Multi-stage Annotation using Pattern-based and Statistical-based Techniques for Automatic Thai Annotated Corpus Construction

Nattapong Tongtep and Thanaruk Theeramunkong
School of Information, Computer, and Communication Technology,
Sirindhorn International Institute of Technology, Thammasat University, Thailand
131 Moo 5, Tiwanont Rd., Bangkadi, Muang, Pathum Thani, Thailand 12000
{nattapong,thanaruk}@siit.tu.ac.th

Abstract

An automated or semi-automated annotation is a practical solution towards large-scale corpus construction. However, special characteristics of Thai language, such as lack of word-boundary and sentence-boundary markers trigger several issues in automatic corpus annotation. This paper presents a multi-stage annotation framework, containing two stages of chunking and three stages of tagging. Two chunking stages are named entity extraction by pattern matching and word segmentation by dictionary; and three following tagging stages are dictionary-based, pattern-based and statistical-based tagging. Applying heuristics of ambiguity priority, entity extraction is performed first on an original text using a set of patterns, ordered by pattern ambiguity. Later segmenting a sequence of characters into words, the chunks are tagged according to the order of ambiguity, using dictionary, pattern and statistics. Focusing on the reduction of human intervention in corpus construction, our experimental results show that the pattern-based tagging was able to reduce the number of tokens marked as unknown by the dictionary-based tagging by 44.76% and the statistical-based tagging was able to reduce the number of terms identified as ambiguous by both above methods by 72.44%. The proposed multi-stage framework reduced the number of tokens requiring human annotation (those that are tagged unknown or with multiple tags) to 16.35% of the entire corpus.

1 Introduction

As fundamental tasks, word segmentation, part-of-speech (PoS) tagging, and named entity (NE) recognition are essential steps for various natural language processing applications such as text summarization, machine translation, and question answering. For languages like Burmese, Khmer, Lao, Tamil, Telugu, Bali, and Thai, which have no distinct boundary marker between words and sentences (similar to space and a full stop in English), word segmentation is required. PoS tagging is another important task which assigns some syntactic categories such as verb, noun, and preposition to a token or a word for resolving innate ambiguities, while more specific predefined categories, such as person name, location, and organization are assigned in the steps of NE recognition (NER). The current trend in PoS tagging and NE recognition is to utilize machine learning techniques, which are trainable and adjustable. Several supervised learning techniques were successfully attempted and have shown reasonable performances. For PoS tagging, Pandian and Geetha (2009) utilized conditional random fields (CRFs), a probabilistic model, to segment and label sequence data, to tag and chunk PoS in Tamil. Huang et al. (2009) showed that a bigram PoS tagger using latent annotations could achieve the accuracy of 94.78% when testing on a set of the Penn Chinese Treebank 6.0. For NE recognition, Lee et al. (2004) presented a two-level Korean named entity classification (NEC) by cascading highly precise lexical patterns and the decision list. Park and Rim (2008) classified bio-entities by using predicate-argument structures as the external context features. Tongtep and Theeramunkong (2010) investigated a method to segment Thai word and recognize named entity simultaneously by using the concept of character clusters together with discriminative probabilistic models. Such machine learning tasks, however, require high quality tagged corpora or annotated corpora for training which are costly and time consuming to construct. Only few research works studied the methods to build the an-

Proceedings of the 9th Workshop on Asian Language Resources, pages 50–58,
Chiang Mai, Thailand, November 12 and 13, 2011.

notated corpus with less human effort. Lee et al. (2010) proposed rules to judge the tagging reliability for constructing a Korean PoS tagged corpus. Since the quality of the PoS annotation in a corpus is crucial for the development of PoS taggers, Loftsson (2009) examined three error detection methods for automatically detecting hand-correct PoS errors in the corpus. For a corpus size, Sasano et al. (2010) reported that the performance was not saturated even with a corpus size of 100 billion Japanese words when analyzing case frame acquisition for predicate-argument structure. In Thai, Isahara et al. (2000) constructed a PoS tagged corpus named ORCHID manually. The ORCHID corpus was annotated on three levels: paragraph, sentence, and word. Charoenporn et al. (2006) constructed another lexicon by using existing machine-readable dictionaries, and a sort of semantic constraint called selectional preference is added into the lexicon by analyzing Thai texts on the web. Lately, Theeramunkong et al. (2010) proposed a framework and annotation tools for tagging named entity and constructing corpus in Thai. With their annotation tools, the Thai-NEST corpus was annotated and verified by collaborative experts. However, the process is very costly and time consuming. Until now, there have been no research reports on minimizing human intervention in automatic construction of either Thai PoS or named entity tagged corpus.

