A Word Labelling Approach to Thai Sentence Boundary Detection and POS Tagging

Nina Zhou
Institute for Infocomm Research
zhoun@i2r.a-star.edu.sg

Aw AiTi
Institute for Infocomm Research
aaiti@i2r.a-star.edu.sg

Nattadaporn Lertcheva
Institute for Infocomm Research
lertchevan@i2r.a-star.edu.sg

Wang Xuangcong
Institute for Infocomm Research
wangxc@i2r.a-star.edu.sg

Abstract

Previous studies on Thai Sentence Boundary Detection (SBD) mostly assumed a sentence ends at a space and formulated the task SBD as a disambiguation problem, which classified a space either as an indicator for Sentence Boundary (SB) or non-Sentence Boundary (nSB). In this paper, we propose a word labelling approach which treats the space character as a normal word, and detects SB between any two words. This removes the restriction for SB to be occurred only at spaces and makes our system more robust for modern Thai writing. It is because in modern Thai writing, the space is not consistently used to indicate SB. As syntactic information contributes to better SBD, we further propose a joint Part-Of-Speech (POS) tagging and SBD framework based on Factorial Conditional Random Field (FCRF) model. We compare the performance of our proposed approach with reported methods on ORCHID corpus. We also performed experiments of FCRF model on the TaLAPi corpus. The results show that the word labelling approach has better performance than previous space-based classification approaches and FCRF joint model outperforms LCRF model in terms of SBD in all experiments.

1 Introduction

Sentence Boundary Detection (SBD) is a fundamental task for many Natural Language Processing (NLP) and analysis tasks, including POS tagging, syntactic, semantic, and discourse parsing, parallel text alignment, and machine translation (Gillick, 2009). Most research on SBD focus on languages that already have a well-defined concept of what a sentence is, typically indicated by sentence-end markers like full-stops, question marks, or other punctuations. However, as we study more contexts of language use (e.g. speech output which lacks punctuations) as well as look at many more different languages, the assumption of clearly-punctuated sentence boundary becomes less valid. One such language is Thai.

In prior research on Thai, the space character has been regarded as a very important element in Thai SBD (Pradit *et al.*, 2000; Paisarn *et al.*, 2001; Glenn *et al.*, 2010). These regard that space characters are always present between sentences. However, in actual fact, as prescribed by Thai linguistic authorities (www.royin.go.th) as well as what can be observed in real texts, spaces do exist in Thai texts not only in such sentence-end contexts. There is some pressure from linguistic authorities (Wathabunditkul, 2003) to set orthographic standards in Thai, prescribing the use of spaces in the context of certain words, following the rules of the Thai Royal Institute Dictionary[1]. Examples of these rules include: using of the space before and after an interjection or an onomatopoeiac word โอ๊ย (Ouch!), อุ๊ย (ow!); before conjunctions และ (*and*), หรือ (*or*), and แต่ (*but*); before and after a numeric expression: มีนักเรียน 20 คน (have 20 students), เวลา 10.00 น. (time 10.00 a.m.). Unfortunately, the rules are not strictly followed in practice and the use of spaces between words, phrases, clauses and sentences vary across different users of the Thai language. According to TaLAPi (Aw et al. 2014), a news domain corpus, it has about 23% sentences ending without a space character. One example of the Thai text from TaLAPi corpus is

[1] https://en.wikipedia.org/wiki/Royal_Institute_Dictionary

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 319–327, Osaka, Japan, December 11-17 2016.

shown in Figure 1, in which the space character is used within a sentence, but not as a sentence-end indicator.

In view of this complexity of spaces in Thai in light of the SBD task, we propose a word-based labelling approach which regards Thai SBD as a word labelling problem instead of a space classification problem. The approach treats the space as a normal word and labels each word as SB or nSB (non-Sentence Boundary). Figure 2 illustrates the space-based classification approach versus the word-based labelling approach.

