Comparison of Six POS Tagging Methods on 12K Sentences Khmer Language POS Tagged Corpus

Ye Kyaw Thu[†*] **Vichet Chea**[‡] **Yoshinori Sagisaka**[*]

[†]Artificial Intelligence Lab., Okayama Prefectural University (OPU), Okayama Prefecture, Japan
[‡]National Institute of Posts, Telecommunications and ICT (NIPTICT), Phnom Penh, Cambodia
[*]Language and Speech Science Research Lab., Waseda University, Tokyo, Japan
ye@c.oka-pu.ac.jp, vichet.chea@niptict.edu.kh, ysagisaka@gmail.com

Abstract

A robust Khmer Part-of-Speech (POS) tagger is necessary for Khmer natural language processing (NLP) research and not available publicly yet. From this reason, we developed manually annotated twelve thousand sentences POS tagged corpus for a general domain. We also evaluated six POS tagging approaches: Hidden Markov Model (HMM), Maximum Entropy (ME), Support Vector Machine (SVM), Conditional Random Fields (CRF), Ripple Down Rules-based (RDR) and Two Hours of Annotation Approach (i.e. combination of HMM and Maximum Entropy Markov Model) on our developed POS tagged corpus. The POS tagging experimental results were measured with accuracy and in terms of confusion pairs. The result shows that RDR and HMM approach is the best performance for open test-set. The SVM approaches achieved highest performance on closed test-set and also comparable result with CRF on open test-set. We plan to release our developed POS tagged corpus and trained models in December 2017.

Keywords: Part-of-Speech Tagging, HMM, Maximum Entropy, SVM, CRF, RDR, Khmer Language

1 Introduction

Part-of-Speech tagging is a fundamental process for various natural language processing (NLP) tasks such as spelling checking, grammar induction, parsing, word sense disambiguation, information retrieval. Although several POS tagging works proposed for Khmer language, currently there is no publicly available trainable size of POS tagged corpus and also a POS tagger yet. Moreover there is no experimental result with well-known POS tagging methods such as HMM, SVM and CRF. In this paper we propose our defined twenty four POS tag-sets for Khmer language and present experimental results of well-known and recent POS tagging approaches on manually annotated twelve thousand POS tagged corpus. We used automatic evaluation in the form of accuracy and also manually evaluated on errors of each POS tag in terms of confusion matrix.

2 Related Work

In this section, we summarize some previous works relating to Khmer POS tagging. NOU et al. (2007) proposed Khmer POS tagger based on transformation based approach [1], [2]. The 27 POS tags were defined and including small classes such as time expression, possibility expression, action expression classes etc. They derived 32,000 words from the Khmer Dictionary published by Royal Academy of Cambodia and manually tagged with their defined tag-set. The semi-automatic annotation was done on 1,102 Khmer sentences (37,452 words) that collected from the Kohsantepheap newspaper for training and testing. The architecture of the POS tagger contained two main phases. The initial phase handle some pre-processing works such as sentence splitting, word segmentation and assigning default tags (i.e. the most frequent tag calculated from the training corpus). The transformation phase update the some default tags to more suitable tags based on contextual rules learned from the annotated corpus. The result shows that the transformation process can reduce 2.72% of tagging error on the trained data and achieved 95.12% accuracy on test data

(5,364 words in total and contained 2,577 ambiguous words). The same authors applied hybrid approach that combined rule-based and tri-gram model for Khmer unknown word POS guessing [3]. They used the same POS tag-set, wordlist and the corpus that they defined for their previous work, the transformation based approach [1]. Although hybrid approach achieves some encouraging results, the POS tagging accuracies are only 88.9% and 78.2% on training and test data respectively.

The PAN (Pan Asia Networking) localization project for Khmer language developed POS tagger based on decision tree approach [4], [5]. The well known POS and lemma information tagger named TreeTagger [6] was used for semi-automatic tagging [7], [8] with defined 21 POS tag-set. The TreeTagger is freely available for research purpose and it has been successfully used to tag several languages such as German, English, French, Italian, Danish, etc. In this tagging method, transition probabilities are estimated using a binary decision tree to avoid the problems of Markov Model based approach with sparse data. The developing process for Khmer language POS tagger was started with manually annotated 20,000 words to get an initial parameter file (used train-tree-tagger program). The initial parameter file was used to tag next 10,000 untagged words with TreeTagger and tagging errors of output POS tagged file was manually corrected for next incremental training process. The experimental result on 5,286 untagged words with the final parameter file that trained with 73,206 words was achieved 98.80%. To the best of our knowledge, only POS tagged corpus (3,998 sentences) was released [9] with Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0) license as the output of the PAN localization project for Khmer and the POS tagger is not publicly available yet.

The NOVA POS tagging system was proposed for Khmer language in 2016 [10]. From the manual of detailed guidelines for the surface annotation of the Khmer texts in Asian language treebank (ALT) [11], [12], we learned that it provides four basic tags: "n", "v", "a", and "o" to represent fundamental word classes, with further three auxiliary tags to represent numbers, punctuations marks, and tokens with weak syntactic roles. In details, there are seven POS tags in total and they are "n" (general nouns, can be subjects or objects of tokens tagged by v), "v" (general verbs, can take tokens tagged by n as arguments), "a" (general adjectives, can directly describe or modify tokens tagged by n), "o" (other modifications or complements for tokens or larger syntactic parts), "1" (general numbers), "." (general punctuation marks) and "+" (a catch-all category, for tokens with weak syntactic roles). Besides the simple tags, a pair of brackets "[" and "]" are further applied to show multiple tags "working (together) as". The brackets are used widely in the annotation to represent various linguistic phenomena, mainly for compounds in the case of Khmer. The 1,393 sentences of NOVA POS tagged sample data is publicly available with Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International (CC BY-NC-SA 4.0) lincense [13].

