2012 IEEE International Conference on Systems, Man, and Cybernetics
October 14-17, 2012, COEX, Seoul, Korea

# Twitter Part-Of-Speech Tagging Using Pre-classification Hidden Markov Model

Shichang Sun[1,2]
[1] Dalian University of Technology
[2]Dalian Nationalities University
116024 Dalian, China
scsun@dlut.edu.cn

Hongbo Liu, Hongfei Lin
Dalian University of Technology
116024 Dalian, China
{lhb,hflin}@dlut.edu.cn

Ajith Abraham[3,4]
[3]VSB - Technical University of Ostrava
Ostrava, Poruba, Czech Republic
[4]Machine Intelligence Research Labs (MIR Labs)
Auburn, Washington 98071, USA
ajith.abraham@ieee.org

***Abstract*—Hidden Markov models (HMM) have been widely used in natural language processing (NLP), especially in syntactic level applications, which appears naturally as short-range-dependent sequence recognition problems. But the structure of HMM limits the usage of global knowledge including the sentiment analysis of the text, which has become an increasingly popular research topic in NLP now. In this paper, we propose a novel treatment of HMM model to use the result of sentimental subjectivity analysis in syntactic level task, i.e. part-of-speech (POS) tagging. The subjectivity information is introduced as a pre-classification procedure into the interval-type HMM. The subjectivity degree of the testing sentence is used as a combination factor to choose an appropriate value from the interval. Experiments results on public tagging data sets shows that the proposed approach enhanced the performance of POS tagging.**

***Index Terms*—Hidden Markov models, Subjectivity analysis, Part-of-speech tagging, Naive Bayes model**

## I. Introduction

POS tagging is considered as a fundamental part of natural language processing, which aims to computationally determine a POS tag for a token in text context. POS tagger is a useful preprocessing tool in many NLP applications such as information extraction and information retrieval [1], [2]. As social media becomes popular, Twitter POS tagging [3], [4] poses additional challenges on existing tagging models due to the conversational expression style and the free spelling style of the text. The sentimental information provides some global knowledge for the POS tagging task. Subjectivity analysis is a popular research topic in NLP, addressing the problem of judging if a text expresses an opinion about a given target. We find that the subjectivity classification can benefit the POS tagging task. Since, POS tagging is a sequential classification, the procedure of subjectivity analysis is called pre-classification.

HMM is a classical tool for POS tagging. HMM can efficiently classify complex and structured objects in sequence recognition problems. However, the structure of HMM limits the usage of global knowledge including the sentiment analysis of the text.

In this paper, we propose a novel treatment of HMM model to use the result of sentimental subjectivity analysis in syntactic level task, i.e. POS tagging. The subjectivity information is introduced as pre-classification procedure into the transition interval matrices of HMM. The subjectivity degree of the testing sentence is used as a combination factor to choose an appropriate value from the interval.

## II. Background and related work

Hidden Markov Models (HMM) have found many successful applications in Natural Language Processing (NLP) areas [5], [6], [7], especially in syntactic level applications, which appears naturally as short-range-dependent sequence recognition problems.

Gimpel [3] achieves good error reduction with the full feature set. Instead of designing an elaborate feature set, this paper makes an effort to use the result of high-level text analysis directly. High-level text analysis addresses the problems on semantics, topics, etc. Semantic information can used in text classification tasks [8], [9]. Recently, topic models have been increasingly used in syntactic application and sentiment analysis. Griffiths et al. [8] present a composite generative model integrating syntax and semantics. According to their experimental results, the function words and the content words are clustered separately. This clustering can be used in unsupervised POS tagging. Our work focuses on the typical supervised POS tagging.

Both supervised and unsupervised methods are used in HMM training. Self-adaptive design approach [10] focus on learning the correct states number with a maximum a prior procedure. It is reported that the classical expectation maximization (EM) method does not fit well in the POS tagging problem [11]. Besides EM method for HMMs training, several papers also use Evolutionary Optimization [12], [13] and Tabu Search [14] instead to achieve better global optima. However, few works focus on the partition of the training set. We use subjectivity information to partition the training set and propose Iterval Type HMM (ITHMM) to make use of the pre-classification.