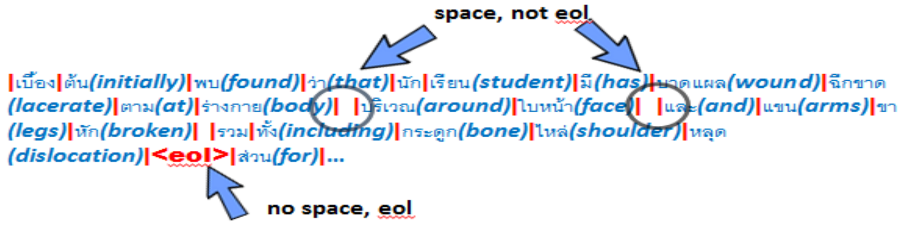
space, not eol
|เบื้อง|ต้น(initially)|พบ(found)|ว่า(that)|นัก|เรียน(student)|มี(has)|บาดแผล(wound)|ฉีกขาด
(lacerate)|ตาม(at)|ร่างกาย(body)| |บริเวณ(around)|ใบหน้า(face)| |และ(and)|แขน(arms)|ขา
(legs)|หัก(broken)| |รวม|ทั้ง(including)|กระดูก(bone)|ไหล่(shoulder)|หลุด
(dislocation)|<eol>|ส่วน(for)|...
no space, eol

Figure 1. Example of a written Thai text in which there are two space characters within the first sentence, but there is no space character at the end of the sentence, i.e., at highlighted <eol>". "eol" refers to end-of-line.

The proposed word labelling approach formulates SBD as a typical sequence labelling task, i.e., labelling each word including spaces as a SB or nSB. It is tested on ORCHID corpus and demonstrates higher accuracy on SB than previous methods (Pradit *et al.*, 2000; Paisarn *et al.*, 2001; Glenn *et al.*, 2010). Furthermore, the contribution of POS in this task is investigated and a Joint framework for POS tagging and SBD is formulated. The results on TaLAPi corpus show that POS information can improve the accuracy of SBD, for both the sequential task of POS tagging followed by SBD and the proposed joint framework. Moreover, in the joint framework, we propose a two-layer classification for POS tagging, which is called as "2-step" Joint approach in the following paper. For comparison, the joint approach in which POS tagging realized in one step is called as "1-step" Joint approach. The proposed "2-step" Joint approach runs considerable faster and achieves similar performance when compared with the Cascade approach and "1-step" Joint approach of POS tagging and SBD. By adding enhanced features, the "2-step" Joint approach yields better SBD accuracy and comparable POS tagging accuracy.

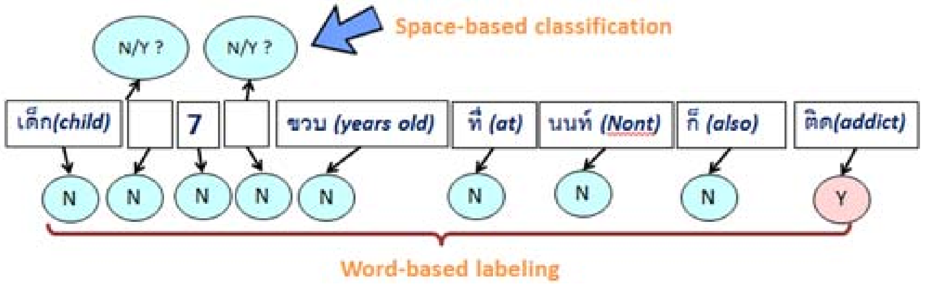

Figure 2. Space-based SBD vs word-based labelling SBD. Space-based SBD detects spaces and assigns Y (SB) or N (nSB) to each space. Word-based labelling assigns Y(SB) or N(nSB) to every word. In this case, the space character is considered as a word.

The rest of the paper is organized as follows. Section 2 reviews the previous studies on Thai SBD. Section 3 describes the proposed word labelling framework and the approaches. Section 4 compares the performance between the proposed word-based methods and reported space-based methods on ORCHID corpus (Sornlertlamvanich *et al.*, 1997), and also studied the different frameworks of word-based approaches. Section 5 concludes the paper.