3 Propose POS Tag-set

Part of speech is a category to which a word is assigned in accordance with its syntactic functions. In Khmer grammatical system, many linguists has defined their own part of speech according to their trend of research. Even though, many books are published, there are no standard agreement yet especially on number and name of POS tags. Comparing to English language, some English POS are not used in Khmer language, such as gerund, comparative and superlative adjectives, particle, etc. Based on CHOUN NATH dictionary, Khmer POS Tag set is defined. Some new POS tags that are not defined in the dictionary are added for considering word disambiguation task. Unlike English grammar, some Khmer sentences consist of more than one verb. In Khmer sentences, words are written continuously without using space and thus, preprocess of automatic word segmentation results will strongly effect the POS tagging. In this paper, we used our previous research outcome of word segmentation [14] and some manual corrections. The definitions and descriptions of POS tags are presented in detail as follow:

1. Abbreviation (AB): For example, គម or គ.ម for kilometer (km), អសប for United Nation (UN), ពស or ព.ស for ពុទ្ធសករាជ (Buddhism era), នប or ន.ប for នគរបាល (police), អហ or អ.ហ for អាវុធហត្ថ (Police Military) etc.

2. Adjective (JJ): Adjective is a word used to modify or describe the noun. Adjective is usually at the right hand side of noun. There are very few adjectives that their positions are before noun. ក្រហម (red), កន្លះ (half), ប្លែក (strange), តូច (small), ល្អ (good), ស្អាត (beautiful) etc.

3. Adverb (RB): An adverb is a word that is used to modify verb, adjective or another adverb. For example, ណាស់ (very), ពុំ (not), ទើប (just), ពេកក្រៃ (very), ហើយ (already) etc.

4. Auxiliary Verb (AUX): Only three groups of verbs are tagged as auxiliary verb that used to make tense.

   - Past form: បាន or មាន + Verb
   - Progressive form: កំពុង + Verb
   - Future form: នឹង + Verb

5. Cardinal Number (CD): A cardinal number is a word or a number that denoting the quality. For example, បី (three), ១០០ (100), ចតុ (four), ពាន់ (thousand), លាន (million) etc.

6. Conjunction (CC): Conjunction is a word to connect between words, phrases, and sentences. ក៏ប៉ុន្តែ (but), ពីព្រោះ (because), ដ្បិត (for, since), ទម្រាំតែ (until), ពុំនោះសោត (otherwise), បើ (if) etc.

7. Currency (CUR): CUR for currency symbol such as: ៛, $, £, € etc.

8. Determiner Pronoun (DT): In Khmer grammar, determiners are classified under pronoun unlike English. It is used to tell location or/and uncertainty of noun. They are equivalent to English words: this, that, those, these, all, every, each, some etc. For example, នេះ (this), នោះ (that), ទាំងនេះ (these), ទាំងអស់ (all), នានា (various), ខ្លះ (some), សព្វ (every) etc.

9. Double Sign (DBL): Double sign (ៗ) is used to remind reader to read the previous word twice. For example, មនុស្ស/NN (people) គ្រប់/DT (every) ៗ/DBL គ្នា/PRO (person), “everybody” in English.

10. Et Cetera (ETC): ។ល។ is equal to et cetera (etc.) in English.

11. Full Stop (KAN): There are two full stops in Khmer language, ។ for sentence and ៕ for paragraph.

12. Interjection (UH): Word represents sound of animal, machine, and surprised sound. Interjections are always at the beginning of a sentence, and mostly followed by exclamation mark. For example, អូ (Oh!), ម៉េវ (Meow), អ៊ុះ (uh) etc.

13. Measure Word (M): Measure Words are classified to describe different quality corresponding class of noun. Some of these words can not be found in English. For example: ព្រះសង្ឃ/NN (monk) ២/CD (2) អង្គ/M (person), សំលៀកបំពាក់/NN (cloth) ១/CD (1), សម្រាប់/M (set), ឆ្កែ/NN (dog) ១/CD (1) ក្បាល/M (head) etc.

14. Noun (NN): A noun is a word or compound word that identifies a person, an animal, an object, an idea, a thing, etc. For example: ឡាន (Car), ការអភិវឌ្ឍន៍ (Development), សកម្មភាព (Action), ខ្មៅដៃ (Pencil), ទឹកកក (Ice) etc.

15. Particle (PA): We consider three types of particle and they are hesitation, response and final. For the two medial particle words ក៏ (“so, then, but” in English) and នូវ (“of, with” in English) [15], we consider them as RB and IN.

- Hesitation Particle: ខ្ញុំ (I) គិត (think) ...អើ/PA (Er. . .) មិន (not) ឃើញ (see), (“I er... don’t think so” in English)
- Response Particle: អើ/PA (Hm, Ah) ខ្ញុំ (I) ដឹង (know) ហើយ (already), (“Hmm I already know” in English)
- Final Particle: There are some final particles such as ណា៎, សិន and បុះ. Example usage of ណា៎: កុំ/RB (don’t) ភ្លេច/VB (forget) ណា៎/PA, (“Hmm don’t forget!” in English), Example usage of សិន: ចាំ/VB (wait) បន្តិច/RB (a while) សិន/PA, Example usage of បុះ: ទៅ/VB (go) បុះ/PA

16. Preposition (IN): Preposition is a word or a compound word that is used to connect two different words or phrases. It indicate the place, time, possession, relation etc. For example, ចំពោះ (to), ដល់ (to), ដើម្បី (in order to), ក្នុង (in), លើ (on), រវាង (between, around) etc.

17. Pronoun (PRO): A pronoun is a word that substitutes of a noun or a noun phrase. Those words are equivalent to Englis word: I, he, she, it, we, they, them, him, her etc. For example, ខ្ញុំ (I), គាត់ (he or she), យើង (we), ពួកយើង (our group or we), ខ្ញុំបាទ (polite form of I, me), ទូលបង្គំ (I, me for conversation with royal family) etc.

18. Proper Noun (PN): A proper noun is a noun that represents of a unique thing, for example, name of person, name of place and name of date etc. For example: សុខា (Sokha) ភ្នំពេញ (Phnom Penh), ថ្ងៃអង្គារ (Tuesday), កាល់តិច (Caltex), មេគង្គ (Mekong) etc.

19. Question Word (QT): In Khmer language, តើ is mostly used in the beginning of an interrogative sentence. For example, តើ/QT អ្នក/PRO (you) ឈ្មោះ/NN (name) អ្វី/PRO (what)?, “What is your name?” in English.

20. Relative Pronoun (RPN): In Khmer language, there is only one relative pronoun. It is ដែល “that, which, where, who” in English.