The delicate structures make HMMs the most applicable Random Process tool for modeling time series data, however, the robustness of model are hindered by the lack of sufficient labeled training data. The potential performance of HMM is limited because it is traditionally based on predetermined model parameters. By extending the values of model parameters into intervals, there is room for HMM algorithms to perform better. The methods in [15], [16] need

978-1-4673-1714-6/12/$31.00 ©2012 IEEE

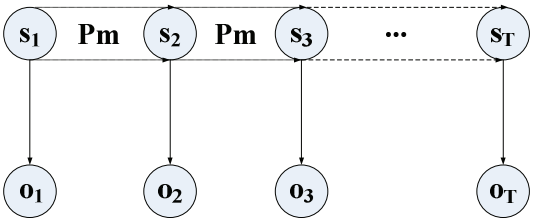


Fig. 1. HMM with transition interval matrix.

the representation of fuzzy relationship or fuzzy rules and are less cost-effective than the proposed work. Zeng [17] presents a fuzzy-set method to allow randomness in HMM and achieves robust performance on speech variation, but a Gaussian primary membership function has to be used for each state. Our approach is more specific to HMM structures and the intervals are calculated from training on two categories.

## III. Proposed approach

Typically one HMM corresponds to one finite state automaton with stochastic state transitions. By taking the pre-class of training set into consideration, the HMM transition matrix et al. are extended to transition interval matrix. So that the expression capacity of HMM can be expanded. The dependency graph of the proposed HMM approach is shown in Figure 1.

In Section III-A, we present the specification of the interval-type HMM. Afterward we discuss the usage of Naive Bayes model as the pre-classification procedure in Section III-B and provide an algorithm to search for the best state sequence in the set in Section III-C.

### A. Interval-Type Hidden Markov Model

HMM is a probabilistic model for modeling time series data. It extends the concept of Markov Random Process to include the case where the observation is a probabilistic function of the states. Its hidden states are not directly visible and each state can emit observable output symbols determined by its own probability distribution. This extension makes HMM applicable to many fields of interest such as Natural Languages Processing (NLP), where the amount of observable events, i.e. words, is often as big as hundreds of thousands.

POS tagging problem has been modeled with many machine learning techniques, which include Hidden Markov Models (HMM) [5], Maximum Entropy Models (MEMM) [18], Support Vector Machines (SVMs), Conditional Random Fields (CRF) [19], etc. Each model can have good performance after careful adjustment such as feature selection, but HMM have the advantages of small amount of data calculation and simplicity of modeling. In [20], HMM combined with good smoothing techniques and with handling of unknown words work better than other models. For such a sequence recognition problem, the classical EM algorithms and Viterbi algorithms for HMM can be found in [21], [22], [23], [24]. However, the delicacy of structures of HMM leaves the robustness of the model easily influenced by the bias of training set. The potential performance of HMM is limited because it is traditionally based on certain model parameters.

To integrate the pre-classification information into HMM, the Interval-Type Hidden Markov Model is presented. Such a HMM model can be viewed intuitively as learned from two pre-classified part of training data where each part has a specific bias on the transition and emission probabilities.

For clarity purposes, the specification of the model is presented based on the classical HMM [21] as follows:

- States
  $S = \{S_1, \cdots, S_N\}$ denotes the hidden state set, $N$ represents the number of these states. In POS tagging problem, $S$ stands for the part-of-speech tags. The part-of-speech tags carry structural significance although they are hidden in human language.
- Outputs
  $V = \{v_1, \cdots, v_M\}$ denotes the set of output symbols produced by states, $M$ represents the number of these symbols. In POS tagging problem, $V$ stands for vocabulary of the language and $M$ is the alphabet size.
- Transition interval matrix
  $\widetilde{A}$ denotes the state transition interval matrix as in Equ.(1).

$$\widetilde{A} = \{a_{ij}\} = \begin{bmatrix} (\underline{a}_{11}, \overline{a}_{11}) & (\underline{a}_{12}, \overline{a}_{12}) & \cdots & (\underline{a}_{1n}, \overline{a}_{1n}) \\ (\underline{a}_{21}, \overline{a}_{21}) & (\underline{a}_{22}, \overline{a}_{22}) & \cdots & (\underline{a}_{2n}, \overline{a}_{2n}) \\ \vdots & \vdots & \ddots & \vdots \\ (\underline{a}_{n1}, \overline{a}_{n1}) & (\underline{a}_{n2}, \overline{a}_{n2}) & \cdots & (\underline{a}_{nn}, \overline{a}_{nn}) \end{bmatrix} \quad (1)$$

