Combined Conditional Random Fields and n-Gram Language Models for Gene Mention Recognition

Craig A. Struble[1]
craig.struble@marquette.edu

Richard J. Povinelli[2]
richard.povinelli@marquette.edu

Michael T. Johnson[2]
mike.johnson@marquette.edu

Dina Berchanskiy[1]
dina.berchanskiy@marquette.edu

Jidong Tao[2]
jidong.tao@marquette.edu

Marek Trawicki[2]
marek.trawicki@marquette.edu

[1] Department of Mathematics, Statistics and Computer Science, P.O. 1881, Milwaukee, WI 53201-1881, USA
[2] Department of Electrical and Computer Engineering, P.O. 1881, Milwaukee, WI 53201-1881, USA

Abstract

In this paper, we propose the use of character n-gram and multiple conditional random field (CRF) models for BioCreAtIvE 2 Task 1, gene/protein name recognition. We investigated different state transition weighting schemes for CRFs and discovered that models provided independent non-overlapping mentions. To improve recall, the results of multiple models are combined. To improve precision, character n-gram models classify gene/protein mention containing sentences. Our best approach achieved a precision of 84.35%, recall of 81.39% and F-measure of 82.85%.

Keywords: conditional random field, named entity recognition, n-gram models

1 Introduction

Effective automated tools for identifying gene mentions can help in rapidly creating large gene centric knowledge bases, identifying associations betweens genes and diseases, and indexing biomedical literature by genes and their products. In 2006, the BioCreAtIvE 2 community challenge provided training, development and evaluation data to critically assess information extraction techniques for several text processing tasks motivated by the biological community [2, 3].

In this paper, we present a method for identifying gene/protein mentions using multiple conditional random field (CRF) [4] and n-gram language models. Our system is similar to McDonald and Pereira's CRF-based tagger in the first BioCreAtIvE contest [6], but utilizes different features and combines multiple models. Other CRF-based tagging systems for biological named entity recognition include ABNER [8] and GeneTaggerCRF [9]. Systems primarily differ in their choice of features, CRF parameters and training data, while achieving similar performance.

The system is described in more detail in Section 2. Evaluation of the system and a brief discussion is in Section 3.

2 System Description

Our system treats the problem of identifying gene/protein names as one of tagging a sequence of tokens with labels indicating the location of gene/protein mentions. Sentences are tokenized into numbers with optional decimals and leading + or -, alphanumeric strings with single quotes (to create tokens such as 5'), and individual punctuation marks. For training and tagging, tokens are labeled with one of three labels *B-GENE*, *I-GENE*, and O representing the beginning, inside and outside of a gene mention.

Conditional Random Fields Gene mention tagging employs linear-chain conditional random fields (CRFs), a conditional probability model for tagging sequences [4]. The conditional probability $P(\mathbf{s}|\mathbf{o})$ of a state sequence $\mathbf{s} = s_1, ..., s_n$ corresponding to labels given the observed token sequence $\mathbf{o} = o_1, ..., o_n$ is defined by

$$P(\mathbf{s}|\mathbf{o}) = \frac{1}{Z(\mathbf{o})} \exp\left(\sum_{i=1}^{n}\sum_{j=1}^{m} \lambda_j f_j(\mathbf{s}, \mathbf{o}, i)\right),$$

where $Z(\mathbf{o})$ is a normalization factor over all state sequences, $f_j(\mathbf{s}, \mathbf{o}, i)$ is a feature function and λ_j is a learned feature weight. The feature functions are written in their most general form.

We developed two CRF models with different Markov-order structures. One is a second-order structure, evaluating the feature function using the current and previous states. Feature functions are represented by $f_j(s_{i-1}, s_i, \mathbf{o}, i)$. The second is a first-order structure, evaluating feature functions in the context of only the current state. Feature functions are represented by $f_j(s_i, \mathbf{o}, i)$. This second model is also known as a *half label* model in the MALLET library [5].

Combining CRF models When evaluating the two CRF models, we noted that performance was similar but the models identified independent non-overlapping gene name mentions. This observation led us to combine the two CRF models using a simple approach in the hopes of improving recall without impacting precision too much. To combine models, one CRF model is chosen as the baseline tagger. The second model is used to assign gene mentions that do not overlap at all with the baseline tagger.