21. Symbol (SYM): SYM for others sign or symbol such as: +, −, *, , , =, @, #, % etc.

22. VB_JJ: VB_JJ is a tag for an adjective which its original form is a Verb. Currently, there is no proposed POS tag name for such kind of Khmer words. Although we can use JJ tag, we want to clarify by using VB_JJ POS tag for its function and also for semantic purpose. For example:

    (a) The word សម្រាប់ (for) or ដើម្បី (to) is normally removed in both written and spoken Khmer.
    កន្លែង/NN (place) សម្រាប់ (for) ធ្វើការ/VB_JJ (working), office in English
    ម៉ាស៊ីន/NN (Machine) សម្រាប់ (for) បោក/VB_JJ (washing) ខោអាវ/NN (cloth), washing machine in English
    ពួកគាត់/PRO (they) អាច/VB (can) មាន/VB (have) ការងារ/NN (work) ធ្វើ/VB_JJ (to do)

    (b) When Khmer Relative Pronoun is removed, the verb form keep the same as it was. It must be VB_JJ it is no longer a Verb in subbordiante clause.

    សិស្ស (student) ដែល (who) មាន/VB (has) ពិន្ទុ (mark) ខ្ពស់ (hight) នឹង (will) ទទួលបាន (get) អាហារូបករណ៍ (scholarship), student who has hight mark will get a scholarship in English but when ដែល who is removed, មាន/VB (has) should become មាន/VB_JJ (having)

23. Verb (VB): Verb is a word that shows the action, even, and condition. Verb

is a middle part of phrase. Normally, verb always need object and sometime it also need complement. For example, ស្ដាប់ (listen), មានប្រសាសន៍ (say), ស្រលាញ់ (love), ច្រៀង (sing), បើកបរ (drive) etc.

24. Verb Complement (VCOM): Its original form is a verb, but it will turn into VCOM when two verbs in a sentence to emphasize the first verb. Especially, a compound verb is splitted by the word មិន (no or not), the first part is a verb and the second part is VCOM. For example, លក់ (sell) ដាច់/VCOM (a lot), ប្រលង (exam) មិន (no) ជាប់/VCOM (pass), ដេក/VB (sleep), មិន/RB (not) លក់/VCOM (sleep well) etc.

4 POS Tagging Methodologies

In this section, we describe the POS tagging methodologies used in the experiments in this paper.

4.1 Conditional Random Fields (CRFs)

Linear-chain conditional random Fields (CRFs) [16] are models that consider dependencies among the predicted segmentation labels that are inherent in the state transitions of finite state sequence models and can incorporate domain knowledge effectively into segmentation. Unlike heuristic methods, they are principled probabilistic finite state models on which exact inference over sequences can be efficiently performed. The model computes the following probability of a label sequence $\mathbf{Y} = \{y_1, ..., y_T\}$ of a particular character string $\mathbf{W} = \{w_1, ..., w_T\}$.

$$P_{\boldsymbol{\lambda}}(\mathbf{Y}|\mathbf{W}) = \frac{1}{Z(\mathbf{W})} exp(\sum_{t=1}^{T} \sum_{k=1}^{|\boldsymbol{\lambda}|} \lambda_k f_k(y_{t-1}, \mathbf{W}, t)) \quad (1)$$

where $Z(\mathbf{W})$ is a normalization term, f_k is a feature function, and $\boldsymbol{\lambda}$ is a feature weight vector.

4.2 Hidden Markov Model (HMM)

The Hidden Markov Model (HMM) is a probabilistic sequence model: given a sequence of units (words, letters, morphemes, sentences, whatever), they compute a probability distribution over possible sequences of labels and choose the best label sequence [17], [18]. In an HMM for POS tagging, observation is a sequence of words $\mathbf{o} = x_1, \ldots, x_n$ and that is associated with a state sequence of POS tags that we cannot observe $\mathbf{s} = y_1, \ldots, y_n$. The model describes the joint state and observation sequence:

$$p(y_1, \ldots, y_n, x_1, \ldots, x_n) = \\ p(y_1)p(x_1|y_1) \prod_{i=2}^{n} p(y_i|y_{i-1})p(x_i|y_i) \quad (2)$$

and the probability of the observation sequence can be obtained by marginalizing:

$$p(x_1, \ldots, x_n) = \sum_{\mathbf{y}} p(x_1, \ldots, x_n, y_1, \ldots, y_n) \quad (3)$$

Here, the key assumptions of HMM are:

- The state sequence $p(y_i|y_1, \ldots, y_{i-1}) = p(y_i|y_{i-1})$ is Markov
- The observations are conditionally independent of next and previous states and observations given the current state:

$$p(x_i|x_1, \ldots, x_n, x_1, \ldots, \\ x_{i-1}, x_{i+1}, \ldots, x_n) = p(x_i|y_i) \quad (4)$$

Graphical representation of a HMM can be seen in Fig 1.

4.3 Maximum Entropy (MaxEnt)

The original concept of Maximum Entropy (MaxEnt) comes from Physics [19]. MaxEnt models have been used in various tasks of natural language processing including POS tagging [20], [21], [22], [23]. The principle of ME estimates probability distribution based on the minimal bias (i.e. maximal entropy) while verifying the statistical properties measured on the observation set. Those properties are referred to as the constraints that derived from the training data. The exponential form of MaxEnt model for POS tagging can be stated as Equation (5):

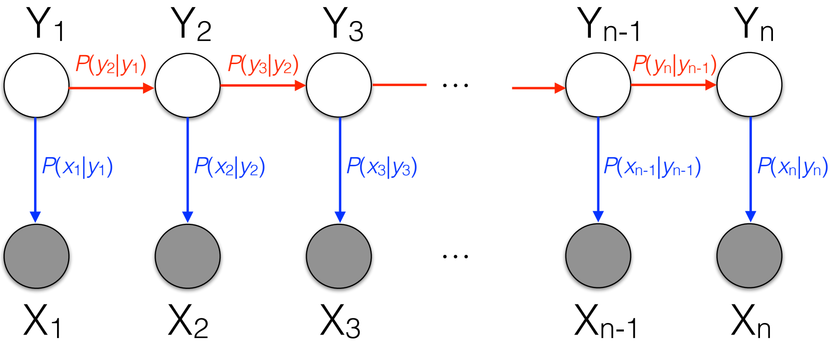

Figure 1. Graphical represenation of a Hidden Markov Model. Here, state variables $Y_1, Y_2, \ldots, Y_n$ form a Markov chain but this sequence of variables is not observed (i.e. hidden). The $X_1, X_2, \ldots, X_n$ are observable variables (i.e. output) of the Markov chain. Horizontal and vertical arrows indicate conditional dependence relations of variables.

