Rich Feature Set, Unification of Bidirectional Parsing and Dictionary Filtering for High F-Score Gene Mention Tagging

Cheng-Ju Kuo[1] cju.kuo@gmail.com
Yu-Ming Chang[2] porter@iis.sinica.edu.tw
Han-Shen Huang[2] hanshen@iis.sinica.edu.tw
Kuan-Ting Lin[2] woody@iis.sinica.edu.tw
Bo-Hou Yang[2,3] ericyang@iis.sinica.edu.tw
Yu-Shi Lin[2] bathroom@iis.sinica.edu.tw
Chun-Nan Hsu[2] chunnan@iis.sinica.edu.tw
I-Fang Chung[1] ifchung@ym.edu.tw

[1] Institute of Bioinformatics, National Yang-Ming University, Taipei, Taiwan
[2] Institute of Information Science, Academia Sinica, Taipei, Taiwan
[3] Department of Electrical Engineering, Chang-Gung University, TaoYuan, Taiwan

In the first BioCreative (2004) [3], conditional random fields (CRFs) [5] were employed in tagging gene and protein mentioned in the biomedical text with high performance [8]. Therefore, we chose CRFs as our starting point and carefully selected a rich set of 5,059,368 predicates as the features. To further improve its performance, we combined the tagging results of forward and backward parsing [4]. We tried different combination methods, including set operations and Co-Training [1]. However, we found that Co-Training performed poorly. Instead, we selected the best solutions from the "adjacent" ten candidates of bidirectional parsing and then applied dictionary filtering to obtain the best F-score result. Details are given as follows.

We applied MALLET [7] to take advantage of its feature induction capability [6]. Due to the special characteristics of name-entities of genes and gene products [10], a rich set of features is required. Not all features proposed in previous work are useful. After hundreds of trials, we carefully selected predicates shown in Table 1 as our feature set, which includes commonly used orthographic predicates and character-n-gram predicates for $2 \leq n \leq 4$ [8]. We used $\{-2, -1, 0, 1, 2\}$ as the offsets and evaluated predicates such as word, stemmed word, part-of-speech tag, and word morphology as the contextual features at each position. Our domain-specific features include nucleotide (i.e., types of DNA or RNA), residues of amino acids, etc. We excluded prefix and suffix predicates used in previous work because we found that they usually increase false positive. To extract features, the Genia Tagger [9] was applied for stemming, tokenization and part-of-speech tagging. We modified the Genia Tagger slightly to tokenize words with a higher granularity. For example, punctuation symbols within words were segmented. We also applied a rule-based filter to clean up some easily fixed mistakes, such as entities with unpaired parentheses or square brackets.

The performance of the CRF models with this feature set and the rule-based filter is given in the first row of Table 2, which is already slightly better than previously reported figures. These inside test results were obtained by randomly selected 10,000 sentences for training and the rest for testing from the training data set provided by the organizers. To further improve its performance, we combined the tagging results of forward and backward parsing. In forward parsing, the tagger reads and tags the input sentences from left to right, while in backward parsing, the tagger reads and tags the input sentences from right to left. Note that the training set and the features must be reversed to train a backward parsing CRF model. We tested the forward and backward parsing models and found that backward parsing constantly outperformed forward parsing in both recall and precision, but its

Table 1: Features.

Feature	Example	Feature	Example	Feature	Example
Word	proteins	Hyphen	-	Nucleoside	Thymine
StemmedWord	protein	BackSlash	/	Nucleotide	ATP
PartOfSpeech	NN	OpenSqure	[	Roman	I, II, XI
InitCap	Kinase	CloseSqure	]	MorphologyTypeI	p53→p*
EndCap	kappaB	Colon	:	MorphologyTypeII	p53→a1
AllCaps	SOX	SemiColon	;	MorphologyTypeIII	GnRH→AaAA
LowerCase	interlukin	Percent	%	WordLength	1, 2, 3-5, 6+
MixCase	RalGDS	OpenParen	(	N-grams(2-4)	p53→{p5, 53}
SingleCap	kDa	CloseParen	)	ATCGUsequece	ATCGU
TwoCap	IL	Comma	,	Greek	alpha
ThreeCap	CSF	FullStop	.	NucleicAcid	cDNA
MoreCap	RESULT	Apostrophe	’	AminoAcidLong	tyrosine
SingleDigit	1	QuotationMark	”	AminoAcidShort	Ser
TwoDigit	22	Star	*	AminoAcid+Position	Ser150
FourDigit	1983	Equal	=		
MoreDigit	513256	Plus	+		

Table 2: System performance in inside test.

