Thai Word Segmentation Web Service

Seksan Poltree
Department of Computer Engineeering
Faculty of Engineering
Khon Kaen Univerity
Muang, Khon Kaen 40000
Email: seksan.poltree@gmail.com

Kanda Saikaew
Department of Computer Engineeering
Faculty of Engineering
Khon Kaen Univerity
Muang, Khon Kaen 40000
Email: krunapon@kku.ac.th

***Abstract*—Word segmentation is the first process for natural language processing especially in Thai language which is written without words or sentences delimiters. Although there are existing Thai word segmentation software, most of them are desktop applications which users need to take a considerable amount of time in studying and installing the programs. In this paper, we propose the first public web service for Thai word segmentation with the part of speech tagging. This web service is a REST service based on available free Thai word segmentation programs. The currently supported transport protocol is HTTP and output data format is JSON. We also have compared the performance of existing Thai word segmentation programs using NECTEC BEST corpus in the perspective of precision, recall and f-score.**

***Index Terms*—natural language processing, word segmentation, http web services**

I. Introduction

Web application is widely used as a common platform for accessing software nowadays. Many web application services are mashing up from HTTP web services. Modern web applications are now exploded with astonishing growing amount of data, thus there is the need for filtering out meaningful data or extracting semantic data.

Using natural language processing technique is an optional. When dealing with Thai documents, word segmentation is the first task for Thai text processing. Thai documents have no boundaries or delimiters between the multiple words and also between sentences. Therefore, in order to extract meaningful words from Thai documents, we need word segmentation program.

Several segmentation algorithms have been proposed to apply word segmentation in Thai documents [18]. Previously almost all new researchers interested to tackle Thai semantic data problems need to develop the programs for words and sentences segmentation. It is a problem especially for new people who need to study the algorithms in processing such tasks as well as implementing the programs. Instead of focusing on solving their core problem, they need to spend an excessive amount of time in learning how to segment words and sentences.

Instead of reinventing the wheels by reimplementing the programs for words segmentation, in this paper, we propose and implement the web service for words segmentation. This work proposes a Thai words segmentation service to to support several semantic web applications developed to analyse and extract web information. The user can use this service to reduce their learning time to segment Thai documents and thus can devote more time in focusing on solving the real semantic problems. Although there is one public web service available for Thai words segmentation [9], our web service will be the first web service for Thai word segmentation that supports part-of-speech tagging based on existing Thai corpus.

There are two major categories of web service implementation. These are Simple Object Access Protocol (SOAP) [13] and REpresentational State Transfer (REST) [14] and representational state transfer (REST). SOAP has some standard supports and more related extensible standards with strict rules. SOAP has an advantage for service with complex structures and multiple interaction but it also requires more complicated implementation Conversely, REST is simpler because it uses justs existing HTTP and user-defined data format. However, this simplicity causes the the lack of standard for the transferred data.

In this paper, we choose to implement a REST web service. The REST web service does not need a complex schema to describe data; moreover, REST is usually more efficient than SOAP in the term of bandwidth utilization and lower round-trip latency when transmission over the network [24] . This advantage increases the service scalability and reduces the service processing time.

For the web service output language, there are two widely used major formats, eXtensible Markup Language (XML) [12] and JavaScript Object Notation (JSON) [15]. XML is a standard suitable for describing complex documents which need much descriptive information while JSON is suitable for transferring data with a simple structured documents. Word segment web service returns a simple data output which just gives the boundaries and tagging of words. T he proposed REST web service uses JSON over XML because JSON gives a smaller size of data. However, one can also translate date between JSON and XML by creating a JSON/XML translator. [25].

II. Backgrounds and Related Works

There is no word segmentation in English because English text has spaces as tokens between sentences. In contrast, there are limited online Thai word segmentation services. TLex [7] uses conditional random field to train the model for segmentation. NECTEC researchers have developed web application [8] and TLexs [9], a SOAP web service for evaluation. However, TLexs is not fully publicly available since it requires user registration before invoking the service. There are two main differences between TLexs and our service. The first difference is that TLexs is a SOAP web service while our proposed one is a REST web service. The other difference is that TLexs does not have part-of-speech tagging while our proposed service does.

Part-of-speech (POS) tagging is an another important component. In English language, there are some of POS tagging web applications. For instance, English Standford POS tagger [10] is a part of speech tagger application. It has multiple extensions and many derived services. It is available through a GUI application, Ruby and Python language binding, XML-RPC service interface and an online web application. Another one is Illinois Part of Speech Tagger [11] which is one of POS tagger and can be downloaded to use. Conversely, Thai POS tagger is limited availability. There is only NECTEC Orchid [5] but it's not public available as a web service.

III. Available Language Software and Resources

There are some of Thai word segmentation software which have various technical methods and licences. In our work, we choose to use only the software that has open source licenses because we are free to use and modify software.