2 Previous Studies

There have been limited studies carried out in Thai SBD over the past twenty years. Longchupole (1995) presented a method to segment a paragraph into small units and then used verbs to estimate the number of sentences. That was a grammatical rule based approach to extract sentences from paragraphs. The reported SBD accuracy was 81.18% (Longchupole, 1995). Pradit *et al.* (2000) applied the statistical POS tagging technique (Brants, 2000) on the detection of SB. They considered SB and non-

SB as POS tags and distinguished SB from other POS tags based on a trigram model. Their method yielded an accuracy of 85.26% on ORCHID corpus. Paisarn *et al.* (2001) utilized the Winnow algorithm to extract features from the context around the target space. The Winnow functioned like a neuron network model where a few nodes were connected to a target node. Each node examined only two features for simplicity. In total, there were 10 features including words around target space and their POS information. The space-correct accuracy for the Winnow on ORCHID was 89.13%. Later, Glenn *et al.* (2010) proposed to use maximum entropy classifier to distinguish each space as SB or non-SB and their results were shown to be consistent with the Winnow (Paisam *et al*., 2001).

Nearly all Thai SBD studies are based on the assumption that there is a space at the position of the SB. While we have shown in the Introduction part that sentence break is not always indicated by a space, especially in modern Thai writing. That inspired us to propose the word-based approach to consider a space as a word and treats SBD as a word labelling task instead of a space disambiguation problem

The word-based approach is further enhanced to label POS tags and SB jointly using joint inferencing. The advantages of this approach are: 1) it relies on contexts instead of spaces to detect SB, 2) it solves SBD and POS tagging jointly to relax the dependency of POS tagging for SBD, 3) it demonstrates higher accuracy on SBD than previous methods (Pradit *et al.*, 2000; Paisarn *et al.*, 2001; Glenn *et al*., 2010).

3 The Models

CRFs (Lafferty *et al.*, 2001; Sutton *et al*., 2011) have demonstrated their strengths of sequence labelling in NLP tasks (McCallum *et al.*, 2003; Liu *et al*., 2005; Sun *et al.*, 2011). They rely on the capacity to capture the sequence's observation O_n $\{i=1,2\ldots\text{n}\}$ (abbreviated as O) and at the same time the local dependency L_i $\{i=1,2,\ldots\text{n}\}$ (abbreviated as L) among nodes in the sequence (see Figure 3 for the example of linear-chain CRF (LCRF) (Sutton et al., 2011)). Conditioned on observations O, dependencies of L form the chain. In the model, the probability of labelling an observed input O with a label sequence L is defined by a conditional probability as in Equation (1):

$$p_\lambda(L \mid O) = \frac{1}{Z(O)} \exp\left(\sum_t \sum_k \lambda_k f_k(O, L, t)\right) (1)$$

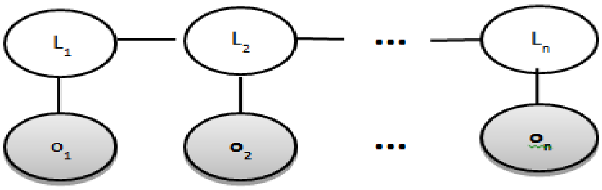

Figure 3. Linear-chain graph CRF (LCRF)

where $\{f_k\}$ is a set of feature functions defined over the observation O and label sequence L at each position t, together with the set of corresponding weights $\{\lambda_k\}$; z(O) is a normalization factor.