In this paper, we propose a multi-stage annotation framework to construct a PoS- and NE-tagged corpus with word segmentation for Thai language with less human effort. First, a list of words and named entities is acquired from online resources. Later, they are used in the two succeeding chunking processes to extract named entities and segment words. Three automatic tagging processes are applied together with designed lexical and context features. In the dictionary-based tagging level, ambiguous tokens, unambiguous tokens, and unknown tokens are discovered. In the next step, the number of unknown tokens is reduced by the pattern-based tagging. Finally, the number of ambiguous tokens is decreased in the statistical-based tagging level. The remaining part of this paper is organized as follows. In Sect. 2, the writing system in Thai is discussed. The overall system architecture is proposed in Sect. 3. In Sect. 4, experimental settings and results are reported. The experimental results are discussed in Sect. 5. Finally, a conclusion is illustrated in Sect. 6.

2 Thai Writing System

An example of Thai texts is depicted as shown in Fig. 1. The Thai language consists of 44 consonants, 21 vowel symbols, 4 tone markers for its 5 tonal levels, and a number of punctuation marks. Thai writing system is left-to-right direction, without spaces between words and no uppercase and lowercase characters. Vowels can be written before, after, above, or below consonants, while all tone marks, and diacritics are written above and below the main character.

A Thai word is typically formed by the combination of one or more consonants, one vowel, one tone mark, and one or more final consonants to make one syllable. Thai verbs are not inflected for any of tense, gender, and singular or plural form. Instead, we put some additional words to express their inflection. Moreover, Thai has no distinct boundary maker between words and sentences, like space and a full stop in English.

3 The Framework

In this paper, we propose a multi-stage annotation framework to construct high-quality annotated corpora with less human effort. The framework comprises two stages for chunking and three stages for tagging (see Fig. 2). Two stages for entity chunking are (1) entity extraction and (2) word segmentation. Three stages for entity tagging are (1) dictionary-based tagging level, (2) pattern-based tagging level, and (3) statistical-based tagging level. Entities are named entities, parts-of-speech and other entities such as punctuation and number. A list of entities and a list of words are reusability resources for developing a tagged corpus. In the step of entity extraction, unsegmented tokens and segmented tokens are extracted from the input texts using a set of patterns, ordered by pattern ambiguity, then unsegmented tokens are segmented by the longest matching technique in the step of word segmentation. Segmented tokens are tagged by three-stage entity tagging. Start with the dictionary-based tagging level, ambiguous tokens, unambiguous tokens, and unknown tokens are discovered. The number of unknown tokens is reduced in the pattern-based tagging level. In the statistical-based tagging level, the number of ambiguous tokens is decreased. Instead of check-

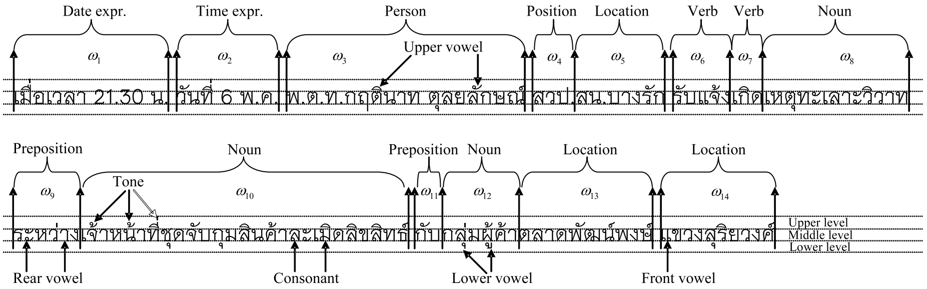


Figure 1: An example of Thai texts

ing all tags in the corpus which is costly and time consuming, our framework can minimize human interposition by indicating which tokens are ambiguous or unknown. Each stage for chunking and tagging is explained in the next section.

3.1 Entity Chunking

For entity chunking, two stages are (1) entity extraction and (2) word segmentation. In this step, a list of words and a list of entities are gathered from online resources such as Thai Wikipedia[1], The Royal Institute[2], The Government Information System[3], Company in Thailand[4], Longdo Dict[5], and YAiTRON[6].

3.1.1 Entity Extraction

In Algorithm 1, we collected the list of entities as entity seeds from online resources. In this step, the list of entities are location (LOC), person name (PER), position (POS), family relationship (FAM), date (DAT), time expressions (TIME) and some parts-of-speech which are longer than two syllables i.e., adverb (ADV), conjunction (CONJ), question phrase (QUE), and verb (VERB). Entity seeds are applied to extract segmented and unsegmented tokens from the input texts. A segmented token is a token which appears in the entity seeds while an unsegmented token is a token which disappears in the entity seeds. Segmented tokens are used to extract left and right contexts, and construct patterns using inner clues and contexts. An inner clue is a set of hint texts which is an apart of named entity. For example, *Her Royal Highness Princess Maha Chakri Sirindhorn* is a person name appears in the person seeds, *Her Royal Highness Princess* will be an inner clue. Generally, one entity may have several entity tags such as "Washington" (person and location). In this paper, constructing patterns using inner clues and contexts will solve the ambiguity in entity tags.