A. Libthai

Libthai [1] is a set of open source libraries for Thai language support. It consists of character support, character properties, string manipulators, string collation, input/output method and word breaking. Libthai word breaking feature implements maximal matching algorithm. It contains 23,563 words from Thai common dictionary [4] and words are manually added by maintainer. Libthai dictionary selects the minimal of words for speed optimization in usage. Our proposed service includes word breaking feature and dictionary based on this Libthai.

B. Smart Word Analysis for THai : SWATH

SWATH [2] was originally developed in 2003. It proposed word segmentation using longest matching and maximal matching algorithms. It also implemented bigram part of speech tagging using Orchid corpora resource. The maintained version [2] has removed the corpus tagging and feature based algorithm features. This version contains 23,944 words for internal dictionary. These words have extract from Thai common dictionary and manually added by maintainer. Our proposed service uses the maintained version because it has fixed some problem issues and ready to use.

C. NECTEC Orchid

Orchid [5] is a part-of-speech (POS) tagged corpus in Thai. It is available for free to use in multiple text formats. It contains some of separated paragraphs, sentences and tagged words. The disadvantage is the corpus document is a technical document so it is a specific domain. It has a problem for its ambiguity in POS assignment. There is only public access tagged corpus available, then we will try to use this resource first. We propose to use this corpus as a train corpora for tagging service.

D. NECTEC BEST corpora

BEST corpus [6] is a free word corpus with segmented words. It has been created as a benchmark tool to use for Thai segmentation program. BEST 2009 corpus has three different segmented documents. There are articles, encyclopedia, news and novels. We have counted the BEST corpus words in the provided documents. It contains about 32,700 non-duplicated common words, 34,100 non-duplicated proper nouns and specific words. We propose to use this corpus for testing as well as serving as an alternative word dictionary resource for segmentation.

IV. System Development

A. System Overview

Figure 1 shows the overview of the system which consists of two main components: a web application and a HTTP JSON web service. We have developed the system using Python programming language [19]. In Python, there is a Natural Language Tool Kit (NLTK) [20], a python module for linguistic data and development in natural language processing and text analytics. We use this tool kit's N-gram tagging feature for part-of-speech tagging with Orchid corpus. For word segmentation, we have developed a python wrapper program to connect programs running in background.

The system web interface is generated by Django web framework [21] in the Python language. The framework connects the MySQL database over internal object-relational mapping system. We create a Django application to process JSON requests from web. This application will use the python modules in the background and return the JSON response to the web request. We have deployed the service application on Lighttpd [22] server using fastcgi interface.

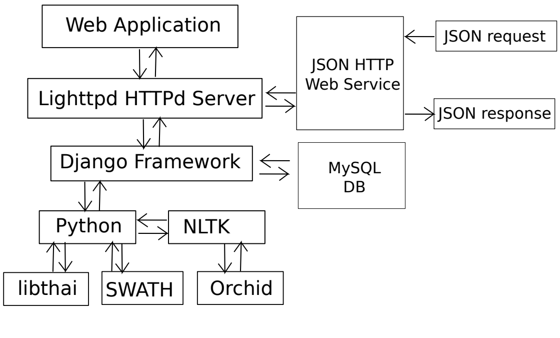
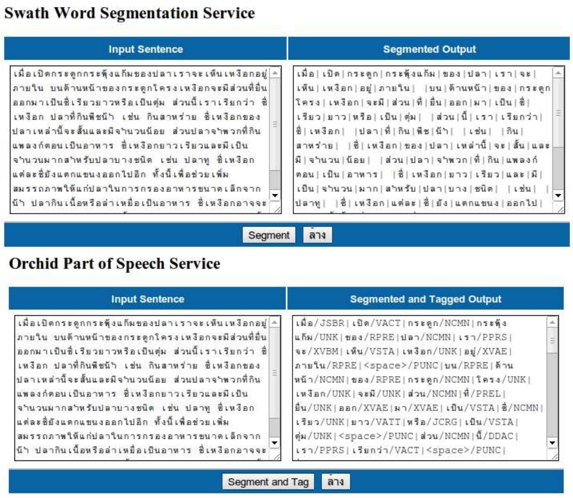


Fig. 1. System Overview

B. Web Application

Our web application is developed to demonstrate and test the invocation of web services or APIs. From our experiments, the web application and the JSON web service give the same result. This web interface does not require any user to register before using. Web application is available at http://thaisemantics.org/service/swath/index for SWATH word segmentation and http://thaisemantics.org/service/orchid/index for orchid part of speech tagging.