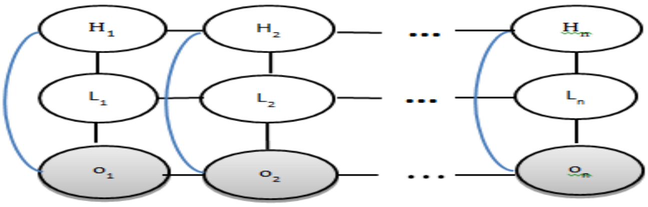

Figure 4. Two-layer Factorial CRF (FCRF)

Dynamic CRF (DCRF) (Sutton *et al.,* 2004; Sutton *et al*., 2011) is a generalization of LCRF, which supports any arbitrary structure graph. It is formally defined as in Equation (2):

$$p_\lambda(L \mid O) = \frac{1}{Z(O)} \exp\left(\sum_t \sum_{c \in C} \sum_k \lambda_k f_k(O, L_{(c,t)}, t)\right) (2)$$

where C is a set of cliques indices which connect the nodes in a sequence in a single layer or among different layers. As a special case of DCRF, Factorial CRF (FCRF) model allows multiple layers' labelling simultaneously for a given sequence. The graphical illustration of two-layer FCRF is shown in Figure 4 where *H* indicates the 1st layer labels and *L* indicates the 2nd layer labels. *O* is the observation sequence. Through the connections between different layers of labels and the given observation, joint conditional distributions of the labels are learnt.

3.1 Isolated and Cascade SB and POS Tagging

We use LCRF for the single task of SB detection or POS tagging. In this scenario, as Thai SB is to detect word sequence and find the end of each sentence, we consider this to be similar to a sentence-end punctuation prediction task with only two labels. Words are labelled with SB if they are at the beginning of a sentence otherwise, they will be labelled as nSB. For POS tagging, we use all the 35 subcategories as described in Aw et al. (2014).

Feature Template	Window Size
w_0	3 or 5
$w_{-1}+w_0$	3 or 5
$w_{-1}+w_0+w_1$	3
$wtype_0$	3 or 5
$wtype_{-1}+wtype_0$	3 or 5
$wtype_{-1}+wtype_0+wtype_1$	3

Table 1. The feature template for Isolated SBD and POS tagging. Window size 3 is used for Isolated SBD and POS tagging. Window size 5 is used in “2-step” Joint model (vii and viii) in Table 6.

Considering POS tagging has much more labels to recognize than SBD, it will increase the memory use for system training, therefore the number of the features and feature template have to be carefully selected. It is essential to use a simple feature set, as shown in Table 1, to make a comparison between Isolated models, Cascade models and the Joint models. It is important for “1-step” Joint model as more features make the process run extremely slow. In Table 1, w_i refers to the word at the ith position relative to the current node; window size is the maximum span of words centered at the current word that the template covers, e.g., $w_{-1}+w_0$ with a window size of 3 refers to $w_{-2}+w_{-1}$, $w_{-1}+w_0$, and w_0+w_1; $wtype_i$ indicates the word type at the ith position relative to the current node; In total, five word types are defined, i.e., English, Thai, punctuation, digits and spaces, for the data used in our experiments.

Feature Template	Window size
pos_0	3 or 5
$pos_{-1}+pos_0$	3
$pos_{-1}+pos_0+pos_1$	3

Table 2. The additional feature template for Cascade models, besides the feature templates in Table 1. Window size 5 is used in Cascade model (iv) in Table 6.

As POS tag provides additional syntactic and some semantic information to the word, they are utilized as additional features to the Cascade approach for detecting the sentence boundary. Besides the features listed in Table 1, more POS features listed in Table 2 are used in the Cascade models.

	Original POS	Pseudo POS	Total No.		Original POS	Pseudo POS	Total No.
1	NN,NR,PPER,PINT, PDEM	NPs	104271	7	CL	CL	5747
2	REFX	REFX	1357	8	OD,CD	OCD	8453
3	DPER,DINT,DDEM, PDT	DPs	7267	9	FXN, FXG, FXAV, FXAJ	FXs	13887
4	JJA, JJV	JJs	14335	10	P, COMP, CNJ	PCs	50301
5	VV, VA, AUX	VVs	72769	11	FWN,FWV,FWA,FWX	FWs	24
6	ADV,NEG	ADs	12275	12	PAR, PU, IJ, X	Os	6270

Table 3. The mapping from original POS to Pseudo POS tags

3.1.1. "1-step and "2-step" Joint Models

The joint model realizes the 2-layer labelling of one sequence using FCRF. We consider the first layer as labels of SBD, and the second layer as labels of POS tagging (see Table 3). However, due to the large number of POS tags, combining the feature templates of both tasks increases the search space tremendously and has a large impact on the processing speed. To address this problem, we propose a "2-step" Joint-model in which we first predict 12 top categories of the POS tags as classified in (Aw *et al.*, 2014) and then restore the pseudo POS tags back to the original POS tags (see Figure 5). On the other side, the "1-step" Joint model uses all the 35 POS tags to realize POS tagging in the 2nd layer of FCRF.