  State transition $a_{ij} = P[q_{t+1} = S_j | q_t = S_i] \in \widetilde{A}$, $1 \le i, j \le N$. In POS tagging problem, $\widetilde{A}$ depicts the interval from which the statistical frequency value of the transitions between part-of-speech tags are chosen.
- Observation symbols interval matrix
  $\widetilde{B} = \{b_j(k)\} = \{(\underline{b_j(k)}, \overline{b_j(k)})\}$ denotes the observation symbols interval matrix, where $b_j(k) = P[V_k \ at \ t | q_t = S_j]$, $1 \le j \le N$, $1 \le k \le M$. In POS tagging problem, $\widetilde{B}$ depicts the interval of statistical frequency of words being categorized into some part-of-speech tags.
- Initial states interval matrix
  $\widetilde{\Pi} = \{\Pi_i\} = \{(\underline{\Pi_i}, \overline{\Pi_i})\}$ denotes the initial states interval matrix. It's a vector of initial states where $\Pi_i = P[q_1 = S_i]$, $1 \le i \le N$. Where $q_1$ is the state at initial time that satisfy two constraints $0 \le \Pi_i \le 1$, $1 \le i \le N$, and $\sum_{i=1}^{N}(\Pi_i) = 1$.
- Observation sequence
  Observation Sequence is a sequence of tokens to recognize. $O$ denotes the observation sequence, and $O = O_1 O_2 ... O_T$, and $T$ is the length of the sequence. In POS tagging problem, $O$ is the sentence in the target language.
- Combination factor
  In $(\widetilde{A}, \widetilde{B}, \widetilde{\Pi})$, the working value is chosen by combining the upper bound and the lower bound and the proportion of the two bounds is call combination factor $Pm$. $Pm$ measures how much the test sentence is similar with the two pre-classified categories. For POS tagging problem in this paper, the subjectivity degree of the test sentence

can be used.

- State sequence
  State Sequence Set is the result of sequence recognition through the proposed approach. In POS tagging problem, $Q = q_1 q_2 ... q_T$ stands for the labeled tags.

### B. Pre-classification procedure

To support subjectivity-aware POS tagging, we introduce sentiment analysis as a pre-classification procedure. Before used as training set, the POS corpus is classified into two categories, labeling "subjective" or "objective". This pre-classification is done by Naive Bayes (NB) model, which is typically used for sentiment analysis.

Subjectivity classification can benefit POS tagging. The genre information can also help POS tagging, but after experiment we found that genre information in brown corpus is not as beneficial for POS tagging as subjectivity information. Besides it's difficult to collect a collection of article for each kind of genres. We use two collections of subjectivity to build the upper and lower bound of $(\widetilde{A}, \widetilde{B}, \widetilde{\Pi})$. Besides, through the human labeling work on the corpus, we found that the subjectivity labeling is easier than POS tagging. So the proposed approach is cost-effective for the POS corpus to contribute more when labeled with "subjective" or "objective" tag.

Using the pre-classified data, the HMM model can be trained by supervised learning. In $(\widetilde{A}, \widetilde{B}, \widetilde{\Pi})$, the lower bound is calculated from training set with "objective" pre-class, and the upper bound is from the "subjective" pre-class.

The pre-classification model training algorithm is stated in Algorithm 1. Lines 1-6 prepares the twitter corpus. The initial subjectivity labels are crafted by hand. Then the POS twitter corpus can be expressed as $(SS_{obj}, SS_{sub})$. The corpus is cleaned by turning words into lower cases and replacing special names. Lines 7-12 generates a feature set using word features and trains the model with the training set. The NB parameters are typically trained according to Equs. (3), (4) and (5). In Lines 13-20, $(SS_{obj}, SS_{sub})$ is re-classifed into $(RSS_{obj}, RSS_{sub})$ using the NB model to get a pre-classified training set for ITHMM.

### C. Sequence recognition algorithm with pre-classification

ITHMM is designed to be trained with the re-classified corpus returned from Algorithm 1. The lower bound of $(\widetilde{A}, \widetilde{B}, \widetilde{\Pi})$ is trained upon $RSS_{obj}$ and the upper bound of $(\widetilde{A}, \widetilde{B}, \widetilde{\Pi})$ is trained upon $RSS_{sub}$. Note that the lower bound is not necessarily smaller than the upper bound.