$$P(t|h) = \frac{1}{Z(h)} \exp\left(\sum_{i=1}^{n} \lambda_i f_i(h,t)\right) \tag{5}$$

where, t is the POS tag, h is the history/context, $f_i(h,t)$ is a feature/class with associated feature-weight parameter λ_i and normalization function $Z(h)$. POS tagging problem can be formally stated as Equation (6):

$$P(t_1, \ldots, t_n, |w_1, \ldots, w_n) = \prod_{i=1}^{n} P(t_i|h_i) \tag{6}$$

where, given a sequence of words $w_1, \ldots, w_n$ and finding the conditional probability of a tag sequence $t_1, \ldots, t_n$. In MaxEnt modeling, the features are binary or multiple valued functions, which associate a POS tag with various elements of the context. For example:

$$f_i(h,t) = \begin{cases} 0, & \text{if } word(h) = \text{Phnom Penh} \quad \& \quad t = \text{PN} \\ 1, & \qquad\qquad\qquad\qquad\qquad \text{otherwise} \end{cases} \tag{7}$$

4.4 Ripple Down Rules-based (RDR)

Ripple-Down Rules (RDR) is an approach to building knowledge-based systems (KBS) incrementally, while the KBS is in routine use [24]. [25], [26] present a new error-driven approach to automatically restructure transformation rules in the form of a Single Classification Ripple Down Rules (SCRDR) tree [24] [27]. A SCRDR can be notated as a triple $< rule, X, N >$, where X and N are the exception RDR and the succeding RDR (i.e. if-not rules) respectively (see Figure 2) [28]. Cases in SCRDR are evaluated by passing a case to the root (Rule 0 in Figure 2). At any node in SCRDR tree (i.e. Rule 1 to Rule 6), if the condition of a node n met, the case is passed on to the exception child of n using the except link if it exists. Otherwise, the case is passed on to the if-not child of n. In SCRDR approach a conclusion is always given by the last node in the process. To ensure that a conclusion is always given, the root node (also known as default node) is usually setup with the condition which is always satisfied.

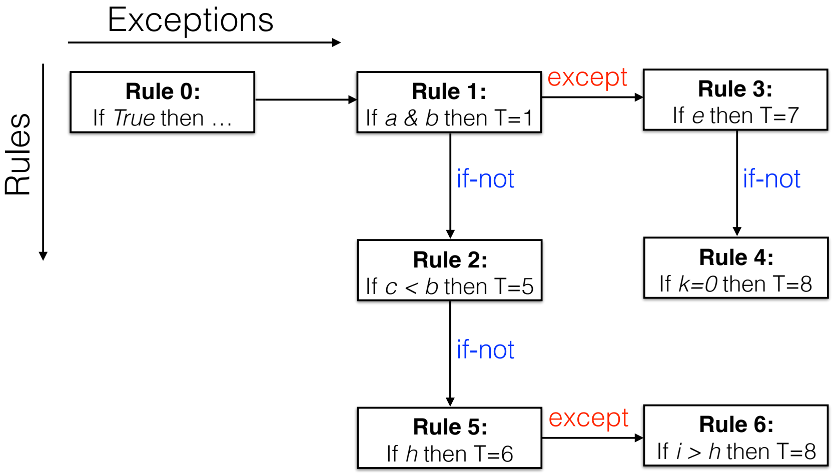

Figure 2. A binary tree of Single Classification Ripple Down Rules

4.5 Support Vector Machine (SVM) based Point-wise Classification

The SVM is a supervised machine learning algorithm for binary classification and considered for non-separable training data [29]. Although SVM methods can handle data consisting of two categories, pair-wise method was applied for multi-class classification or POS tagging problem [30], [31], [32]. The combination of two separate techniques to achieve more efficient corpus annotation: point-wise estimation and word-based annotation was proposed in 2010 [33]. Point-wise estimation assumes that every decision about a segmentation point or word pronunciation is independent from the other decisions [33]. From this concept, a single annotation model can be trained on single annotated words, even if the surrounding words are not annotated. We selected SVM based point-wise classification as one of the POS tagging methods for this paper.

4.6 Two Hours of Annotation Approach

Semi-supervised learning of POS tagging from two hours manually annotation data was proposed for low-resource languages [34]. Proposed system has four main steps, (1) Tag dictionary expansion, (2) Weighted model minimization, (3) Expectation maximization (EM) HMM training and (4) Maximum Entropy Markov Model (MEMM) training. The label propagation, the Modified Adsorption (MAD) algorithm [35] is used for tag dictionary expansion, step 1. It is a graph-based technique for spreading labels between related items and their graphs are seeded with POS-tag labels from the two hours human-annotated data. After the step 1, all words have information but still noisy and thus induce a cleaner hard tagging [36], [37] from a noisy soft tagging in step 2 model minimization. The step 3 uses the EM algorithm initialized with the noisy labels and constrained with the expanded tag dictionary to produce an HMM. However, the HMM produced by stage 3 will contain zero probabilities for out-of-vocabulary (OOV) words of test-corpus and thus it cannot used directly for POS tagging. It is used to provide a Viterbi labeling of the raw corpus, following the auto-supervision [37]. Output of step 3 can be concatenated with the token-supervised corpus if it is available, and used to train a MEMM POS tagger.

5 Experimental Setup

5.1 Corpus Developing

We collected 12,000 sentences (25,626 words in total and 7,623 unique words) mainly from various Khmer websites and including students list, and voter list of national election committee of Cambodia. Initial word segmentation was done with our developed Khmer word segmenter based on Conditional Random Fields (CRF) [14]. POS tagging with defined POS tags for each words

Table 1. Statistic of Part-of-Speech Tag-set in 12K corpus.