To train the "2-step" Joint model, all train data are labelled with two SB labels (i.e., SB and nSB) and 12 pseudo POS tags. The 12 pseudo POS tags are obtained by combining similar POS tags into one category as illustrated in Table 3. The Original POS column lists the original 35 POS tags and the Pseudo POS column lists the corresponding 12 pseudo POS tags. To restore the pseudo POS tags back to the original tags, we train different LCRF models for each pseudo POS tag. As no restoration is required for "CL" and "REFX", a total of 10 LCRF models are built to restore the original POS tags. The diagram of the proposed "2-step" Joint model is shown as follows (Figure 5).

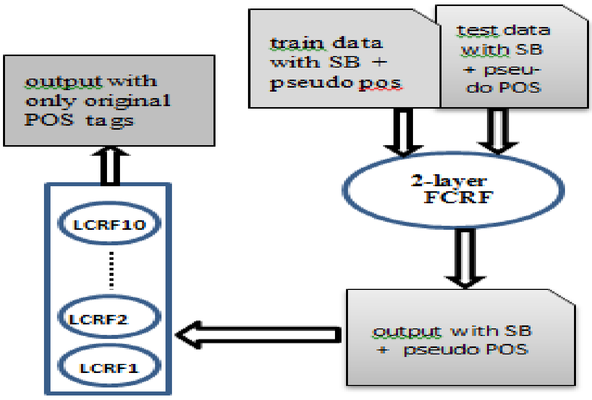
train data with SB + pseudo pos
test data with SB + pseu-do POS
output with only original POS tags
2-layer FCRF
LCRF10
LCRF2
LCRF1
output with SB + pseudo POS

Figure 5. The proposed "2-step" Joint model for Thai SBD and POS Tagging based on two-layer FCRF and the LCRF

For fair comparison between Isolated and Joint models, we used the same feature templates in Table 1 in two of the "2-step" Joint models, i.e., (v) and (vi) in Table 6. Since the "2-step" Joint model run much faster than "1-step" Joint, more features can be added. As in Table 4 shown, name entity recognition (NER) information was added to improve the performance of the "2-step" joint models besides the feature template in Table 1.

Feature Template	Window Size
NER_0	3
$NER_{-1}+NER_0$	3

Table 4. The enhanced feature template for "2-step" Joint model (viii) in Table 6

4 Experimentation

4.1 Data Preparation

Our experiments were performed on the ORCHID corpus (Sornlertlamvanich *et al.*, 1997) and the TaLAPi corpus (Aw *et al.*, 2014).

The processing of the ORCHID corpus follows the work of Sornlertlamvanich *et al.* (1997) to remove all comments and concatenate all sentences and paragraphs. Different from the previous experiments, we did not insert a space at the end of sentence if it was not originally present. As such, the percentage of sentences ending without a space was almost 100% for the ORCHID corpus used in our experiment. We portioned the ORCHID data into 10 parts with equal size and used 10 fold cross validation for evaluation.

The experiments on TaLAPi corpus were performed only on the news domain which was annotated with word segmentation, POS tags and name entities. It had 3633 paragraphs, 10,478 sentences and

311,637 words. We split 80% for training and 20% for testing. During the splitting, we tried to balance the distribution of spaces and POS tags. Thus in the training data, we have a total of 282,678 words, of which 8,034 words (2.842%) are SB and 274,644 words are nSB. In the test data we have 2,091 (2.836%) SB and 71,635 nSB.