Unsegmented tokens such as named entities outside the list of entity seeds, will be detected and segmented by entity patterns. Segmented tokens or extracted entities from this step will be verified and added to the existing list of entities by experts. This work, the entity extraction is performed before the word segmentation and PoS tagging, since entities in the Thai language are formed by the combination of two or more words, and likely to be transliterated words and unknown words. The remaining unsegmented tokens will be segmented in the word segmentation process.

3.1.2 Word Segmentation

Words are basic components in the language processing. Detecting words in an inherent-vowel alphabetic language that does not have explicit word boundary is highly difficult. To segment words with a good performance and minimizing human interposition for constructing tagged corpus, pattern matching techniques are applied by using a suitable list of words or dictionaries as a tool. It is known that the word segmentation performance will decrease when the processed text contains words that not existing in the dictionary (e.g., unregistered words or unknown words or misspelling words). In order to simply discover unknown word, the longest matching is utilized. Dictionaries or list of words are gathered from online resources, and applied to segment the remaining unsegmented tokens from the prior entity extraction process using the longest matching technique. In this paper, we exploit the longest matching technique implemented by Haruechaiyasak (2006) and

[1] http://th.wikipedia.org

[2] http://www.royin.go.th

[3] http://www.oic.go.th/ginfo

[4] http://www.companyinthailand.com

[5] http://dict.longdo.com

[6] http://th.lug.wikia.com/wiki/YAiTRON_(English-Thai_dictionary)

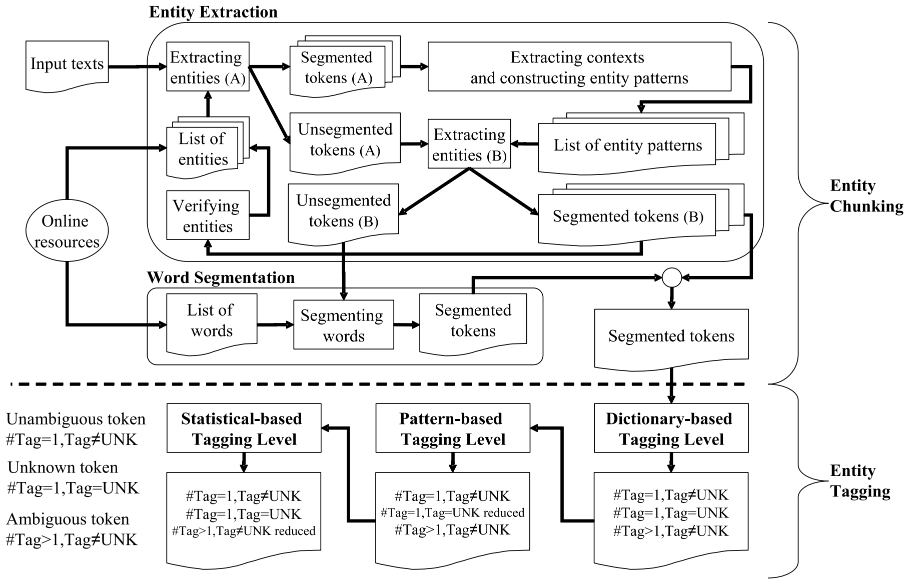

Entity Extraction
Input texts
Extracting entities (A)
Segmented tokens (A)
Extracting contexts and constructing entity patterns
List of entities
Unsegmented tokens (A)
Extracting entities (B)
List of entity patterns
Online resources
Verifying entities
Unsegmented tokens (B)
Segmented tokens (B)
Entity Chunking
Word Segmentation
List of words
Segmenting words
Segmented tokens
Segmented tokens
Unambiguous token #Tag=1,Tag≠UNK
Unknown token #Tag=1,Tag=UNK
Ambiguous token #Tag>1,Tag≠UNK
Statistical-based Tagging Level
Pattern-based Tagging Level
Dictionary-based Tagging Level
#Tag=1,Tag≠UNK #Tag=1,Tag=UNK #Tag>1,Tag≠UNK reduced
#Tag=1,Tag≠UNK #Tag=1,Tag=UNK reduced #Tag>1,Tag≠UNK
#Tag=1,Tag≠UNK #Tag=1,Tag=UNK #Tag>1,Tag≠UNK
Entity Tagging

Figure 2: The framework of multi-stage annotation for automatic Thai annotated corpus construction

use our collected list of words. The output from the entity extraction and the word segmentation, i.e., segmented tokens, will be used as the input data in the dictionary-based tagging level which is one of entity tagging stages.