We state the algorithm of sequence recognition with pre-classification in Algorithm 2. The sequence recognition algorithm is basically a Viterbi Algorithms. Equ. (6) is the iteration of the dynamic programming procedure using the criterion of the single best state sequence [21]. $\delta_t(j)$ is an array to save the maximum probability of state $j$ at time $t$ during previous states in the procedure. In line 03, $Psi$ is an array to keep track the previous state index that the maximum value of $\delta$ can be accumulated. Line 04-05 calculates the probability

**Algorithm 1** Pre-classification model training

**Input:** POS-tagging corpus for twitter
**Output:** Trained Naive Bayes model for pre-classification, Re-classified corpus to train Hidden Markov Model

01. **Prepare twitter corpus:**
02. Initial subjectivity labeling
03. by hand: $Category = 'obj', 'sub'$
04. $SS\_obj = [(Word, Tag)^T]^M$
05. $SS\_sub = [(Word, Tag)^T]^M$
06. Clean the corpus
07. **Model training:**
08. Feature generation:
09. $feature(SS\_obj, SS\_sub, word\_features)$
10. $Training\_set$ = $\{feature, true/false, obj/sub\}$
11. Evaluate the parameters of Naive Bayes model
12. $NB$ according to Equ.(3,4,5)
13. **Re-classification:**
14. $RSS\_obj = \{\}$
15. For each $sentence$ in $\bigcup\{SS_{obj}, SS_{sub}\}$:
16. $if P(y = 1|O) < 0.5$ :
17. $RSS_{obj} = RSS_{obj} \bigcup \{sentence\}$
18. $else$ :
19. $RSS_{sub} = RSS_{sub} \bigcup \{sentence\}$
20. **return:** $NB, RSS_{obj}, RSS_{sub}$

$P(y = 1|O)$ as the subjectivity degree according to Equ. (2) and $y = 1$ means the pre-class is "subjective". When NB is used to label a POS sentence without a subjectivity tag, the probability of "subjective" label given $O$ is called subjectivity degree and is used as $Pm$ according to Equ. (7). The HMM parameters are calculated using the intervals and combination factor $Pm$ in ITHMM in lines 7-9. The parameter $\lambda$ is introduced to deal with the sparseness problem of the training set.

$$P(y = 1|O) = \frac{P(y = 1)P(O|y = 1)}{P(O)} = \frac{P(y = 1)\prod_{i=1}^{n}(P(O_i|y = 1))}{\sum_{j=0}^{1}(P(y = j)\prod_{i=1}^{n}(P(O_i|y = j)))} \quad (2)$$

$$P(x_j = 1|y = 1) = \frac{\sum_{i=1}^{M} 1\{x_j^i = 1 \ and \ y^i = 1\}}{\sum_{i=1}^{M} 1\{y^i = 1\}} \quad (3)$$

$$P(x_j = 1|y = 0) = \frac{\sum_{i=1}^{M} 1\{x_j^i = 1 \ and \ y^i = 0\}}{\sum_{i=1}^{M} 1\{y^i = 0\}} \quad (4)$$

$$P(y = 1) = \frac{\sum_{i=1}^{M} 1\{y^i = 1\}}{M} \quad (5)$$

$$\delta_t(j) = \max_{1 \le i \le N}[\delta_{t-1}(i)a_{ij}]b_{jo_t}(O_{t+1}) \quad (6)$$

$$Pm = sigmoid(P(y = 1|O)) \quad (7)$$

**Algorithm 2** Sequence recognition under ITHMM

**Input:** A sequence of tokens $O$

**Require:** Transition interval matrix $\widetilde{A}$, Observation symbols interval matrix $\widetilde{B}$, Initial states interval matrix $\widetilde{\Pi}$, Trained Naive Bayes model $NB$ according to Algorithm (1)

**Output:** A sequence of tags $Q$

**Initialization:**
  $\delta_1(i) = \Pi_i b_i(o_1)$, $1 \leq i \leq N$
  $\psi_1(i) = 0$ // an array to store best states
  Calculate $Pm = P(y = 1|O)$ using $NB$
    according to Equ. (2)
**For** $2 \leq t \leq T$, $1 \leq j \leq N$:
  $a_{ij} = (Pm + \lambda) * \underline{a_{ij}} + (1 - Pm + \lambda) * \overline{a_{ij}}$
  $b_j(k) = (Pm + \lambda) * \underline{b_j(k)} + (1 - Pm + \lambda) * \overline{b_j(k)}$
  $\Pi_i = (Pm + \lambda) * \underline{\Pi_i} + (1 - Pm + \lambda) * \overline{\Pi_i}$
  update $\delta_t(j)$ according to Equ. (6)
  $\psi_t(j) = argmax_{1 \leq i \leq N}[\delta_{t-1}(i)a_{ij}]$
**Termination:**
  Set $q_t* = argmax_{1 \leq i \leq N}[\delta_T(i)]$
  $L = log(\max_{1 \leq i \leq N}[\delta_T(i)])$
  Store state sequence from $t = T - 1$
    to 1: $q_t* = \psi_{t+1}(q_{t+1}*)$
**return** $Q = q_1 q_2 ... q_T$

## IV. Experiment and Discussion

### A. Datasets

The data set we use is based on TWPOS [3]. TWPOS data is a part-of-speech corpus on 1500 twitter messages. The words are turned into lowercase. There are 25 tags in TWPOS, some of which are seldom used in common text tagging, i.e., "E" for "Emotion" such as ":)".

