Exploring Representations for Semantic-Rich Part of Speech Tagging

Weidong Qu[1, a], Sicong Yue[1, b]

[1]School of Information Engineering, Chang'an University, Xian, 710064, China

[a]email: qu.weidong@chd.edu.cn, [b]email:scyue@chd.edu.cn

Keywords: Part-of-Speech (POS); Treebank; Maximum Entropy; N-gram Model

Abstract. Part-of-speech (POS) tagging is the basic and primary analysis step in many natural language processing (NLP) applications. For English, it is often considered a solved problem. There are well established approaches, and the accuracy is around 97% with sufficient domain-specific training data. However, many NLP applications have very different special requirements, and the POS tageset has its own characteristics. These challenges can greatly affect the quality of the part-of-speech tagging process. To address these issues and achieve high POS tagging accuracy, we investigate the representations that can be applied to improve the performance of POS task. Our experiments show that the accuracy of POS tagging degrades significantly when tested with a large semantic and syntactic tagset. In addition, our analysis of experiments suggests that tokens rather than POS tags have more effect on tagging accuracy. Our best results were reached by using the most appropriate representations for POS tagging task.

Introduction

Part-of-speech tagging is the process of assigning one of the parts of speech or categories to a given token according to their contextual and grammatical properties [1, 2]. It is the basic and primary analysis step in many natural language processing applications. There are well established approaches, and accuracy is around 97% with sufficient domain-specific training data. Most related researches are conducted on data sets from the Penn Treebank. With this sufficient well-defined domain specific (financial news) training data, the tagger systems can yield state-of-art performance [3]. However, POS taggers have been increasingly applied to the Web, scientific domains, and other non-English languages among many other kinds of linguistic communication. These texts have their own characteristics different from the carefully edited financial news corpus. The performances of POS tagging are greatly influenced by these new challenges.

Previous work by Neil-Barrett et al. tried to deal with the cross-domain POS tagging problem. Traditionally, POS taggers are typically trained on linguistically annotated corpora and applied to the same domain. A difficulty with POS tagging task is that the domain specific training corpus cannot always be obtained. When the training data and applying tasks are from different domains, for example, training POS taggers on non-biomedical corpora and applying to biomedical corpora, the tagging accuracy decreases by approximately 10%. They observed that ignoring previously assigned POS tags and restricting the tagger's scope to the assigning token, previous token and following token can achieved significant performance improvement [4].

Researches in non-English POS tagging show that characteristics of language can also affect the quality of the POS tagging task [1, 2]. For example, Persian is a free order language, the adverbs can occur anywhere in the sentences without any changes in the meaning of the sentences, and many compound verbs can be separated by non-verbal elements [1]. The specific language characteristics make the POS tagging task more challengeable. By incorporating rich features into the POS tagger, their Persian POS tagging system outperforms the other state-of-the-art Persian taggers [1].

To the best of our knowledge, there are few researches at present investigating the POS tagging effectiveness on a large semantic and syntactic tagset. Although the state-of-the-art POS tagger on Penn Treebank can reach 97% accuracy [5], there are significant differences between the Penn Treebank and a large semantic and syntactic tagset. For example, the Penn Treebank has only 45

different tags, but the ATR Treebank has 3000 tags [6]. With such a large tageset, the training data insufficient problem becomes more serious, and on the other hand, semantic and syntactic tageset may also make the POS tagging task very challengeable.

In this work, we investigate the representations that can be applied to improve the performance of POS task on ATR Treebank. When adapting the established approaches to a large semantic tagset, the performances degrade significantly. By exploring representations for semantic-rich POS tagging task, we can achieve a significant increase in tagging accuracy. The experiments demonstrate that the tokens rather than POS tags contribute significantly to tagging accuracy improvement. This differs from the popular used HMM POS taggers that make significant use of POS tags and generally consider POS tags over an entire sentence through dynamic programming methods.

The remainder of this paper is organized as follows. In section II, we briefly introduce the POS tagging method in this work. Section III evaluates the effectiveness of our method. Finally, section IV presents our conclusion.

Method

To investigate what kind of information about the context can improve tagging accuracy, we used two different POS tagging approaches, Markov models, and maximum entropy models. Our response to this new POS tagging challenge is to seek new representations that allow systems to improve their performance. We pay special attention to the representations for POS tagging. We adopt a maximum entropy approach because it allows the inclusion of diverse sources of information without causing fragmentation and without necessarily assuming independence between the predictors [7]. In the following, the two tagging models are briefly explained.