Character n-gram Models In some cases, sentences not containing mentions were tagged. This typically happens when orthographic features of a token strongly indicate that the token is part of a gene mention (e.g., all capital letters). To improve precision, a 6-gram character language model predicted whether or not a sentence contains a gene mention. The n-gram classifier uses untokenized sentences as input. When the n-gram model is used, only sentences predicted to contain mentions are tagged by CRF models.

Features We utilized boolean features of the text being labeled. Orthographic features were used including: the token itself, all capital letters, all lowercase letters, punctuation, quote, alphanumeric, lowercase letters followed by capital letters, initial capital letter, single capital letter, single letter, all alphabetic, single digit, double digits, integer, real number, contains a digit, three letter amino acid code, contains *globin* or *globulin*, contains a Roman numeral, or contains a Greek letter. Additional features included all prefixes and suffixes of lengths 2–4 and whether a token is part of a short form or long form of an abbreviation definition [7]. Contextual features included all features of the 2 preceding and 2 following tokens.

Post Processing A simple post-processing step was used to ignore gene mentions that contained mis-matched parentheses, which indicated a tagging mistake

Implementation The system was implemented in Java using the MALLET [5] and LingPipe [1] libraries.

3 Results and Discussion

During development, the provided set of 15,000 sentences was split into a training set and test set containing 10,500 and 4,500 sentences respectively. For the final submission, all 15,000 sentences were used for training and testing was performed on a blind collection of 5,000 sentences. Precision, recall,

Table 1: System performance on test data. Quartile placement is shown in parentheses.

Submission	Precision	Recall	F-Measure
Combined CRFs without n-gram	84.35 (3)	81.39 (2)	82.85 (2)
Combined CRFs and n-gram	87.53 (1)	77.52 (3)	82.22 (2)
Second-order and n-gram	88.88 (1)	76.02 (3)	81.95 (2)

F-Measure and quartiles for each submission are in Table 1. The results are comparable to McDonald and Pereira [6], with slight improvements in recall and F-measure. As hoped, the combined CRFs improve recall without impacting precision too much. The n-gram models improve precision and may be desirable in situations where mislabeling is problematic.

Two classes of gene mentions were problematic. The first was due to gene mention coordination, such as in *clotting factors II, V, VIII, IX, X*. Often only the first part, *clotting factors II*, was tagged resulting in a false positive and false negative contributions. The second was due to parenthesized tokens embedded in the mention, such as in *serum neutralizing (SN) antibody*. Often, the first part, *serum neutralizing*, the part preceding the closing parenthesis, *serum neutralizing (SN*, or the part following the opening parenthesis *SN) antibody*, was tagged. Apparently, clear cues for the proper tagging of parentheses, which are included sometimes, are not learned.

In summary, we obtained modest improvements in recall and F-measure by combining multiple CRFs. Recall and precision could be improved by investing more effort in handling coordination and mentions with embedded parenthesized terms.

References

[1] alias i. LingPipe. http://www.alias-i.com/lingpipe/index.html, 2006. Version 2.3.0.

[2] BioCreAtIvE II: Critical assessment for information extraction in biology challenge (2006–2007). http://biocreative.sourceforge.net/biocreative_2.html.

[3] L. Hirschman, A. Yeh, C. Blaschke, and A. Valencia. Overview of BioCreAtIvE: critical assessment of information extraction for biology. *BMC Bioinformatics*, 6 Suppl 1:S1, 2005.

[4] J. Lafferty, A. McCallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proc. 18th International Conf. on Machine Learning*, pages 282–289. Morgan Kaufmann, San Francisco, CA, 2001.

[5] A. K. McCallum. MALLET: A machine learning for language toolkit. http://mallet.cs.umass.edu, 2002.

[6] R. McDonald and F. Pereira. Identifying gene and protein mentions in text using conditional random fields. *BMC Bioinformatics*, 6 Suppl 1:S6, 2005.

[7] A. S. Schwartz and M. A. Hearst. A simple algorithm for identifying abbreviation definitions in biomedical text. *Pac Symp Biocomput*, pages 451–462, 2003.

[8] B. Settles. Biomedical Named Entity Recognition using Conditional Random Fields and Rich Feature Sets. In *COLING 2004 International Joint workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP) 2004*, pages 107–110, 2004.

[9] R. Talreja. GeneTaggerCRF. http://www.cis.upenn.edu/datamining/software_dist/biosfier/.