### B. Experimental Settings

In targeted POS tagging problem, we choose the NLTK [25] implementation of HMM as baseline. The Brown corpus can be imported into nltk. To have a fair comparison, both the developed method and the baseline use the same supervised training algorithm implemented in NLTK, which is implemented in hmm module. The performance metric used in this study is the accuracy of the prediction of token-tag pairs.

Since we need a corpus that have subjectivity label as well as POS tags, we tag each message as "subjective" or "objective" as the training set for Naive Bayes model. We only use words as features to train the model. The tags that are irrelevant to subjectivity analysis are removed. The tags we used is in Table I. We can see that in subjective twitters, there are more interjection and adjectives. We use the balanced corpus that includes 900 sentences, pre-classified into "objective" with 443 sentences and "subjective" with 457 sentences. Now we use Algorithm 2 and choose $\lambda$ as 1 to test the POS tagging accuracy.

### C. Results

We use Naive Bayes model to pre-classify the training set according to their subjectivity inclination. The informative degree of learned features is evaluated according to Equ. (8). The part-of-speech of the most informative words mainly include adjectives, verbs, interjections and punctuations. Although the performance of NB can influence the accuracy of POS tagging, we can use NB to re-classify the training set for HMM. We find that the training set re-classified by NB supports ITHMM tagging better than the human-classified training set.

$$Informative\_degree(O_i) = \frac{P(O_i|y =' subjectivity')}{P(O_i|y! =' subjectivity')} \quad (8)$$

In testing the performance of POS tagging, we use 10-fold cross validation on 80 test sentences. The results regarding the performance of the developed method are reported in Figure 2,

TABLE I
The Twitter part-of-speech tags that used in subjectivity analysis.

| Tag | Description | Example | % in subjective | % in objective |
|---|---|---|---|---|
| A | adjective | great | 6.5 | 4.5 |
| R | adverb | very | 5.2 | 4.5 |
| ! | interjection | lol | 4.6 | 1.7 |
| E | emotion | :-) | 1.2 | 0.8 |
| , | punctuation | !!! | 12.7 | 11.7 |
| G | abbreviation | ily | 0.6 | 1.1 |
| V | verbal | want | 14.9 | 15.0 |
| N | common noun | gift | 12.5 | 14.3 |

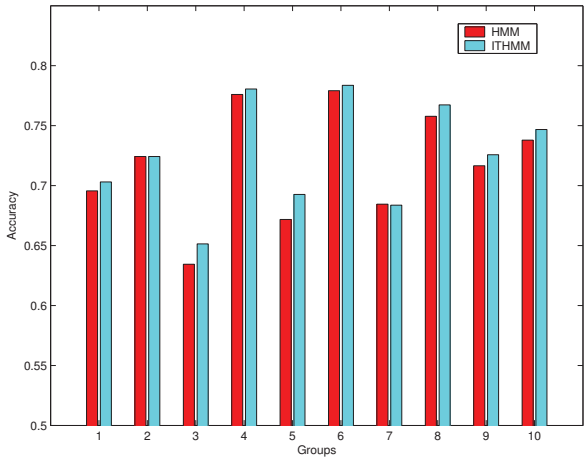


Fig. 2. Mean accuracy of POS tagging with HMM and ITHMM.

TABLE II
Accuracy details of POS tagging with HMM and ITHMM.

| Groups | HMM | ITHMM |
|---|---|---|
| 1 | 69.56% | 70.31% |
| 2 | 72.43% | 72.43% |
| 3 | 63.44% | 65.14% |
| 4 | 77.60% | 78.06% |
| 5 | 67.18% | 69.27% |
| 6 | 77.92% | 78.37% |
| 7 | 68.45% | 68.37% |
| 8 | 75.78% | 76.74% |
| 9 | 71.65% | 72.58% |
| 10 | 73.79% | 74.68% |
| Average | 71.78% | 72.59% |

TABLE III
ACCURACY DETAILS OF HMM AND ITHMM WHEN CORPUS SIZE INCREASES.