4.2 Experimental results

The GRMM toolkit (Sutton, 2006) was used in our experiments to build the 2-layer FCRF models and one layer LCRF models. To demonstrate the proposed methods, we performed 5 different experiments as follows:

Orchid Corpus

i. Isolated LCRF model to detect SBD using POS information to make it comparable with reported work.

TaLAPi Corpus

ii. Isolated LCRF model for POS tagging and SBD without POS information for SBD.
iii. Cascade LCRF model on SB utilizing same feature as (i) and POS information with different feature templates.
iv. "1-step" Joint model using same features as (ii)
v. "2-step" Joint model using same features as (ii) and with additional features and different feature configurations.

	POS-trigram(%)	**Winnow(%)**	**ME(%)**	**our work(%)**
sb-precision	74.35	92.69	86.21	93.64
sb-recall	79.82	77.27	83.50	89.84
sb-fscore	76.98	84.28	84.83	91.70
nsb-precision	90.27	91.48	93.18	99.27
nsb-recall	87.18	97.56	94.41	99.56
nsb-fscore	88.70	94.42	93.79	99.41
space correct	85.26	89.13	91.19	95.91

Table 5. Comparison of our word-labelling approach based on LCRF (last column) with previous studies on ORCHID corpus. POS-trigram (Pradit *et al.*, 2000); Winnow (Paisarn *et al.*, 2001); ME (*Glenn et al.*, 2010). Space correct =(#correct sb+#correct nsb)/(total # of space tokens). '#' indicate the number of items followed.

In the ORCHID corpus experiment, we used the features described in Table 1 and Table 2. Table 5 shows the result of the word-labelling approach and its comparison with reported methods. Compared to the reported results (Glenn *et al.*, 2010), our word-labelling approach yielded consistent improvement on precision, recall, F-score for both SB and non-SB and also "space correct". Our SB precision is 1% higher than Winnow method and our recall is 6.3% higher than ME method. The F-score is 7% higher than Winnow and ME. As mentioned in 4.1, not all sentence boundaries in ORCHID are indicated by space. To have a fair comparison, we consider all sentence boundaries as a "space" when calculating "space correct" (Glenn *et al.*, 2010). In Table 5, the short form "sb" and "nsb" refers to the sentence break and non-sentence break respectively.

For the experiments on TaLAPi corpus, we study the performance in Isolated, Cascade and Joint model. The same experiment can be run on ORCHID corpus, but due to time and space limitation, we only show the results of the experiments on TaLAPi corpus in Table 6.

Comparing Cascade with Isolated Model

All Cascade models have higher F-score than the Isolated model. The best F-score of the Cascade model is 67.29% when we used 18 features in the experiment (iv). The experiment affirms that POS information is helpful in sentence boundary detection.

Comparing Isolated, Cascade with Joint Model

With the same set of features as in (i), "1-step" Joint (v) yields 3% increase on recall and 2% increase on F-score for SBD when compared to the Isolated model in (i). Comparing (vi) with (i), a similar increase in accuracy for SBD, with the same features, is observed. These results demonstrate that SBD can benefit from the other layer’s label information, i.e., POS tagging labels, in the Joint model (v). When compared to Cascade models, “1-step” Joint shows comparable SB detection performance with the Cascade model (iii) that uses additional 3-gram POS features. By enhancing the feature set for SB detection in the Cascade model (iv), we yielded 1% increase on F-score when compared to the Cascade model (iii).

	Description	**POS**	**SBD**		
		Accuracy	**Precision**	**Recall**	**F-score**
i	Isolated without pos information for SB	94.24	82.67	53.37	64.86
ii	Cascade with same features as (i) and 1-gram POS		81.93	54.85	65.71
iii	Cascade with same features as (i) and 3-gram POS		80.26	56.38	66.24
iv	Cascade with same features as (iii) and window size of 5		79.12	58.53	67.29
v	“1-step” Joint with same features as (i)	94.64	81.72	56.24	66.63
vi	“2-step” Joint with same features as (i)	95.46	82.29	55.57	66.34
		94.49			
vii	“2-step” Joint with same features as (vi) and window size of 5	95.63	80.52	58.29	67.62
		94.99			
viii	“2-step” Joint with same features as (vii) and NER	95.64	80.36	60.26	68.87
		94.99			

Table 6. Comparison of our methods based on FCRF and LCRF on TaLAPi corpus.