No.	POS-tag	Frequency	Proportion
1	AB	69	0.0534822%
2	AUX	2466	1.91141%
3	CC	2788	2.16099%
4	CD	3337	2.58652%
5	CUR	0	0.00000%
6	DBL	392	0.303841%
7	DT	4311	3.34147%
8	ETC	7	0.00542573%
9	IN	13444	10.4205%
10	JJ	4446	3.44611%
11	KAN	3028	2.34701%
12	M	343	0.265861%
13	NN	31946	24.7615%
14	PA	829	0.642561%
15	PN	20084	15.5672%
16	PRO	12194	9.45161%
17	QT	79	0.0612332%
18	RB	6428	4.98237%
19	RPN	1756	1.36108%
20	SYM	2412	1.86955%
21	UH	56	0.0434058%
22	VB	16519	12.8039%
23	VB_JJ	1721	1.33395%
24	VCOM	364	0.282138%

based on the context and error checking were done manually by three Khmer natives. We used the POS tag-sets that we mentioned in Section 3. The average number of words per sentence in the whole corpus is 10.75. Here, some symbols such as "។" (Khmer sign Khan), "៖"(Khmer sign Camnuc pii kuuh), "–", "?", "[", "]" etc. also counted as words. The shortest sentence in the corpus contained only 1 word and the longest sentence of the current corpus contained 169 words. Statistic of POS tag-set in 12K corpus is as shown in Table 1.

5.2 Closed and Open Test Set

There are two types of test data: closed and open data sets. Both of them also taken from various Khmer websites and each test-set contained 1,000 sentences. In details, 11,449 words in total, 2,742 unique words and the average number of words per sentence is 10.40 for closed data-set. In total 11,824 words, 2,795 unique words and the average number of words per sentence is 10.78 for open data set.

5.3 Software

We used following open source POS Taggers for the experiments:

- CRFSuite: We used the CRFsuite tool (version 0.12) [38], (`https://github.com/chokkan/crfsuite`) for training and testing CRF models. The main reason was its speed relative to other CRF toolkits.

- Jitar (version 0.3.3): is a simple part-of-speech tagger, based on a trigram Hidden Markov Model (HMM). It (partly) implements the ideas set forth in [39]. Jitar is written in Java [40] and thus easy to use in other Java programs, or languages that run on the JVM.

- Maximum Entropy Modeling Toolkit for Python and C++: provides a (Conditional) Maximum Entropy Modeling. We used a python extension module (maxent module) for building Maximum Entropy POS tagger [41], [42], [43].

- RDRPOSTagger (Version 1.2.3): is a rule-based Part-of-Speech and morphological tagging toolkit [26], [25]. It is a is a robust, easy-to-use and language-independent toolkit. It employs an error-driven approach to automatically construct tagging rules in the form of a binary tree. The main properties of RDRPOSTagger are it obtains fast performance in both learning and tagging process and achieves a very competitive accuracy in comparison to the state-of-the-art results.

- KyTea: is a general toolkit (version 0.47) [33], (`https://github.com/neubig/kytea`) and it is able to handle word segmentation and tagging. It uses a point-wise classifier-based (SVM or logistic regression) approach and the classifiers are trained with LIBLINEAR (`http://www.csie.ntu.edu.tw/~cjlin/liblinear/`).

- Low-Resource POS-Tagging toolkit (2014): contains Scala code for training and tagging using the approach described in the papers [34], [44]. We used it for experiments of two hour annotation approach that we mentioned in Section 4.6.

We ran all above software with default parameters for building the POS tagging models. Although feature engineering is usually an important component of machine-learning approaches, the POS tagging models were built with features from only the corpus, to allow for a fair comparison between the six approaches.

6 Evaluation Criteria

The POS tagging performance was measured using the accuracy defined as follows:

$$Accuracy = \frac{\#of\ correct\ POS - tags}{\#of\ tokens\ in\ test\ corpus} \quad (8)$$

We also used the SCLITE (score speech recognition system output) program from the NIST scoring toolkit SCTK version 2.4.10 [45] for making dynamic programming based alignments between reference and hypothesis POS-tag strings and calculation of Word Error Rate (WER). In our case, WER will be equal to POS tagging error rate (POS-ERR). The SCLITE scoring method for calculating the erroneous words in WER: first make an alignment of the hypothesis (the output from the trained model) and the reference POS strings (POS-tagged manually) and then perform a global minimization of the Levenshtein distance function which weights the cost of correct words, insertions (I), selections (D), substitutions (S) and the number of POS (N). The formula for WER can be stated as Equation (9):

$$WER = \frac{(I + D + S) \times 100}{N} \quad (9)$$

For example, scoring I, D and S for the POS-tagged Khmer sentence អ្នកកំលោះ/NN នោះ/DT បាន/AUX ចូល/VB ធ្វើ_ជា/VB កម្មករ/NN ព្រះពិស្ណុការ/PN ។/KAN (”That man has served as worker of Pisnoka God.” in English) is as follow:

```
Scores: (#C #S #D #I) 7 0 1 1
REF: nn dt AUX vb ** vb nn pn kan
HYP: nn dt  *** vb IN vb nn pn kan
Eval:           D      I
```

In this case, one deletion (AUX => ***) and one insertion (** => IN) happened, that is $S = 0$, $D = 1$, $I = 1$, $C = 7$, $N = 8$ and thus WER or POS-ERR is equal to 25%.

7 Results

The accuracies of six POS-tagging methods that we trained with 12K POS-tagged sentences are shown in Table 2. Test data sizes are 1K sentences for both closed and open test data. Underlined numbers indicate the highest scores of the six different approaches. The experimental results show that RDR achieved the highest accuracy 95.33% with open-test data. On the other hand, SVM based point-wise classification gives highest accuracy 99.71% with closed-test data.

In order to study how six models behave with varying amounts of training data, we run a sequence of experiments that trained CRF, HMM, Max-Ent, RDR, SVM and Two-Hours models from training set 2K, 4K, 6K, 8K, 10K to 12K sentences respectively. From the results evaluation with 1K open test data in Figure 3, it is very clear that HMM and RDR models are learning very well for the whole training process. The final accuracy results of both methods are also comparable. Although the first result of SVM with 2K training data is lower than CRF, generally, their results are very close from 4K to 12K training sets. Interestingly, as the Two-Hours approach was considered true low resource scenarios [34], it was achived higher accuracies from 2K to 6K training data and comparable with CRF and SVM but ustable results produced from 8K to the final 12K.