| Groups | Size | Algorithm | | | |
|---|---|---|---|---|---|
| | | HMM(obj) | ITHMM(obj) | HMM(sub) | ITHMM(sub) |
| 1 | 200 | 60.78% | 60.69% | 65.27% | 66.06% |
| | 300 | 62.37% | 63.12% | 67.02% | 67.28% |
| | 400 | 65.64% | 65.73% | 67.36% | 68.76% |
| | 500 | 67.32% | 67.23% | 68.50% | 70.07% |
| | 600 | 67.79% | 68.44% | 71.47% | 71.82% |
| | 700 | 68.81% | 69.47% | 71.73% | 72.16% |
| 2 | 200 | 54.20% | 55.30% | 70.03% | 70.18% |
| | 300 | 58.27% | 58.44% | 73.78% | 73.47% |
| | 400 | 60.31% | 61.07% | 74.39% | 74.39% |
| | 500 | 60.56% | 61.58% | 76.45% | 76.68% |
| | 600 | 62.17% | 62.51% | 77.14% | 77.98% |
| | 700 | 62.77% | 64.63% | 77.37% | 77.91% |
| 3 | 200 | 63.37% | 64.28% | 71.63% | 72.89% |
| | 300 | 63.37% | 64.28% | 71.63% | 72.89% |
| | 400 | 65.37% | 65.00% | 73.70% | 74.33% |
| | 500 | 66.00% | 67.45% | 74.15% | 75.76% |
| | 600 | 66.36% | 67.91% | 75.58% | 76.93% |
| | 700 | 67.27% | 68.99% | 77.74% | 78.82% |
| 4 | 200 | 66.34% | 66.52% | 74.31% | 74.13% |
| | 300 | 66.34% | 66.52% | 74.31% | 74.13% |
| | 400 | 66.34% | 66.52% | 74.31% | 74.13% |
| | 500 | 66.78% | 66.70% | 73.78% | 74.48% |
| | 600 | 67.40% | 67.57% | 74.31% | 75.26% |
| | 700 | 68.37% | 68.45% | 75.35% | 76.22% |

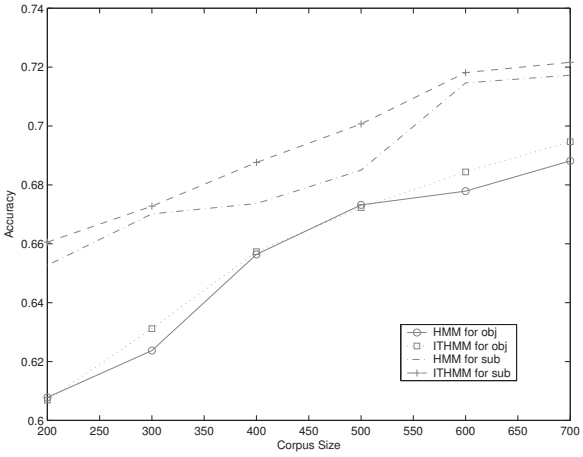


Fig. 3. Corpus size as a variable of mean accuracy for test data group 1.

comparing the mean accuracies of the developed method with that of HMM. Table II shows the details of accuracy. As illustrated, the proposed method always performs better than HMM.

Since there is no large-scale twitter POS corpus currently, the corpus size as a variable is investigated to show the performance as the corpus size increases. We test the performance of both of the algorithms according to corpus size. From the corpus 80 sentences of a single pre-classification is used to test in 8-fold cross validation. The tested corpus size ranges from 200 to 700. The result is reported in Figures 3, 4, 5, 6 and Table III. Both algorithms get better result when the corpus size increase and ITHMM tends to get better result when corpus is bigger.

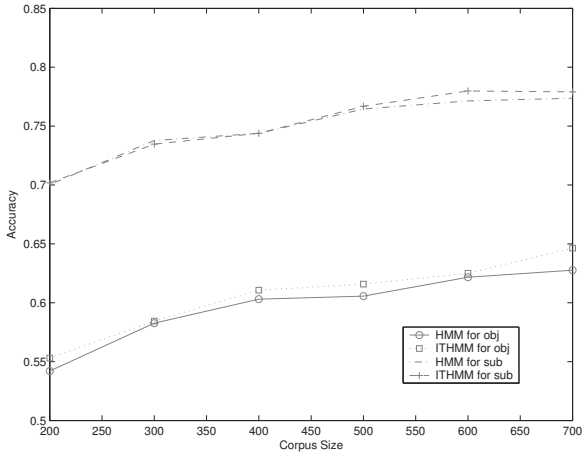


Fig. 4. Corpus size as a variable of mean accuracy for test data group 2.