System	Precision	Recall	F-Measure
Forward	0.8660	0.8077	0.8359
Backward	0.8733	0.8118	0.8414
Union	0.8349	0.8578	0.8462
Intersection	0.9076	0.7186	0.8021
Adjacent Ten Union + Dictionary	0.8773	0.8263	0.8510

reason is unclear. We assume that some “signals” at the end of entities are more important to well demarcate boundaries of entities. However, distributions of nonzero features in both parsing directions show no significant difference (data not reported here). Then, we tried different ways to combine the bidirectional tagging results. Simple set operations failed to improve the performance. Though recall may be enhanced by union and precision by intersection, they also degraded the other measure and the F-score. Table 2 shows their inside test F-scores. We tried to apply Co-Training [1]. However, since the output scores (negative log likelihood) of MALLET were not reliable to select unlabeled training data, Co-Training seriously degraded the F-score to as low as 0.6.

Meanwhile, we found that the union of the “adjacent” ten tagging solutions of bidirectional parsing may achieve a nearly perfect recall (0.9810 for the final test, with 0.1387 precision). That is, nearly all true positives are in this union. The “adjacent” solutions were obtained by MALLET’s `n-best` option. However, we found that the solutions are not actually the best n solutions. Instead, they are candidate tagging results adjacent to the best tagging in the search tree grown by the A^* search algorithm, according to our trace of MALLET’s source code. This also explains why its output score ranking is not appropriate for Co-Training. In fact, exhaustively search for the best n candidates is intractable. Nevertheless, knowing that nearly all true positives are actually in the union of the adjacent ten solutions, we distill real true positives from this union as follows.

1. Parse the input sentence in both directions to obtain the adjacent ten solutions for each direction

Table 3: System performance of submitted runs.

System	Precision	Recall	F-Measure
Backward	0.8930	0.8383	0.8648
Union	0.8610	0.8708	0.8658
Adjacent Ten + Dictionary	0.8930	0.8449	**0.8683**

with their output scores;

2. Compute the intersection of bidirectional parsing and select the solution in the intersection that minimizes the sum of its output scores;

3. For the other 18 solutions, select the labeled terms appearing in a dictionary with its length greater than three.

We used approved gene symbols and aliases obtained from HUGO [2] as our dictionary for the final dictionary filtering. We submitted the results of the top three performing methods in our inside test (see Table 2) for the 2nd BioCreative (2006). Their performances are shown in Table 3.

References

[1] Blum, A. and Mitchell, T. Combining labeled and unlabeled data with co-training. In *COLT: Proceedings of the Workshop on Computational Learning Theory, Morgan Kaufmann Publishers*, pages 92–100, 1998.

[2] Eyre, T. A., Ducluzeau, F., Sneddon, T. P., Povey, S., Bruford, E. A., and Lush, M. J. The HUGO gene nomenclature database, 2006 updates. *Nucleic Acids Research*, 34:D319–D321, 2006.

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

[4] Kudo, T. and Matsumoto, Y. Chunking with support vector machines. In *Second Meeting of the North American Chapter of the Association for Computational Linguistics*, 2001.

[5] Lafferty, J., McCallum, A., and Pereira, F. 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.

[6] McCallum, A. Efficiently inducing features of conditional random fields. In *Nineteenth Conference on Uncertainty in Artificial Intelligence (UAI03)*, 2003.

[7] McCallum, A. K. MALLET: A Machine Learning for Language Toolkit. http://mallet.cs.umass.edu, 2002.

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

[9] Tsuruoka, Y., Tateishi, Y., Kim, J.-D., Ohta, T., McNaught, J., Ananiadou, S., and Tsujii, J. Developing a Robust Part-of-Speech Tagger for Biomedical Text. In *Advances in Informatics - 10th Panhellenic Conference on Informatics*, pages 382–392, 2005.

[10] Zhou, G. D. Recognizing names in biomedical texts using mutual information independence model and SVM plus sigmoid. *International Journal of Medical Informatics*, 75:456–467, 2006.