3.2 Entity Tagging

The entity tagging process consists of 3 tagging levels; (1) dictionary-based tagging level, (2) pattern-based tagging level, and (3) statistical-based tagging level. In this work, 25 entity types i.e., 13 parts-of-speech, 6 named entities, and 6 other entities, are defined for constructing tagged corpus as shown in Table 1. “UNK” is one of entity types which is used to assign tokens that do not belong to 24 predefined entity types.

Figure 3 illustrates an example of the transformation of tags among three tagging levels in the entity tagging process. Three entity tagging stages are described in the next section.

3.2.1 Dictionary-based Tagging Level

In this level, unknown tokens, ambiguous tokens and unambiguous tokens are detected.

An unknown token is a string which does not belong to 24 predefined entity types. This token will be assigned by “UNK” entity tag ($\#Tag = 1, Tag =$ UNK).

An unambiguous token is a string which belongs to one of existing entity types, except “UNK” ($\#Tag = 1, Tag \neq$ UNK).

An ambiguous token is a string which has more than one possible parts-of-speech in the dictionary. ($\#Tag > 1, Tag \neq$ UNK). A set of entity tags assigned to each ambiguous token is called “multi-entity” tag.

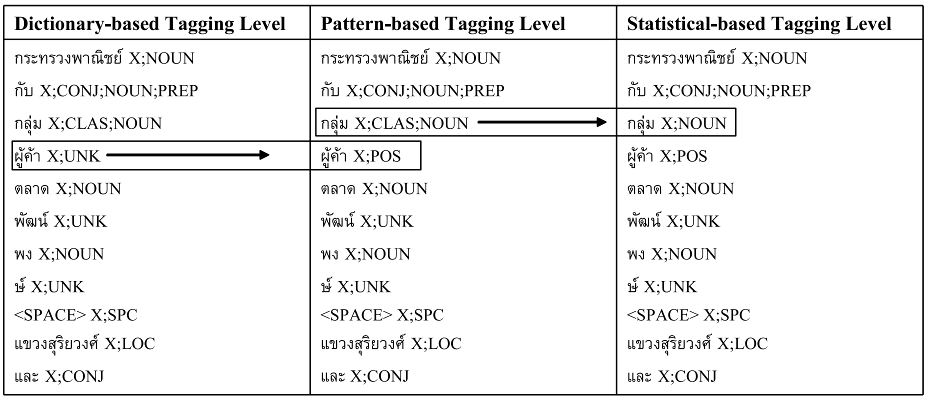
For example, $\underline{w\ \text{X;UNK}}$ means a token w is assigned a single entity tag as unknown (UNK) since the token does not belong to predefined 24 entity types. $\underline{x\ \text{X;NOUN}}$ means a token x is assigned a single entity tag as noun (NOUN) only. $\underline{y\ \text{X;CLAS;NOUN}}$ means a token y is possible to have an entity tag as classifier (CLAS) or noun (NOUN). A set of entity tags i.e., CLAS;NOUN, is a “two-entity” tag. $\underline{z\ \text{X;CONJ;NOUN;PREP}}$ means a token z is possible to have an entity tag as conjunction (CONJ), noun (NOUN), or preposition (PREP). A set of entity tags, i.e., CONJ;NOUN;PREP, is a “three-entity” tag. “X;” is a separator among a token and a set of entity tags, and “;” is a separator among entity tags.

In this paper, the YAiTRON[7] dictionary is exploited to assign the entity tags. YAiTRON: Yet Another (Lex)iTRON is a Thai-English and English-Thai dictionary data, stored in a well-formed XML format. YAiTRON is a homogeneous structure dictionary, adapted from National Electronics and Computer Technology Center (NECTEC[8])’s LEXiTRON[9] dictionary. YAiTRON covers 32,350 unique words

[7] http://th.lug.wikia.com/wiki/YAiTRON_(English-Thai_dictionary)