The N-gram models (Markov models), such as TnT [8], we use here are second order models for part-of-speech tagging. The states of the models represent tags; outputs represent the words. Transition probabilities depend on the states, thus pairs of tags. Output probabilities only depend on the most recent category. The underlying model is as fowling form:

$$arg\ max_{t_1 \ldots t_T}\left[\prod_{i=1}^{T} P(t_i|t_{i-1}, t_{i-2})P(w_i|t_i)\right] P(t_{T+1}|t_T) \quad (1)$$

For a give sequence of words $w_1, w_2, \ldots, w_T$ of length T. $t_1, t_2, \ldots, t_T$ are elements of tagset. The additional tags t_{-1}, t_0, and t_{T+1} are beginning of sentence and the end of sentence markers. Using these additional markers can improve tagging results [8]. The transition and output probabilities are estimated from a tagged corpus.

Ratnaparkhi describes a maximum entropy approach to POS tagging [9]. The maximum Entropy Model is defined over H×T, where H is the set of possible word and tag contexts, or histories, and T is the set of allowable tags. The model parameters are learned from training data. The model's probability of a history *h* together with a tag *t* is as following form:

$$P(t|h) = \gamma \prod_{k=0}^{K} \alpha_k^{f_k(h,t)} p_0 \quad (2)$$

Where: ***t*** is tag we are predicting; ***h*** is the history of ***t***;
f is trigger functions have value 1 or 0; $\boldsymbol{\alpha}$ is the weight of trigger ***f***;
$\boldsymbol{\gamma}$ is a normalization coefficient; $\boldsymbol{P_0}$ is the default-tagging model

For maximum entropy based tagger, we use features very similar to the ones proposed in Ratnaparkhi's work. Our baseline model differs from Ratnaparkhi's model is that we do not use any information about the occurrence of words (except the word whose tag we are predicting) in the history or their properties. Table I shows the features we used in our baseline ME model:

TABLE I. DEFAULT FEATURES

Features	Description
$w = W \& t = T$	*where: w is the word to be tagged; t is tag we are predicting; t_{-1} is the tag to the left of tag t; t_{-2} is the tag to the left of tag t_{-1};*
$t_{-1} = X \& t = T$	
$t_{-2}t_{-1} = XY \& t = T$	

Experiments

The main objective of our work is to investigate the representations that can be applied to improve the performance of POS task on ATR Treebank. So we run our experiments under different conditions to investigate the advantages of different representations and their combination.

A. ATR Treebank

We use ATR Treebank as our dataset [6]. We split the dataset into two sets for training and testing. The training dataset consists of approximately 850,000 words. We test on a 53,000-word test treebank. For ATR Treebank, a very large, highly detailed part of speech tagset is used to label each word of each sentence with its syntactic and semantic categories.

B. Trigger Typel

We use mutual information to select the most useful trigger pairs from the bigram and trigram candidates. There are 18 trigger types in our model, the trigger types are shown in Table II:

TABLE II. LOCAL TRIGGER TYPES

No.	Feature templates	No.	Feature templates	No.	Feature templates	No.	Feature templates
1	w_{+1}	6	$t_{-2}t_{-1}t_0$	11	$w_0w_{+1}w_{+2}$	16	$t_{-1}w_0w_{+1}$
2	$t_{-1}w_0$	7	w_0w_{+1}	12	$w_{-2}w_{-1}w_0$	17	$t_{-2}w_{-1}w_0$
3	$t_{-2}w_0$	8	$w_{-1}w_0w_{+1}$	13	$w_{-2}t_{-1}w_0$	18	$t_{-2}w_{-1}$
4	$w_{-2}w_{-1}$	9	w_{-1}	14	$w_{-2}t_{-1}w_0$		
5	$w_{-2}t_{-1}$	10	$t_{-1}t_0$	15	$w_{-1}w_0$		

C. POS tagging experiments

We first run our experiment on ATR treebank by using n-gram models. The tagging performance is 77.6% far below the accuracy of 96.7% on Penn Treebank. The result shows that when evaluating on a large semantic-rich tagset, the POS tagging performance decreases significantly. In the second set of experiments, we just use default ME model and one type trigger to investigate contribution of the different trigger type. The results are showed in Figure 1. From the figure we can see that almost all triggers can improve the performance. The top 5 useful triggers are 1, 2, 7, 9 and 15. We should note that these triggers are all token-based triggers.

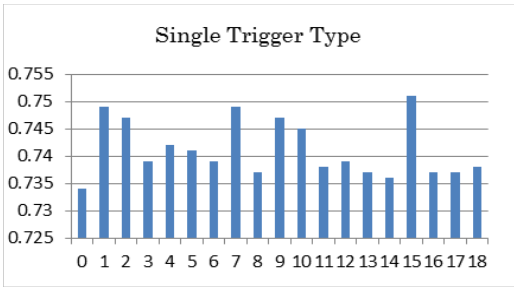

Figure 1: The leftmost column in figure shows the accuracy of default ME model's performance. The others show the accuracy by adding different trigger type.

