Extracting Protein-Protein Interaction based on Discriminative Training of the Hidden Vector State Model

Deyu Zhou and Yulan He
Informatics Research Centre, The University of Reading, Reading RG6 6BX, UK
Email:d.zhou@reading.ac.uk, y.he@reading.ac.uk

1 Introduction

The knowledge about gene clusters and protein interactions is important for biological researchers to unveil the mechanism of life. However, large quantity of the knowledge often hides in the literature, such as journal articles, reports, books and so on. Many approaches focusing on extracting information from unstructured text, such as pattern matching, shallow and deep parsing, have been proposed especially for extracting protein-protein interactions (Zhou and He, 2008).

A semantic parser based on the Hidden Vector State (HVS) model for extracting protein-protein interactions is presented in (Zhou et al., 2008). The HVS model is an extension of the basic discrete Markov model in which context is encoded as a stack-oriented state vector. Maximum Likelihood estimation (MLE) is used to derive the parameters of the HVS model. In this paper, we propose a discriminative approach based on parse error measure to train the HVS model. To adjust the HVS model to achieve minimum parse error rate, the generalized probabilistic descent (GPD) algorithm (Kuo et al., 2002) is used. Experiments have been conducted on the GENIA corpus. The results demonstrate modest improvements when the discriminatively trained HVS model outperforms its MLE trained counterpart by 2.5% in F-measure on the GENIA corpus.

2 Methodologies

The Hidden Vector State (HVS) model (He and Young, 2005) is a discrete Hidden Markov Model (HMM) in which each HMM state represents the state of a push-down automaton with a finite stack size.

Normally, MLE is used for generative probability model training in which only the correct model needs to be updated during training. It is believed that improvement can be achieved by training the generative model based on a discriminative optimization criteria (Klein and Manning, 2002) in which the training procedure is designed to maximize the conditional probability of the parses given the sentences in the training corpus. That is, not only the likelihood for the correct model should be increased but also the likelihood for the incorrect models should be decreased.

Assuming the most likely semantic parse tree $\hat{C} = C_j$ and there are altogether M semantic parse hypotheses for a particular sentence W, a parse error measure (Juang et al., 1993; Chou et al., 1993; Chen and Soong, 1994) can be defined as

$$d(W) = -\log P(W, C_j) + \log[\frac{1}{M-1} \sum_{i, i \neq j} P(W, C_i)^{\eta}]^{\frac{1}{\eta}} \quad (1)$$

where η is a positive number and is used to select competing semantic parses. When $\eta = 1$, the competing semantic parse term is the average of all the competing semantic parse scores. When $\eta \rightarrow \infty$, the competing semantic parse term becomes $\max_{i.i \neq j} P(W, C_i)$ which is the score for the top competing semantic parse. By varying the value of η, we can take all the competing semantic parses into consideration. $d(W) > 0$ implies classification error and $d(W) \leq 0$ implies correct decision.

The sigmoid function can be used to normalize $d(W)$ in a smooth zero-one range and the loss function is thus defined as (Juang et al., 1993):

$$\ell(W) = \text{sigmoid}(d(W)) \quad (2)$$

BioNLP 2008: Current Trends in Biomedical Natural Language Processing, pages 98–99,
Columbus, Ohio, USA, June 2008.

where

$$\text{sigmoid}(x) = \frac{1}{1 + e^{-\gamma x}} \quad (3)$$

Here, γ is a constant which controls the slope of the sigmoid function.

The update formula is given by:

$$\lambda^{k+1} = \lambda^k - \epsilon^k \nabla \ell(W_i, \lambda^k) \quad (4)$$

where ϵ^k is the step size.