[8] http://www.nectec.or.th

[9] http://lexitron.nectec.or.th



Dictionary-based Tagging Level	Pattern-based Tagging Level	Statistical-based Tagging Level
กระทรวงพาณิชย์ X;NOUN	กระทรวงพาณิชย์ X;NOUN	กระทรวงพาณิชย์ X;NOUN
กับ X;CONJ;NOUN;PREP	กับ X;CONJ;NOUN;PREP	กับ X;CONJ;NOUN;PREP
กลุ่ม X;CLAS;NOUN	กลุ่ม X;CLAS;NOUN →	กลุ่ม X;NOUN
ผู้ค้า X;UNK →	ผู้ค้า X;POS	ผู้ค้า X;POS
ตลาด X;NOUN	ตลาด X;NOUN	ตลาด X;NOUN
พัฒน์ X;UNK	พัฒน์ X;UNK	พัฒน์ X;UNK
พง X;NOUN	พง X;NOUN	พง X;NOUN
ษ์ X;UNK	ษ์ X;UNK	ษ์ X;UNK
<SPACE> X;SPC	<SPACE> X;SPC	<SPACE> X;SPC
แขวงสุริยวงศ์ X;LOC	แขวงสุริยวงศ์ X;LOC	แขวงสุริยวงศ์ X;LOC
และ X;CONJ	และ X;CONJ	และ X;CONJ

Figure 3: An example of tag transformation in entity tagging

Entity	#Patterns	Example
Adverb	1	อย่าง-
Location	49	กระทรวง-, สถานี-, ชมรม-, ธนาคาร-, อุทยาน-,...
Noun	61	เครื่อง-, หนังสือ-, กระเป๋า-, กล้อง-, โครงการ-,...
Position	11	นัก-, ผู้-, รัฐมนตรี-,...
Verb	3	ตัด-, ร้อง-, ไม่-

Figure 4: An example of Thai grammatical patterns

with 13 parts-of-speech i.e., adjective (ADJ), adverb (ADV), auxiliary verb (AUX), classifier (CLAS), conjunction (CONJ), determiner (DET), end (END), interjection (INT), noun (NOUN), preposition (PREP), pronoun (PRON), question phrase (QUE), and verb (VERB).

3.2.2 Pattern-based Tagging Level

There are some tokens that always have only one PoS when beginning with some specific texts. For example, every token begins with "Ministry of" always be a location, or every token begins with "Minister of" always be a person's position. We assemble such texts by observing prefix's tokens from the dictionary. So far we have had 125 patterns with 100% correctness; 1 pattern for adverb, 49 patterns for locations, 61 patterns for nouns, 11 patterns for positions and 3 patterns for verbs. An example of Thai grammatical patterns is shown in Fig. 4. Furthermore, other tokens such as comment, number, punctuation, space, and English characters, will be automatically assigned with an entity tag as COMMENT, NUM, PUNC, SPC and ENG, respectively. Every unknown token which does not match with these patterns in this tagging level will be assigned with an entity tag as UNK.

3.2.3 Statistical-based Tagging Level

In this level, only ambiguous tokens will be transformed to unambiguous tokens. Since an ambiguous token comprises more than one possible parts-of-speech which specified in the dictionary, we need a PoS classifier to select the best PoS tag among them. In machine learning tasks, several PoS classifiers were trained from the large PoS tagged corpora which are costly and time consuming to construct. In this work, we exploit naïve Bayes classifier since it only requires a small amount of training data to estimate the parameters necessary for classification. A naïve Bayes classifier is a simple probabilistic classifier based on applying Bayes' theorem with strong (naive) independence assumptions. In spite of their naive design and apparently over-simplified assumptions, naïve Bayes classifier has worked quite well in many complex real-world situations. In this paper, nine context features are defined as shown in Table 2.

Predicting an entity tag t given a vector of context features $F = (f_1, f_2, ..., f_{|F|})$. One simple way to accomplish this is to assume that once the entity tag is known, all the features are independent. The result is based on a joint probability model of the form:

$$p(t|F) = p(t_j) \prod_{i=1}^{|F|} p(f_i|t_j). \quad (1)$$

The best entity tag t_{best} among the output tags T is

```
Algorithm 1: Entity extraction
  Input  : texts
  Output: segmented tokens $sTB$ and
          unsegmented tokens $uTB$
  Extracting entities (A):
    - Collect list of entities $e$ from online
    resources, ordered by longest matching
    - Label $e$ in the input texts
      → segmented tokens (A) $sTA$
      → unsegmented tokens (A) $uTA$
  Extracting contexts and constructing entity
  patterns:
    - Extract contexts surround $sTA$
      → 20 characters from $sTA$'s left $cL20$
      → 20 characters from $sTA$'s right $cR20$
    - Collect inner clue entities from $e$
      → list of inner clues $iClue$
    - Construct patterns for $e$
      → pattern $p$ = $\{cL20\}\{uTA \in iClue$
      and $uTA \ni cL20$, $cR20\}\{cR20\}$
  Extracting entities (B):
    - Label $p$ in $uTA$
      → segmented tokens (B) $sTB$
      → unsegmented tokens (B) $uTB$
  Verifying entities:
    - Verify $sTB$
    - Add the correct $sTB$ to $e$
```

$$t_{best} = \arg\max_{t_j \in T} p(t_j) \prod_{i=1}^{|F|} p(f_i|t_j). \qquad (2)$$