Table 2. Accuracy of six POS tagging approaches with 12K model

Methods	**Closed Test-set**	**Open Test-set**
Methods	Accuracy	Accuracy
CRF	98.61	94.82
HMM	96.56	95.26
Max-Ent	95.32	92.97
RDR	97.14	95.33
SVM	99.71	94.57
Two-Hours	93.66	92.68

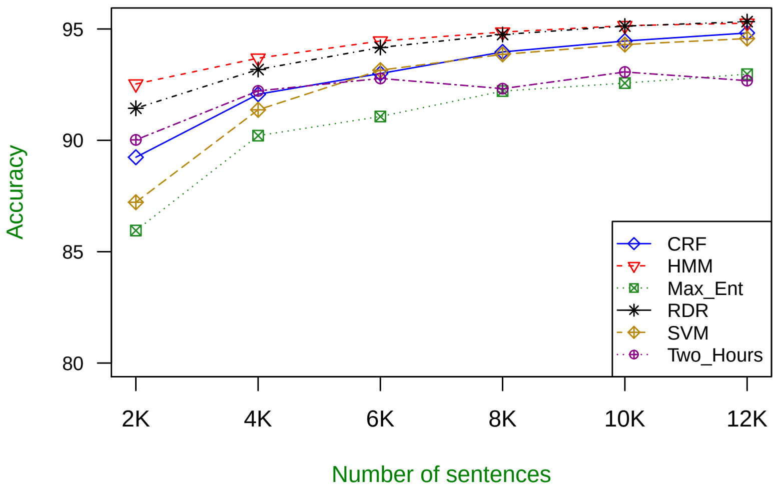

Figure 3. Accuracies of six POS tagging methodologies on varying training data sizes

8 Error Analysis

The error analysis of six POS tagging approaches has been done with the help of confusion matrix produced by the SCLITE program (see Section 6). The top 10 mistakes of six POS tagging models for both closed and open test-sets are shown in Table 3 to Table 8.

From our analysis, we found that IN->VB, RB->VB, VB->IN and VB_JJ->VB confusion pairs are exist in all six models. Four confusion pairs are occured in 5 POS tagging models. In details, JJ->NN, PN->NN and VB -> NN confusion pairs are found in CRF, HMM, Max-Ent, RDR and Two-Hours models. VB->VB_JJ confusion pair is found in CRF, HMM, RDR, SVM and Two-Hours models. NN->VB confusion pair is found in four models and they are CRF, HMM, Max-Ent and SVM. Similarly, RB->JJ is found in HMM, RDR, SVM and Two-Hours models. On the other hand, PN->NN pair is the highest and PRO->NN pair is the second highest pair of Two-Hours model. AUX -> VB POS tagging errors only found in Max-Ent model for both closed and open test-sets (Table 5). Unknown tag errors such as JJ->UNK, CD->UNK, NN->UNK, PN->UNK, VB->UNK were given by the SVM model (trained with KyTea program) for the evaluation with open test-set (see Table 7).

9 Discussion

We calculated OOV (Out of Vocabulary) for all incremental training data with 1K open test-set and the results are 934 OOVs for 2K training data, 537 OOVs for 4K training data, 390 OOVs for 6K training data, 307 OOVs for 8K training data, 259 OOVs for 10K training data and 233 OOVs for 12K training data. From the experimental results we confirmed that RDR model is able to achieve 95.33% accuracy even with 12K training data and 233 OOVs on current open test-set.

We studied all confusion pairs for six models and found that some confusion pairs such as VB_JJ->VB are same for the POS tagging methodologies (see Section 8). The main reason is as we explained in the definition of the VB_JJ in Section 3, it's original is the tag VB and it is really depends on the context or the meaning. For example, in the Khmer sentence សូម្បី_តែ លុយ អង្គការ ឧបត្ថម្ភ គ្រូ បន្ទុក ថ្នាក់ ៣ ដុល្លារ ក៏ គាត់ កាត់ យក ១ ដុលារ ដែរ ។, here, the reference POS tag of ឧបត្ថម្ភ word ("support" in English) is "VB_JJ" because the following word is គ្រូ ("teacher" in English), but models wrongly predicted as "VB". We also found that some errors are occured because of manual POS tagging errors. For example, one of the confusion pair of Two-Hours model "PN-> NN" on the Khmer sentence ការ ឃុំ_ខ្លួន បុគ្គលិក របស់ អង្គការ សមធម៌ កម្ពុជា ដោយ ខុស ច្បាប់ ក្រុម សង្គម ស៊ីវិល, the reference tag of the word កម្ពុជា should be "NN". The SVM model have many "UNK" errors for the OOV words such as អព្យាក្រឹតភាពNN ("neutrality" in English) in the Khmer sentence ចៅក្រម មិន មាន អព្យាក្រឹតភាព មិន ឯក_រាជ្យ ។.

RDRPOSTagger employs an error-driven approach to automatically construct tagging rules in the form of a binary tree and it obtains fast performance in both learning and tagging process. Moreover, one of the merit points of the RDR approach is producing human readable SCRDR rules as a trained model and that will be useful for analysis on word-category disambiguation of the Khmer language. The following is a part of SCRDR tree that we trained with 12K:

```
True : object.conclusion = "NN"
 object.tag == "VB_JJ" : object.conclusion = "VB_JJ"
   object.prevTag1 == "VB" : object.conclusion = "VB"
     object.prevWord2 == "គម្រោង" : object.conclusion = "VB_JJ"
     object.prevTag1 == "VB" and object.word == "ចុះ_បញ្ជី" : object.conclusion = "NN"
```

10 Conclusion and Future Work

In this paper, we conducted six POS tagging experiments on Khmer language with our developing POS-tagged corpus. We found that RDR approach can achieved accuracy 95.33% on open data set and best among six POS tagging methods. CRF, HMM and SVM approaches also achieved comparable results with RDR. In further work, we plan to check again on errors of manual segmentation and POS tagging of the whole corpus and re-evaluate with cross-validation. We plan to release our POS-tagged corpus including trained models for Khmer language NLP research in December 2017.

Acknowledgment

We would like to express our gratitude to Mr. Sorn Kea and Miss Leng Sreyhuy for their help in POS tagging 12,100 sentences of kh-POS Corpus manually.