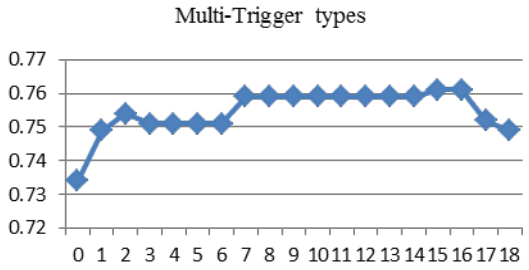

Figure2: The tagger's performance varies by adding each trigger type one by one.

In the third set of experiments, we add each trigger type one by one to the default ME model to investigate the impact of these trigger type. The experiment results are showed in Figure 2. The results show that different trigger types have different effect on performance improvement. Some trigger types even degrade the tagger's performance. From the Figure1 and Figure 2, we can see that most useful triggers are those with word information near the word that will be predicted. The tokens rather than POS tags have more effect on tagging accuracy improvement.

In our fourth set of experiments, we combine n-gram and ME method and investigate the combined system's performance with the combination of different trigger types. Table III shows the top 5 trigger types we selected in our experiments. As show in Table IV, by using all trigger types,

the tagging accurate is about 78.2 %. Using the top five useful trigger types, the tagging accurate is about 78.5 %. Note that these five trigger types are all of token-based triggers. For semantic-rich POS tagging task, we argue that tokens rather than POS tags contribute significantly to POS tagging accuracy improvement.

TABLE III. SELECTED LOCAL TRIGGER TYPES

No.	Feature templates
1	w_{+1}
2	$t_{-1}w_0$
7	w_0w_{+1}
9	w_{-1}
15	$w_{-1}w_0$

TABLE IV. TAGGING ACCURACY ON ATR TREEBANK

Tagger	Accuracy
HHM	77.6
ME	75.1
HMM+ME with no trigger types	78.1
HMM+ME with all trigger types	78.2
HMM+ME with selected trigger types	78.5

Conclusion

In this paper, we investigate the representations that can be applied to improve the performance of POS task on a large semantic and syntactic tagset. By exploring the proper representations for semantic-rich POS tagging, we can achieve a significant increase in tagging accuracy. The experiments demonstrate that the tokens rather than POS tags have more effect on tagging accuracy improvement. Our experiments also show that careful feature selection for a sematic-rich POS tagging system is very important.

Acknowledgement

The research described in this paper was partially supported by the Special Fund for Basic Scientific Research of Central Colleges, Chang'an University (CHD2010JC036, CHD2011JC075). Part of this work was done when the author was at ATR ITL.

References

[1] A.A.Kardan, M.B.Imani, "Improving Persian POS Tagging Using the Maximum Entropy Model", 2014 Iranian Conference on Intelligent Systems (ICIS2014), pp.1-5 ,Feb. 2014.

[2] M.Okhovvat, B.Bidgolib, "A hidden markov model for persian part-of-speech tagging", Procedia Computer Science, pp.977–981, 3, 2011.

[3] S.Kübler, M.Scheutz, E.Baucom, R.Israel,"Adding Context Information to Part Of Speech Tagging for Dialogues", Proc. of the Ninth International Workshop on Treebanks and Linguistic Theories, Vol. 9 , pp.115-126, 2010.

[4] N.Barrett, J.Weber-Jahnke J, "A token centric part-of-speech tagger for biomedical text", Artificial intelligence in medicine, 61(1):pp.11-20, May, 2014.

[5] X.Zhang,H.Huang,Z.Liang , "The application of CRF in Part of Speech Tagging" , IEEE-Nov 2009,9778-0-7695-3752-8.

[6] T.Large, E.Black, A.Finch, R.Zhang, "Applying Extrasentential Context To Maximum Entropy Based Tagging With A Large Semantic And Syntactic Tagset", Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora, pp.46-52,1999.

[7] K.Toutanova, C.Manning, "Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger", In Proc. of EMNLP/VLC-2000, pp. 63-70, 2000.

[8] T.Brants, "TnT-A statistical part-of-speech tagger", proc. of the 6th Applied Natural Language Processing Conference, Seattle, Washington, USA, 2000.

[9] A.Ratnaparkhi, "A maximum entropy model for part-of-speech tagging", proc. of EMNLP-96, Philadelphia, PA. 1996.