Comparing “1-step” with “2-step”

While the Cascade models and “1-step” Joint model was limited by the running speed due to the number of POS tags, the “2-step” Joint model was therefore proposed to improve the running speed and not degrade the accuracy. With the same set of features, the “2-step” Joint (vi) run much faster than “1-step” Joint (v), while yielded almost the same SBD F-score as (v). The run time comparison can be found in Table 7. Experiments were run on Intel(R) Xeon(R) 8 core processor E5-2667 V2 3.30GHz, 25M cache with multi-thread 16.

Different models	**Train time (s)**	**Test time (s)**	**All process** (hr)
Isolated SB (i)	389.1	4.012	0.11
Isolated POS (i)	33230.8	98.2	9.26
Cascade (ii)	33938.2	101.3	9.46
Cascade (iii)	33992.7	102.4	9.48
Cascade (iv)	34181.3	103.8	9.54
“1-step” Joint (v)	41009.8	125.79	11.43
“2-step” Joint (vi)	12895.7	60.147	3.61
“2-step” Joint (vii)	16543.8	74.065	4.62
“2-step” Joint (viii)	18210.2	76.080	5.08

Table 7. Comparison of speed among different methods (test time does not include the serializing time).

The “2-step” Joint (vi) reduces more than half of the running time, compared to “1-step” Joint (v). This decrease in processing time enables us to include more feature set to further improve the performance of SBD in the “2-step” Joint model. By increasing window size from 3 to 5 (i.e., from (vi) to (vii)), (vii) yields 1.3% increase on F-score for SBD, compared to (vi). To further improve the performance, we added NER information with different grams on top of experiment (vii) and found that NER information with unigram (i.e., NER_0) and bigram (i.e., $NER_{-1}+NER_0$) improves the performance, i.e., (viii) shown in Table 6. Undoubtedly, with increased features, the running time of “2-step” Joint model (viii) is more than (vi) and (vii), but it is still faster than the “1-step” Joint model (v). More importantly, it achieved 2% increase on F-score for SBD. Compared to Cascade model (iv), it saved 50% time and achieved 1.6% increase on F-score for SBD.

5 Conclusion

In this paper, we have demonstrated for the first time a word-based labelling approach to Thai SBD. The word-based labelling approach achieved very good performance compared to reported results on ORCHID data. The Cascade model is to use evaluated POS information as features to help SB detection. Higher accuracy in the POS information will yield better accuracy in the Thai SBD. In fact, we also used manually annotated POS tags in SB detection, and it yielded better accuracy, i.e., 79.31% in precision, 62.70% in recall and 70.03% in F-score, compared to the Cascade approach (iv).

Different from Cascade models, Joint models are supposed to make SBD benefit from POS tagging labels in the second layer. Different features are tried in our experiments. Additional features do not always yield better accuracy. For example, when we use more features, e.g., “w_{-1}+w_1” and “wtype_{-1}+wtype_1, on the top of “2-step” Joint (viii), it does not improve the performance. We noticed that the pseudo-POS tagging performance was not improved in the same way as SBD when more features were added. Besides, more experiments will be explored in the future to see how word boundary information, POS and sentence boundary information affect each other.

In this paper, we demonstrated for the first time a word-based labelling approach to Thai SBD. The word-based labelling approach was proposed to leverage LCRF to do sequence labelling which achieved very good performance compared to reported results on ORCHID data. Furthermore, the performance of SBD with the help of POS tagging was investigated on the corpus TaLAPi. Cascade models and Joint models were compared and the “2-step” Joint POS tagging with SB detection was proposed. This proposed model saved more than half of the time, while obtaining almost the same accuracy for SBD as “1-step” Joint model, when using the same feature set. With increased speed, more features were therefore used to improve SBD and yields comparable POS tagging performance.