We train our statistical-based entity tagger by using the traditional naïve Bayes classifier. Since an ambiguous token will be transformed to an unambiguous token if the best single entity tag obtained from its multi-entity tag, we modify Equation 2 to support our constraint. The best entity tag t'_{best} among a multi-entity tag $T' = (t'_1, t'_2, ..., t'_n)$ is

$$t'_{best} = \arg\max_{t'_j \in T'} p(t'_j) \prod_{i=1}^{|F|} p(f_i|t'_j), \qquad (3)$$

$$t'_{best} = \begin{cases} t'_{best} & \text{if } t'_{best} \in T' \\ T' & \text{otherwise} \end{cases}$$

Type	Entity	Description
PoS	ADJ	Adjective
	ADV	Adverb
	AUX	Auxiliary verb
	CLAS	Classifier
	CONJ	Conjunction
	DET	Determiner
	END	End
	INT	Interjection
	NOUN	Noun
	PREP	Preposition
	PRON	Pronoun
	QUE	Question phrase
	VERB	Verb
NE	DAT	Date expr.
	FAM	Family rlat.
	LOC	Location
	PER	Person
	POS	Position
	TIM	Time expr.
Other	COMMENT	Comment
	ENG	English
	NUM	Number
	PUNC	Punctuation
	SPC	Space
	UNK	Unknown

Table 1: The list of possible entity tags

4 Experimental Settings and Results

We collected 764 Thai news documents comprised 1,559,330 characters from the web. In the step of entity extraction, we acquired 19,528 segmented tokens as shown in Table 3. In the step of word segmentation, a list of 155,088 unique words acquired from online resources were applied to segment unsegmented tokens. Using the longest matching technique, we obtained 316,653 segmented tokens. By entity chunking i.e., 336,181 tokens were used as the input data for the entity tagging process.

Table 4 shows the experimental results from the entity tagging process. Unambiguous tokens, unknown tokens, and ambiguous tokens were classified in the dictionary-based tagging level, while the pattern-based tagging level and the statistical-based tagging level reduced the number of unknown tokens and the number of ambiguous tokens, respectively. In the dictionary-based tagging level, 24.14% and 10.94% of all token texts were tagged as unknown tokens and ambiguous tokens. The number of unknown tokens reduction in the pattern-based tagging level is 44.76% (reduced from 81,170 unknown tokens in the dictionary-based tagging level to 44,841 unknown tokens in the pattern-based tagging level). The number of

Token Type	Dictionary-based	Pattern-based	Statistical-based
$\#Tag = 1, Tag \neq$ UNK (Unambiguous tokens)	218,237 (64.92%)	254,566 (75.72%)	281,205 (83.65%)
$\#Tag = 1, Tag =$ UNK (Unknown tokens)	81,170 (24.14%)	44,841 (13.34%)	44,841 (13.34%)
$\#Tag > 1, Tag \neq$ UNK (Ambiguous tokens)	36,774 (10.94%)	36,774 (10.94%)	10,135 (3.01%)
Total tokens	336,181 (100.00%)	336,181 (100.00%)	336,181 (100.00%)

Table 4: The experimental results of the entity tagging

Feature	Definition
tagL2	The second left entity tag
tagL1	The first left entity tag
tagR1	The first right entity tag
tagR2	The second right entity tag
tagL2L1	Two entity tags from left
tagR1R2	Two entity tags from right
tagL2L1R1	Two entity tags from left and one entity tag from right
tagL1R1R2	One entity tag from left and two entity tags from right
tagL2L1R1R2	Two entity tags from left and right

Table 2: Features for the statistical-based tagging level in the entity tagging process

Entity	#Tokens
Adverb	1,342
Conjunction	658
Date	1,386
Family relationship	99
Location	3,781
Person	5,010
Position	467
Question phrase	6
Time	2,571
Verb	4,208
TOTAL	19,528

Table 3: The statistical results of the entity extraction

UnK → UnA	#Tokens
UNK → UNK	44,841
UNK → COMMENT	14,331
UNK → NUM	6,867
UNK → PUNC	4,350
UNK → NOUN	3,386
UNK → POS	2,738
UNK → VERB	1,947
UNK → SPC	1,058
UNK → LOC	956
UNK → ADV	447
UNK → ENG	249
TOTAL	81,170

Table 5: The statistical results of the tag transformation from unknown tokens in the dictionary-based tagging level to unambiguous tokens in the pattern-based tagging level (UnK → UnA)

unknown tokens in the dictionary-based tagging level transformed to unambiguous tokens in the pattern-based tagging level is described in Table 5.