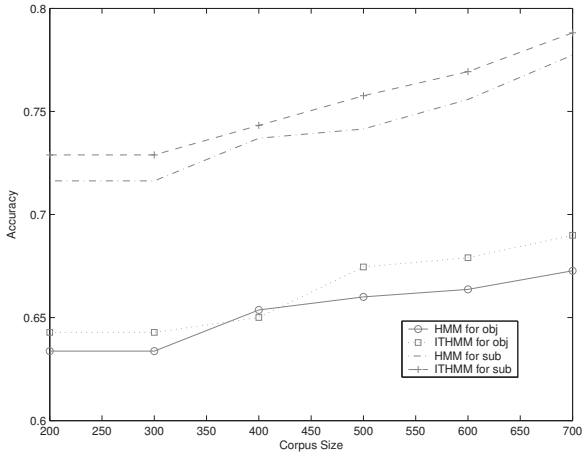


Fig. 5. Corpus size as a variable of mean accuracy for test data group 3.

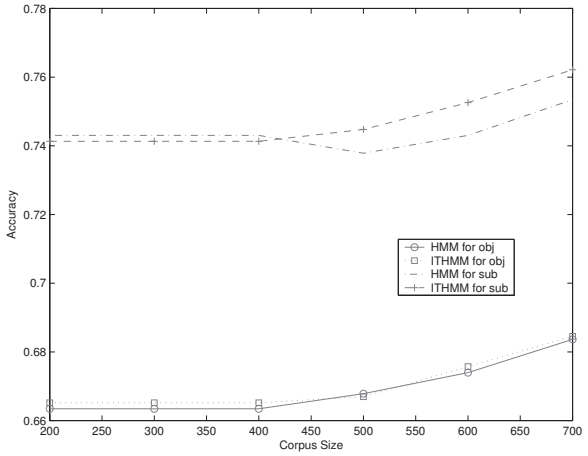


Fig. 6. Corpus size as a variable of mean accuracy for test data group 4.

### D. Discussion

Experimental results illustrate that the proposed method clearly outperforms the baseline. This improvement comes from the usage of global knowledge in our approach. The result validates that classifying the training set can benefit the sequence recognition task. In this paper, we use Naive Bayes (NB) model to pre-classify the training set. Only words features are used to train the model. But because we only use to NB to partition the training set into two groups that may yield two essentially different HMM parameters, the accuracy of NB classification is not vital in the pre-classification step. But the accuracy of subjectivity analysis model can influence the performance of tagging when used as a combination factor.

## V. Conclusion

The sentimental information provides some global knowledge for POS tagging task at semantic level. Our experiments illustrate that the subjectivity classification can benefit the POS tagging task. The proposed approach improves the POS tagging performance in an attempt to use existing or calibrated global information rather than to generate detailed features. Interval-type HMM allows un-determined model parameters to cope with global information, which is conveyed by the pre-classification procedure through training set. We also find that the training set re-classified by NB supports ITHMM tagging better than the human-classified training set. Experiments on public tagging data sets validate the applicability of the whole approach.

## Acknowledgments

The first author would like to thank Liyong Zhang of Dalian University of Technology for his scientific collaboration in this research work. Ajith Abraham is supported by the SGS in VSB - Technical University of Ostrava, Czech Republic, Under the Grant No. SP2012/58, and this research was supported by the European Regional Development Fund in the IT4Innovations Centre of Excellence project (CZ.1.05/1.1.00/02.0070) and by the Bio-Inspired Methods: Research, Development And Knowledge Transfer Project, Reg. No. CZ.1.07/2.3.00/20.0073 funded by Operational Programme Education for Competitiveness, Co-financed by ESF and State Budget of the Czech Republic.

## References

[1] M. Chang, D. Goldwasser, D. Roth, and V. Srikumar, "Structured output learning with indirect supervision," in *Proceedings of the International Conference on Machine Learning*. IEEE Computer Society, 2010, pp. 199–206.

[2] F. Salvetti, S. Lewis, and C. Reichenbach, "Impact of lexical filtering on overall opinion polarity identification," in *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications, Stanford, US*, 2004.

[3] K. Gimpel, N. Schneider, B. O. Connor, D. Das, D. Mills, J. Eisenstein, M. Heilman, D. Yogatama, J. Flanigan, and N. A. Smith, "Part-of-Speech Tagging for Twitter : Annotation , Features , and Experiments," in *Proceedings of the Annual Meeting of the Association for Computational Linguistics, companion volume*, 2011.