Table 3. Most common mistakes with CRF model

Closed Test-set		**Open Test-set**	
Frequency	REF –> HYP	Frequency	REF –> HYP
20	VB –> IN	56	VB_JJ –> VB
15	VB_JJ –> VB	45	VB –> NN
9	IN –> VB	32	PN –> NN
9	VB –> VB_JJ	31	JJ –> NN
8	JJ –> NN	29	VB –> IN
7	RB –> VB	27	NN –> VB
5	JJ –> VB	23	RB –> VB
5	NN –> VB	19	IN –> VB
5	RB –> IN	17	JJ –> VB
5	VB –> NN	16	VB –> VB_JJ

Table 4. Most common mistakes with HMM model

Closed Test-set		**Open Test-set**	
Frequency	REF –> HYP	Frequency	REF –> HYP
43	VB –> IN	53	VB_JJ –> VB
31	VB –> VB_JJ	38	VB –> IN
31	VB_JJ –> VB	31	VB –> VB_JJ
16	IN –> VB	29	PN –> NN
13	PN –> NN	25	VB –> NN
12	NN –> JJ	19	NN –> VB
12	RB –> JJ	19	RB –> VB
12	RB –> VB	17	JJ –> NN
11	PRO –> NN	17	NN –> PN
9	CC –> RB	16	PRO –> NN

Table 5. Most common mistakes with Maximum Entropy model

Closed Test-set		**Open Test-set**	
Frequency	REF –> HYP	Frequency	REF –> HYP
60	VB_JJ –> VB	78	VB_JJ –> VB
30	JJ –> NN	56	IN –> VB
25	VB –> NN	45	NN –> VB
22	IN –> VB	41	VB –> NN
21	JJ –> VB	38	JJ –> NN
19	AUX –> VB	34	AUX –> VB
19	VB –> IN	34	PN –> NN
18	RB –> VB	30	JJ –> VB
18	VCOM –> VB	27	RB –> VB
17	PN –> NN	27	VB –> IN

Table 6. Most common mistakes with RDR model

Closed Test-set		**Open Test-set**	
Frequency	REF –> HYP	Frequency	REF –> HYP
40	VB_JJ –> VB	61	VB_JJ –> VB
27	VB –> IN	41	VB –> NN
14	IN –> VB	32	PN –> NN
13	NN –> PN	29	VB –> IN
12	VB –> VB_JJ	25	JJ –> NN
9	CC –> RB	22	IN –> VB
8	JJ –> RB	16	RB –> VB
8	JJ –> VB	16	VB –> VB_JJ
8	NN –> JJ	14	NN –> PN
8	RB –> JJ	10	NN –> PRO

Table 7. Most common mistakes with SVM model

Closed Test-set		**Open Test-set**	
Frequency	REF –> HYP	Frequency	REF –> HYP
10	VB –> VB_JJ	106	NN –> UNK
4	NN –> JJ	59	VB_JJ –> VB
2	RB –> IN	40	VB –> UNK
2	RB –> JJ	34	PN –> UNK
1	IN –> VB	27	VB –> VB_JJ
1	JJ –> RB	25	IN –> VB
1	JJ –> VB_JJ	24	VB –> IN
1	NN –> VB	19	JJ –> UNK
1	RB –> VB	14	CD –> UNK
1	RB –> VB_JJ	13	RB –> VB

Table 8. Most common mistakes with Two-Hours model

Closed Test-set		**Open Test-set**	
Frequency	REF –> HYP	Frequency	REF –> HYP
75	PN –> NN	89	PN –> NN
61	PRO –> NN	75	PRO –> NN
46	VB –> IN	48	VB –> VB_JJ
46	VB –> VB_JJ	48	VB_JJ –> VB
34	VB –> AUX	42	VB –> IN
33	IN –> AUX	32	IN –> AUX
33	VB_JJ –> VB	28	VB –> AUX
23	IN –> VB	21	VB –> NN
17	RB –> VB	20	RB –> VB
15	RB –> JJ	18	JJ –> NN

References

[1] Chenda Nou and Wataru Kameyama. *Transformation-based Khmer Part-of-Speech tagger*, volume 2, pages 581–587. 2007.

[2] Eric Brill. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Comput. Linguist.*, 21(4):543–565, December 1995.

[3] C. Nou and W. Kameyama. Hybrid approach for khmer unknown word pos guessing. In *2007 IEEE International Conference on Information Reuse and Integration*, pages 215–220, Aug 2007.

[4] PAN Localization Cambodia (PLC) of IDRC. Part of speech template. In *Cambodia Country Component, PAN Localization Project, PAN Localization Cambodia (PLC) of IDRC*, pages 1–23, 2007.

[5] PAN Localization Cambodia (PLC) of IDRC. Research report on khmer automatic pos tagging. In *Cambodia Country Component, PAN Localization Project, PAN Localization Cambodia (PLC) of IDRC*, pages 1–10, 2008.

[6] Helmut Schmid. Treetagger, 1994.

[7] Helmut Schmid. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*, Manchester, UK, 1994.

[8] Helmut Schmid. Improvements in part-of-speech tagging with an application to german. In *In Proceedings of the ACL SIGDAT-Workshop*, pages 47–50, 1995.

[9] PAN Localization Cambodia (PLC) of IDRC. Khmer pos tagged corpus, 2009.

[10] Chenchen Ding, Hour Kaing, Vichet Chea Masao Utiyama, and Eiichiro Sumita. Tokenization and part-of-speech annotation guidelines for khmer (cambodian), (version 0.2, december 2016), 2016.

[11] Ye Kyaw Thu, Win Pa Pa, Masao Utiyama, Andrew Finch, and Eiichiro Sumita. Introducing the asian language treebank (alt). In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*, Paris, France, may 2016. European Language Resources Association (ELRA).

[12] Hammam Riza, Michael Purwoadi, Gunarso, Teduh Uliniansyah, Aw Ai Ti, Sharifah Mahani Aljunied, Luong Chi Mai, Vu Tat Thang, Nguyen Phuong Thai, Vichet Chea, Rapid Sun, Sethserey Sam, Sopheap Seng, Khin Mar Soe, Khin Thandar Nwet, Masao Utiyama, and Chenchen Ding. Introduction of the asian language treebank. In *Proceedings of the 2016 Conference of The Oriental Chapter of International Committee for Coordination and Standardization of Speech Databases and Assessment Technique (O-COCOSDA)*, pages 1–6, Oct 2016.

