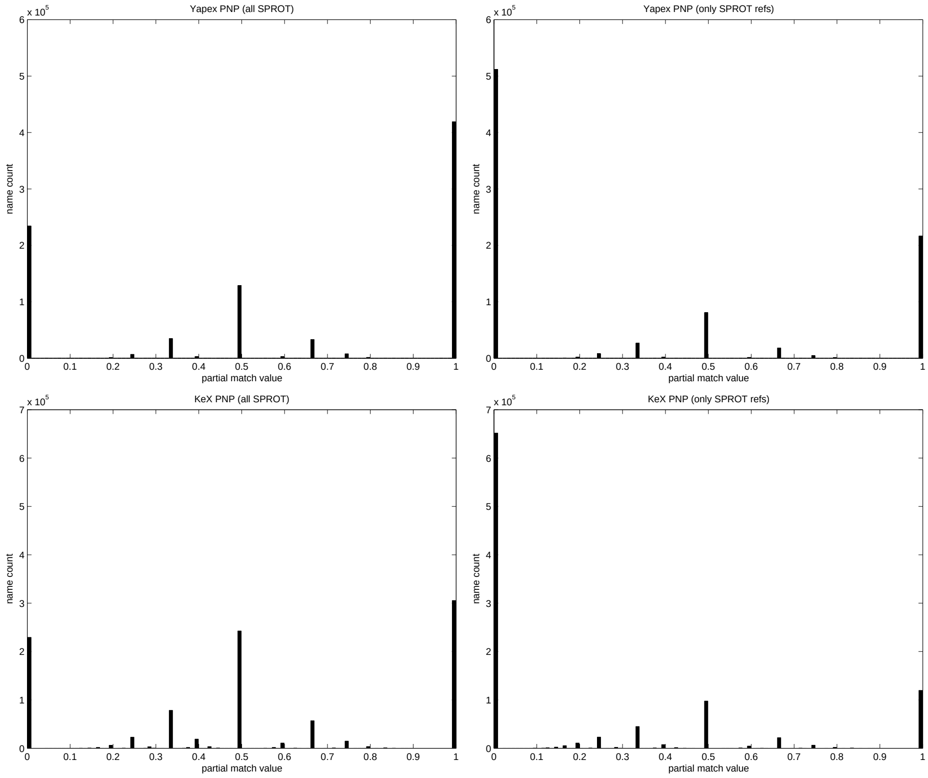

Fig. 1. This figure shows histograms for the PNP comparison: Yapex on top, KeX on the bottom. The figures on the left show results vs. all of SPROT; those on the right, vs. only references SPROT entries.

also indicated by the average length of recognized proteins: for Yapex, this is 1.59±0.95 which for KeX it is much higher at 2.17±1.56. For Yapex, the values are quite symmetric in their arrangement between match values 0 and 1.

4 Related Research

[4] gives an excellent overview on the challenges and issues of protein name recognition; and also compares Yapex to KeX on manually annotated data. We found their discussion of comparison methodologies for protein names quite enlightening.

[5] introduces a system to extract protein names, which has been extended towards KeX. They report excellent results, which have been called in question by [3, 4], and incidentially also by our work here.

5 Conclusion

We have applied an automatic comparison methodology on two protein name recognizers. We were able to validate some conclusions from an earlier manual comparison of the same two recognizers by [4], concerning the comparable performance of KeX and Yapex when compared via *Sloppy*, and the overlong matches of KeX.

We look forward to compare all current approaches within the same methodology. Ultimately, we hope to create useful training sets; both for our own machine learning approach to protein name recognition which will be made freely available, as well as for benchmarking.

Acknowledgements

This research was supported by the European Union under project no. QLRI-2002-02770 (BioMinT). The Austrian Research Institute is supported by the Austrian Federal Ministry of Education, Science and Culture. We want to thank Kristofer Franzen for supplying the results for Yapex, so we did not have to push 83,044 MEDLINE publications through their web form; and to the team of [5] for making their system publicly available, although it does have some minor bugs.

References

1. Blaschke, C., Andrade, M.A., Ouzounis, C., Valencia, A. (1999) Automatic Extraction of biological information from scientific text: protein-protein interactions, in 7th International Conference on Intelligent Systems in Molecular Biology, (ISMB'99), Heidelberg, pp.60-67.
2. Boeckmann B., Bairoch A., Apweiler R., Blatter M.-C., Estreicher A., Gasteiger E., Martin M.J., Michoud K., O'Donovan C., Phan I., Pilbout S., Schneider M. (2003) The SWISS-PROT protein knowledgebase and its supplement TrEMBL in 2003, Nucleic Acids Research, 31(1):365-370.
3. deBrujin, B., Martin, J. (2000) Protein name tagging, presented as a poster at the 8th International Conference on Intelligent Systems for Molecular Biology (ISMB'00)
4. Franzen K., Eriksson G., Olsson F., Asker L., Liden P., Coester J. (2002) Protein names and how to find them, International Journal of Medical Informatics, Vol. 67/1-3, Special Issue on NLP in Biomedical Applications, pp.49-61.
5. Fukuda, K., Tsunoda, T., Tamura, A., Takagi, T. (1998) Toward information extraction: identifying protein names from biological papers, in Pacific Symposium on Biocomputing, pp.705-716.
6. Hanisch, D., Fluck, J., Mevissen, H.-T., Zimmer, R. (2003) Playing Biology's Name Game: Identifying Protein Names in Scientific Text, Proceedings of the Pacific Symposium on Bioinformatics (PSB), 2003.
7. O'Donovan,C., Martin,M.J., Gattiker,A., Gasteiger,E., Bairoch,A. and Apweiler,R. (2002) High-quality protein knowledge resource: SWISS-PROT and TrEMBL. Briefings in Bioinformatics, *3*, 275–284.
8. Ono, T., Hishigake, H., Tanigami, A., Takagi, T. (2001) Automated extraction of information on protein-protein interactions from the biological literature, Bioinformatics, 17(2), 155–161.