Moreover, in the pattern-based tagging level, 13.34% of all tokens were tagged as unknown tokens. The number of ambiguous tokens reduction in the statistical-based tagging level is 72.44% (reduced from 36,774 ambiguous tokens in the pattern-based tagging level to 10,135 ambiguous tokens in the statistical-based tagging level). In the statistical-based tagging level, 3.01% of all token texts were tagged as ambiguous tokens and 13.34% of all token texts were tagged as unknown tokens. Our entity tagging process can increase the number of unambiguous tokens up to 83.65%. In this experiment, there were 34 multi-entity tags; 20 two-entity tags, 12 three-entity tags; 1 four-entity tag and, 1 five-entity tag. There are some tokens or words that can be classified as classifier (CLAS), noun (NOUN), preposition (PREP), or pronoun (PRON) depending on contexts. The maximum number of possible tags for a token is set to 5 that is CLAS;CONJ;NOUN;PREP;VERB. NOUN;VERB is a two-entity tag which is highly occurred in the pattern-based tagging level as an ambiguous token, followed by CLAS;NOUN. All ambiguous tokens with their two-entity tag i.e., INT;NOUN can be transformed to unambiguous tokens. Among three-entity tags, ADJ;ADV;AUX is highly occurred in the pattern-based tagging level, followed by NOUN;PREP;VERB. More than 80% of tokens with one of the following 11 multi-entity tags can be transformed to unambiguous tokens by using the proposed context features together with the additional constraint of the joint probability model in the statistical-based tagging level.

- Two-entity tag: INT;NOUN, CLAS;VERB,

PREP;VERB, CONJ;VERB, NOUN;VERB

- Three-entity tag: CLAS;INT;NOUN, NOUN;PREP;VERB, CLAS;NOUN;VERB, CLAS;NOUN;PRON, DET;NOUN;VERB
- Five-entity tag: CLAS;CONJ;NOUN;PREP;VERB

5 Discussion

In this section, the experimental results are discussed. The accuracy of each process was independently verified. This applies to each of the steps including the dictionary-based tagging.

In the entity extraction, we can successfully tag parts-of-speech for tokens longer than two syllables (i.e., adverb, conjunction, question phrase and verb) and named entities (i.e., date, family relationship, location, position and time) with 100% correctness. For tagging person name, we achieved up to 91.95% correctness. Due to the fact that Thai language has no word boundary, extracting entities may not be straightforward. For example, a string whose spelling is equivalent to a short verbal word in a dictionary may not be such a verbal word but just a part of a longer string which indicates another word. From this point of view, it seems better to focus on only a longer verb phrase. Then one potential constraint is to handle a verb phrase that is longer than two syllables. This constraint also handles adverb, conjunction and question phrase that are longer than two syllables. Furthermore, named entities i.e., date, family relationship, location, position and time have explicit boundary, except person name.

In the word segmentation process, the correctness decrease when the processed text contains words that do not exist in the dictionary or list of words. Longest matching algorithm can be considered as using some heuristics to solve the ambiguity problem by selecting the longest possible term. From the experimental results, we obtained 13.34% unknown tokens.

In the dictionary-based tagging level, the performance depends on the reliability of the dictionary. In this work, a token was assigned with possible parts-of-speech defined in well-known dictionaries. From this assumption, all unambiguous tokens, unknown tokens and ambiguous tokens were correctly classified.

In the pattern-based tagging level, our patterns can transform 36,329 unknown tokens to be unambiguous tokens with 100% correctness (from 81,170 unknown tokens in the dictionary-based tagging level to 44,841 unknown tokens in the pattern-based tagging level). Among 44,841 unknown tokens from this level could be unknown words or misspelling words or unregistered word in the dictionary. To solve these problems, human effort is required.

In the statistical-based tagging level, 26,639 ambiguous tokens were transformed to be unambiguous tokens in this level (from 36,774 ambiguous tokens in the pattern-based tagging level to 10,135 ambiguous tokens in the statistical-based tagging level). The accuracy of this tagger is relatively high since it selects the best entity tag for ambiguous tokens from theirs possible tags. If the best entity tag can not obtain from its possible tags, ambiguous tokens will not be transformed to unambiguous tokens.

We conclude that the dictionary-based tagging level and the pattern-based tagging level achieved 100% correctness. Based on statistics, the statistical-based tagging level can help to reduce unambiguous tokens. The performance of the entity chunking stage, especially the word segmentation process, affects the overall performance of the entity tagging stage. Anyway, our multi-stage annotation framework helps to minimize the manual effort in constructing a Thai entity annotated corpus. The expert can focus on selecting the correct PoS from all possible parts-of-speech provided by the dictionary for ambiguous tokens and correct unknown tokens, that is only 16.35% to complete the annotated corpus construction (3.01% from ambiguous tokens and 13.34% from unknown tokens). However, there might be errors even in unambiguous tokens since the correctness of entity extraction, the correctness of word segmentation, and the accuracy of the statistical-based tagger are not 100%. In order to construct an annotated corpus where all annotations are correct, human experts should check not only unknown and ambiguous tokens but also unambiguous tokens. Since the accuracies of the automatically determined word segments and tags are high enough, the proposed system would alleviate human burden even when experts should check all tokens.