Reference

AiTi Aw, Sharifah Mahani Aljunied, Nattadaporn Lertcheva and Sasiwimon Kalunsima. 2014. TaLAPi – A Thai Linguistically Annotated Corpus for Language Processing, LREC, 125-132.

Thorsten Brants. 2000. TnT-A Statistical Part-of-Speech Tagger, ANLP, 224-231.

Paisarn Charoen Pornsawat and Virach Sorlertlamvanich. 2001. Automatic Sentence Break Disambiguation for Thai. ICCPOL.

Dan Gillick. 2009. Sentence Boundary Detection and the Problem with the U.S., Proceedings of NAACL HLT, 241-244.

Evang, Kilian and Basile, Valerio and Chrupala, Grzegorz and Bos, Johan. 2013. Elephant: Sequence Labelling for Word and Sentence Segmentation., EMNLP, 1422--1426

J. Laferty, Andrew McCallum and F. C. N. Pereira. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labelling Sequence Data. In ICML Proceedings of the Eighteenth International Conference on Machine Learning.

S. Longchupole. 1995. Thai Syntactical Analysis System by Method of Splitting Sentences from Paragraph form Machine Translation. Master Thesis. King Mongkut’s institute of technology Ladkrabang (in Thai).

Yang Liu, Andreas Stolcke, Elizabeth Shriberg and Mary Harper. 2005. Using Conditional Random Fields for Sentence Boundary Detection in Speech, ACL, 451-458.

Andrew McCallum and Wei Li. 2003. Early Results for Named Entity Recognition with Conditional Random Fields, Feature Induction and Web-Enhanced Lexicons, CONLL.

Pradit Mittrapiyanuruk and Virach Sornlertlamvanich. 2000. The Automatic Thai Sentence Extraction. The Fourth Symposium on Natural Language Processing.

Xian Qian and Yang Liu. 2012. Joint Chinese Word Segmentation, POS tagging and Parsing, ACL.

Glenn Slayden, Mei-Yuh Hwang and Lee Schwartz. 2010. Thai Sentence-Breaking for Large-Scale SMT, WSSANLP, pp. 8-16.

Weiwei Sun and Jia Xu. 2011. Enhancing Chinese Word Segmentation Using Unlabeled Data, ACL.

Virach Sornlertlamvanich, Naoto Takahashi and Hitoshi Isahara. 1997. Building A Thai Part Of-Speech Tagged Corpus (ORCHID).

Charles Sutton, 2006. GRMM: Graphical Models in Mallet. http://mallet.cs.umass.edu/grmm/.

Charles Sutton and Andrew McCallum. 2011. An introduction to conditional random fields, Machine Learning.

Charles Sutton, Khashayar Rohanimanesh and Andrew McCallum. 2004. Dynamic Conditional Random Fields: Factorized Probabilistic Models for Labelling and Segmenting Sequence Data. In International Conference on Machine Learning (ICML).

Katrin Tomanek, Joachim Wermter, and Udo Hahn. 2007. Sentence and token splitting based on conditional random fields. In Proceedings of the 10th Conference of the Pacific Association for Computational Linguistics, pages 49–57, Melbourne, Australia.

Aobo Wang and Min-Yen Kan, 2013. Mining Informal Language from Chinese Microtext: Joint Word Recognition and Segmentation, ACL.

Xuancong Wang, Khe Chai Sim and Hwee Tou Ng. Combining Punctuation and Disfluency Prediction: An Empirical Study, EMNLP 2014, pp.121-130

Xuancong Wang, Hwee Tou Ng, Khe Chai Sim. 2012. Dynamic Conditional Random Fields for Joint Sentence Boundary and Punctuation Prediction. INTERSPEECH 2012, 1384-1387

Suphawut Wathabunditkul. 2003. Spacing in the Thai Language. http://www.thailanguage.com/ref/spacing

Yue Zhang and Stephen Clark. 2008. Joint Word Segmentation and POS Tagging using a Single Perceptron, ACL.