[13] Chenchen Ding, Hour Kaing, Vichet Chea Masao Utiyama, and Eiichiro Sumita. Sample data of word segmentation and pos tags following nova, 2016.

[14] Chea Vichet, Ye Kyaw Thu, Ding Chenchen, Utiyama Masao, Finch Andrew, and Eiichiro Sumita. Khmer word segmentation using conditional random fields. In *In Khmer Natural Language Processing 2015 (KNLP2015)*, Phnom Penh, Cambodia, December 2015.

[15] Madeline Elizabeth. Ehrman, Kem Sos, Foreign Service Institute (U.S.), and Defense Language Institute (U.S.). *Contemporary Cambodian: grammatical sketch, by Madeline E. Ehrman, with the assistance of Kem Sos.* Foreign Service Institute, Dept. of State; [for sale by the Supt. of Docs., U.S. Govt. Print. Off.] Washington, 1972.

[16] John D. Lafferty, Andrew McCallum,

and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[17] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, February 1989.

[18] Daniel Jurafsky and James H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition.* Prentice Hall PTR, Upper Saddle River, NJ, USA, 1st edition, 2000.

[19] E. T. Jaynes. Information theory and statistical mechanics. *Phys. Rev.*, 106:620–630, May 1957.

[20] Adam L. Berger, Stephen A. Della Pietra, and Vincent J. Della Pietra. A maximum entropy approach to natural language processing. *COMPUTATIONAL LINGUISTICS*, 22:39–71, 1996.

[21] Adwait Ratnaparkhi. A maximum entropy model for part-of-speech tagging. In *Conference on Empirical Methods in Natural Language Processing*, pages 133–142, 1996.

[22] Kristina Toutanova and Christopher D. Manning. Enriching the knowledge sources used in a maximum entropy part-of-speech tagger. 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 - Volume 13*, EMNLP '00, pages 63–70, 2000.

[23] Pascal Denis and Benoît Sagot. Coupling an annotated corpus and a lexicon for state-of-the-art POS tagging. *Language Resources and Evaluation*, 46(4):721–736, 2012.

[24] P. Compton and R. Jansen. A philosophical basis for knowledge acquisition. *Knowledge Acquisition*, 2(3):241 – 258, 1990.

[25] Dat Quoc Nguyen, Dai Quoc Nguyen, Dang Duc Pham, and Son Bao Pham. A robust transformation-based learning approach using ripple down rules for part-of-speech tagging. *AI Communications*, 29(3):409–422, 2016.

[26] Dat Quoc Nguyen, Dai Quoc Nguyen, Dang Duc Pham, and Son Bao Pham. RDRPOSTagger: A Ripple Down Rules-based Part-Of-Speech Tagger. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 17–20, Gothenburg, Sweden, April 2014. Association for Computational Linguistics.

[27] Debbie Richards. Two decades of ripple down rules research. *Knowledge Eng. Review*, 24(2):159–184, 2009.

[28] Tobias Scheffer. Algebraic foundation and improved methods of induction of ripple down rules. In *In*, pages 23–25, 1996.

[29] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Mach. Learn.*, 20(3):273–297, September 1995.

[30] Jesús Giménez and Lluís Màrquez. Svmtool: A general pos tagger generator based on support vector machines. In *In Proceedings of the 4th International Conference on Language Resources and Evaluation*, 2004.

[31] Taku Kudoh and Yuji Matsumoto. Use of support vector learning for chunk identification. In *Proceedings of the 2Nd Workshop on Learning Language in Logic and the 4th Conference on Computational Natural Language Learning - Volume 7*, ConLL '00, pages 142–144, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

[32] Tetsuji Nakagawa, Taku Kudo, and Yuji Matsumoto. Unknown word guessing and part-of-speech tagging using support vector machines. In *Proceedings*

of the Sixth Natural Language Processing Pacific Rim Symposium, November 27-30, 2001, Hitotsubashi Memorial Hall, National Center of Sciences, Tokyo, Japan, pages 325–331, 2001.

[33] Graham Neubig and Shinsuke Mori. Word-based partial annotation for efficient corpus construction. In *The seventh international conference on Language Resources and Evaluation (LREC 2010)*, pages 2723–2727, Malta, May 2010.

[34] Dan Garrette and Jason Baldridge. Learning a part-of-speech tagger from two hours of annotation. pages 138–147, June 2013.

[35] Partha Pratim Talukdar and Koby Crammer. New regularized algorithms for transductive learning. In *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases: Part II*, ECML PKDD '09, pages 442–457, Berlin, Heidelberg, 2009. Springer-Verlag.

[36] Sujith Ravi, Ashish Vaswani, Kevin Knight, and David Chiang. Fast, greedy model minimization for unsupervised tagging. In Chu-Ren Huang and Dan Jurafsky, editors, *COLING*, pages 940–948. Tsinghua University Press, 2010.

[37] Dan Garrette and Jason Baldridge. Type-supervised hidden markov models for part-of-speech tagging with incomplete tag dictionaries. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 821–831, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

[38] Naoaki Okazaki. Crfsuite: a fast implementation of conditional random fields (crfs), 2007.

[39] Thorsten Brants. Tnt: A statistical part-of-speech tagger. In *Proceedings of the Sixth Conference on Applied Natural Language Processing*, ANLC '00, pages 224–231, Stroudsburg, PA, USA, 2000. Association for Computational Linguistics.

[40] Daniël de Kok. Jitar: A simple trigram hmm part-of-speech tagger, 2014. [accessed 2016].

[41] Le Zhang. Maximum entropy modeling toolkit for python and c++, 2003. [accessed 2016].

[42] Adam L. Berger, Vincent J. Della Pietra, and Stephen A. Della Pietra. A maximum entropy approach to natural language processing. *Comput. Linguist.*, 22(1):39–71, March 1996.

[43] Stephen Della Pietra, Vincent J. Della Pietra, and John D. Lafferty. Inducing features of random fields. *IEEE Trans. Pattern Anal. Mach. Intell.*, 19(4):380–393, 1997.

[44] Dan Garrette, Jason Mielens, and Jason Baldridge. Real-world semi-supervised learning of pos-taggers for low-resource languages. pages 583–592, August 2013.

[45] (NIST) The National Institute of Standards and Technology. Speech recognition scoring toolkit (sctk), version: 2.4.10, 2015.