[4] F. Ji, Z. Liu, X. Qiu, and X. Huang, "Part-of-speech tagging for micro blog via 2d sequence labeling," *Journal of Computational Information Systems*, vol. 8, no. 3, pp. 1149 – 1156, 2012.

[5] J. Kim, H. Rim, and J. Tsujii, "Self-organizing markov models and their application to part-of-speech tagging," in *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics-Volume 1*. Association for Computational Linguistics, 2003, pp. 296–302.

[6] G. Zhou and J. Su, "Error-driven hmm-based chunk tagger with context-dependent lexicon," in *Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2000, pp. 71–79.

[7] D. Freitag and A. McCallum, "Information extraction with hmm structures learned by stochastic optimization," in *Proceedings of the National Conference on Artificial Intelligence*. Menlo Park, CA, 2000, pp. 584–589.

[8] T. L. Griffiths, M. Steyvers, D. M. Blei, and J. B. Tenenbaum, "Integrating topics and syntax," in *In Advances in Neural Information Processing Systems 17*. MIT Press, 2005, pp. 537–544.

[9] J. Meng, H. Lin, and Y. Yu, "A two-stage feature selection method for text categorization," *Computers and Mathematics with Applications*, vol. 62, no. 7, pp. 2793–2800, 2011.

[10] J. Li, J. Wang, Y. Zhao, and Z. Yang, "Self-adaptive design of hidden markov models," *Pattern Recognition Letters*, vol. 25, no. 2, pp. 197–210, 2004.

[11] M. Johnson, "Why doesnt em find good hmm pos-taggers," in *In EMNLP*, 2007, pp. 296–305.

[12] N. Najkar, F. Razzazi, and H. Sameti, "A novel approach to hmm-based speech recognition systems using particle swarm optimization," *Mathematical and Computer Modelling*, vol. 52, no. 11-12, pp. 1910–1920, 2010.

[13] J. Meng, S. Xu, X. Wang, Y. Yi, and H. Liu, "Swarm-based dhmm training and application in time sequences classification," *Journal of Computational Information Systems*, vol. 1, pp. 197–203, 2010.

[14] T. Chen, X. Mei, J. Pan, and S. Sun, "Optimization of hmm by the tabu search algorithm," *Journal of Information science and engineering*, vol. 20, no. 5, pp. 949–957, 2004.

[15] Y. Cheng and S. Li, "Fuzzy time series forecasting with a probabilistic smoothing hidden markov model," *Fuzzy Systems, IEEE Transactions on*, vol. 20, no. 2, pp. 291–304, 2012.

[16] M. Rafiul Hassan, B. Nath, M. Kirley, and J. Kamruzzaman, "A hybrid of multiobjective Evolutionary Algorithm and HMM-Fuzzy model for time series prediction," *Neurocomputing*, vol. 81, pp. 1–11, Apr. 2012.

[17] J. Zeng and Z. Liu, "Type-2 fuzzy hidden markov models and their application to speech recognition," *IEEE Transactions on Fuzzy Systems*, vol. 14, no. 3, pp. 454–467, 2006.

[18] A. McCallum, D. Freitag, and F. Pereira, "Maximum entropy markov models for information extraction and segmentation," in *Proceedings of the Seventeenth International Conference on Machine Learning*. IEEE, 2000, pp. 591–598.

[19] J. Lafferty, A. McCallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *Proceedings of the Eightteenth International Conference on Machine Learning*. IEEE, 2001, pp. 282–289.

[20] T. Brants, "Tnt: a statistical part-of-speech tagger," in *Proceedings of the Sixth Conference on Applied Natural Language Processing*. Association for Computational Linguistics, 2000, pp. 224–231.

[21] L. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[22] B. Juang and L. Rabiner, *Fundamentals of speech recognition*. Prentic-Hall, Englewood Cliffs, NJ, 1993.

[23] L. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state markov chains," *The Annals of Mathematical Statistics*, vol. 37, no. 6, pp. 1554–1563, 1966.

[24] L. Baum and J. Eagon, "An inequality with applications to statistical estimation for probabilistic functions of markov processes and to a model for ecology," *Bulletin of American Mathematical Society*, vol. 73, no. 3, pp. 360–363, 1967.

[25] E. Loper and S. Bird, "NLTK: The Natural Language Toolkit," in *Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*. Somerset, NJ: Association for Computational Linguistics, 2002, pp. 62–69.