6 Conclusion and Future Work

This paper has presented a multi-stage annotation framework to minimize the manual effort in constructing a Thai entity annotated corpus. We propose a new tagging strategy that can automatically detect and reduce unknown tokens and ambiguous tokens. Even a small decrease in the amount of manual annotation task can achieve significant cost savings in constructing a large-scale entity annotated corpus. The proposed framework can provide a new and convenient way to construct annotated corpora, control the quality of the corpus, and reduce the amount of manual annotation. As future work, we plan to create new rules to detect more new content entities among various domains. The measurement of the tagger's reliability and developing an annotation verification system are also investigated.

Acknowledgments

This work has been supported by the Thailand Research Fund (TRF), under Grant No. BRG50800013, a TRF Royal Golden Jubilee Ph.D. Program, under Grant No. PHD/0034/2550, the National Electronics and Computer Technology Center, Thailand, under Grant No. NT-B-22-KE-38-54-01, Government Research Fund via Thammasat University, Thailand, and the National Research University Project of Thailand Office of Higher Education Commission.

References

Thatsanee Charoenporn, Canasai Kruengkrai, Thanaruk Theeramunkong, and Virach Sornlertlamvanich. 2006. Construction of thai lexicon from existing dictionaries and texts on the web. *IEICE - Trans. Inf. Syst.*, E89-D:2286–2293, July.

Choochart Haruechaiyasak. 2006. Longlexto: Tokenizing thai texts using longest matching approach.

Zhongqiang Huang, Vladimir Eidelman, and Mary Harper. 2009. Improving a simple bigram hmm part-of-speech tagger by latent annotation and self-training. In *NAACL '09: Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, pages 213–216, Morristown, NJ, USA. Association for Computational Linguistics.

Hitoshi Isahara, Qing Ma, Virach Sornlertlamvanich, and Naoto Takahashi. 2000. Orchid: building linguistic resources in thai. *Lit. Linguist Computing*, 15(4):465–478.

Seungwoo Lee, Joohui An, Byung-Kwan Kwak, and Gary Geunbae Lee. 2004. Learning korean named entity by bootstrapping with web resources. *IEICE - Trans. Inf. Syst.*, 87(12):2872–2882, December.

D.-G. Lee, G. Hong, S. K. Lee, and H.-C. Rim. 2010. Minimizing Human Intervention for Constructing Korean Part-of-Speech Tagged Corpus. *IEICE - Trans. Inf. Syst.*, 93:2336–2338.

Hrafn Loftsson. 2009. Correcting a pos-tagged corpus using three complementary methods. In *EACL '09: Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 523–531, Morristown, NJ, USA. Association for Computational Linguistics.

S. Lakshmana Pandian and T. V. Geetha. 2009. Crf models for tamil part of speech tagging and chunking. In *ICCPOL '09: Proceedings of the 22nd International Conference on Computer Processing of Oriental Languages. Language Technology for the Knowledge-based Economy*, pages 11–22, Berlin, Heidelberg. Springer-Verlag.

Kyung-Mi Park and Hae-Chang Rim. 2008. Semantic classification of bio-entities incorporating predicate-argument features. *IEICE - Trans. Inf. Syst.*, E91-D(4):1211–1214.

R. Sasano, D. Kawahara, and S. Kurohashi. 2010. The Effect of Corpus Size on Case Frame Acquisition for Predicate-Argument Structure Analysis. *IEICE - Trans. Inf. Syst.*, 93:1361–1368.

Thanaruk Theeramunkong, Monthika Boriboon, Choochart Haruechaiyasak, Nichnan Kittiphattanabawon, Krit Kosawat, Chutamanee Onsuwan, Issariyapol Siriwat, Thawatchai Suwanapong, and Nattapong Tongtep. 2010. THAI-NEST: A Framework for Thai Named Entity Tagging Specification and Tools. In *CILC '10: Proceedings of the 2nd Int'l Conference on Corpus Linguistics (CILC10), May 13-15, 2010*, pages 895–908, May.

Nattapong Tongtep and Thanaruk Theeramunkong. 2010. Simultaneous character-cluster-based word segmentation and named entity recognition in thai language. In *Proceedings of the Fifth International Conference on Knowledge, Information and Creativity Support Systems (KICSS 2010), November 25-27, 2010*, pages 167–175.