Using the definition of $\ell(W_i, \lambda^k)$ and after working out the mathematics, we get the update formulae 5, 6, 7,

$$\left(\log P(n|\mathbf{c}')\right)^* = \log P(n|\mathbf{c}') - \epsilon\gamma\ell(d_i)(1 - \ell(d_i)) \left(-I(C_j, n, \mathbf{c}') + \sum_{i, i \neq j} I(C_i, n, \mathbf{c}') \frac{P(W_i, C_i, \lambda)^\eta}{\sum_{i, i \neq j} P(W_i, C_i, \lambda)^\eta}\right) \quad (5)$$

$$(\log P(c[1]|c[2..D]))^* = \log P(c[1]|c[2..D]) - \epsilon\gamma\ell(d_i)(1 - \ell(d_i)) \left(-I(C_j, c[1], c[2..D]) + \sum_{i, i \neq j} I(C_i, c[1], c[2..D]) \frac{P(W_i, C_i, \lambda)^\eta}{\sum_{i, i \neq j} P(W_i, C_i, \lambda)^\eta}\right) \quad (6)$$

$$(\log P(w|\mathbf{c}))^* = \log P(w|\mathbf{c}) - \epsilon\gamma\ell(d_i)(1 - \ell(d_i)) \left(-I(C_j, w, \mathbf{c}) + \sum_{i, i \neq j} I(C_i, w, \mathbf{c}) \frac{P(W_i, C_i, \lambda)^\eta}{\sum_{i, i \neq j} P(W_i, C_i, \lambda)^\eta}\right) \quad (7)$$

where $I(C_i, n, \mathbf{c}')$ denotes the number of times the operation of popping up n semantic tags at the current vector state $\mathbf{c}'$ in the C_i parse tree, $I(C_i, c[1], c[2..D])$ denotes the number of times the operation of pushing the semantic tag $c[1]$ at the current vector state $c[2..D]$ in the C_i parse tree and $I(C_i, w, \mathbf{c})$ denotes the number of times of emitting the word w at the state $\mathbf{c}$ in the parse tree C_i.

3 Experimental Setup and Results

GENIA (Kim et al., 2003) is a collection of 2000 research abstracts selected from the search results of MEDLINE database using keywords (MESH terms) *"human, blood cells and transcription factors"*. All these abstracts were then split into sentences and those containing more than two protein names and at least one interaction keyword were kept. Altogether 3533 sentences were left and 2500 sentences were sampled to build our data set.

The results using MLE and discriminative training are listed in Table 1. Discriminative training improves on the MLE by relatively 2.5% where N and I are set to 5 and 200 individually. Here N denotes the number of semantic parse hypotheses and I denotes the the number of sentences in the training data.

Table 1: Performance comparison of MLE versus Discriminative training

Measurement	GENIA	
	MLE	Discriminative
Recall	61.78%	64.59%
Precision	61.16%	61.51%
F-measure	61.47%	63.01%

References

J.K. Chen and F.K. Soong. 1994. An n-best candidates-based discriminative training for speech recognition applications. *IEEE Transactions on Speech and Audio Processing*, 2:206 – 216.

W. Chou, C.H. Lee, and B.H. Juang. 1993. Minimum error rate training based on n-best string models. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '93*, volume 2, pages 652 – 655.

Y. He and S. Young. 2005. Semantic processing using the hidden vector state model. *Computer Speech and Language*, 19(1):85–106.

B.H. Juang, W. Chou, and C.H. Lee. 1993. Statistical and discriminative methods for speech recognition. In Rubio, editor, *Speech Recognition and Understanding*, NATO ASI Series, Berlin. Springer-Verlag.

JD. Kim, T. Ohta, Y. Tateisi, and J Tsujii. 2003. GENIA corpus–semantically annotated corpus for bio-textmining. *Bioinformatics*, 19(Suppl 1):i180–2.

D. Klein and C. D. Manning. 2002. Conditional structure versus conditional estimation in nlp models. In *Proc. the ACL-02 conference on Empirical methods in natural language processing*, pages 9–16, University of Pennsylvania, PA.

H.-K.J. Kuo, E. Fosle-Lussier, H. Jiang, and C.H. Lee. 2002. Discriminative training of language models for speech recognition. In *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '02*, volume 1, pages 325 – 328.

Deyu Zhou and Yulan He. 2008. Extracting Interactions between Proteins from the Literature. *Journal of Biomedical Informatics*, 41:393–407.

Deyu Zhou, Yulan He, and Chee Keong Kwoh. 2008. Extracting Protein-Protein Interactions from the Literature using the Hidden Vector State Model. *International Journal of Bioinformatics Research and Applications*, 4(1):64–80.