Swath Word Segmentation Service

```
เมื่อเปิดกระดูกกระพุ้งแก้มของปลาเราจะเห็นเหงือกอยู่
ภายใน บนด้านหน้าของกระดูกโครงเหงือกจะมีส่วนที่ยื่น
ออกมาเป็นซี่เรียวยาวหรือเป็นตุ่ม ส่วนนี้เราเรียกว่า ซี่
เหงือก ปลาที่กินพืชน้ำ เช่น กินสาหร่าย ซี่เหงือกของ
ปลาเหล่านี้จะสั้นและมีจำนวนน้อย ส่วนปลาจำพวกที่กิน
แพลงก์ตอนเป็นอาหาร ซี่เหงือกยาวเรียวและมีเป็น
จำนวนมากสำหรับปลาบางชนิด เช่น ปลาทู ซี่เหงือก
แต่ละซี่ยังแตกแขนงออกไปอีก ทั้งนี้เพื่อช่วยเพิ่ม
สมรรถภาพให้แก่ปลาในการกรองอาหารขนาดเล็กจาก
น้ำ ปลากินเนื้อหรือล่าเหยื่อเป็นอาหาร ซี่เหงือกอาจจะ
```

```
เมื่อ|เปิด|กระดูก|กระพุ้งแก้ม|ของ|ปลา|เรา|จะ|
เห็น|เหงือก|อยู่|ภายใน| |บน|ด้านหน้า|ของ|กระดูก
โครง|เหงือก|จะมี|ส่วน|ที่|ยื่น|ออก|มา|เป็น|ซี่|
เรียว|ยาว|หรือ|เป็น|ตุ่ม| |ส่วน|นี้|เรา|เรียกว่า|
ซี่|เหงือก| |ปลา|ที่|กิน|พืช|น้ำ| |เช่น| |กิน|
สาหร่าย| |ซี่|เหงือก|ของ|ปลา|เหล่านี้|จะ|สั้น|และ
มี|จำนวน|น้อย| |ส่วน|ปลา|จำพวก|ที่|กิน|แพลงก์
ตอน|เป็น|อาหาร| |ซี่|เหงือก|ยาว|เรียว|และ|มี|
เป็น|จำนวน|มาก|สำหรับ|ปลา|บาง|ชนิด| |เช่น| |
ปลาทู| |ซี่|เหงือก|แต่ละ|ซี่|ยัง|แตกแขนง|ออกไป|
```

Orchid Part of Speech Service

```
เมื่อเปิดกระดูกกระพุ้งแก้มของปลาเราจะเห็นเหงือกอยู่
ภายใน บนด้านหน้าของกระดูกโครงเหงือกจะมีส่วนที่ยื่น
ออกมาเป็นซี่เรียวยาวหรือเป็นตุ่ม ส่วนนี้เราเรียกว่า ซี่
เหงือก ปลาที่กินพืชน้ำ เช่น กินสาหร่าย ซี่เหงือกของ
ปลาเหล่านี้จะสั้นและมีจำนวนน้อย ส่วนปลาจำพวกที่กิน
แพลงก์ตอนเป็นอาหาร ซี่เหงือกยาวเรียวและมีเป็น
จำนวนมากสำหรับปลาบางชนิด เช่น ปลาทู ซี่เหงือก
แต่ละซี่ยังแตกแขนงออกไปอีก ทั้งนี้เพื่อช่วยเพิ่ม
สมรรถภาพให้แก่ปลาในการกรองอาหารขนาดเล็กจาก
น้ำ ปลากินเนื้อหรือล่าเหยื่อเป็นอาหาร ซี่เหงือกอาจจะ
```

```
เมื่อ/JSBR|เปิด/VACT|กระดูก/NCMN|กระพุ้ง
แก้ม/UNK|ของ/RPRE|ปลา/NCMN|เรา/PPRS|
จะ/XVBM|เห็น/VSTA|เหงือก/UNK|อยู่/XVAE|
ภายใน/RPRE|<space>/PUNC|บน/RPRE|ด้าน
หน้า/NCMN|ของ/RPRE|กระดูก/NCMN|โครง/UNK|
เหงือก/UNK|จะมี/UNK|ส่วน/NCMN|ที่/PREL|
ยื่น/UNK|ออก/XVAE|มา/XVAE|เป็น/VSTA|ซี่/NCMN|
เรียว/UNK|ยาว/VATT|หรือ/JCRG|เป็น/VSTA|
ตุ่ม/UNK|<space>/PUNC|ส่วน/NCMN|นี้/DDAC|
เรา/PPRS|เรียกว่า/VACT|<space>/PUNC|
```

Fig. 2. Web Application Interface Example

Figure 2 is the example of these web interfaces. The usage instruction for this web application:

1) Browse the web browser to application page.
2) On the left box ”Input Sentence”, enter the input sentences.
3) Press ”Segmented” button
4) The result will be shown on right box

The main implemented features are:

1) Registration and User Account: We have implemented registration and email validation module. A user requires to register at the fist time. The system will send email validation then the user can log on to the system. After logged on, the user can create or recreate an API key for using with JSON Service.

2) API-Key Hashing: This key is a required information to call the JSON service. The registered users can recreate their API key by click rehash. The key is randomly generated by a 64 characters hashing algorithm.

C. HTTP JSON Web Service

Since the input of the service is a text thus the HTTP POST is more suitable to use than HTTP GET. We have defined the input and output of functions of our proposed service. JSON web service endpoint is http://thaisemantics.org/service/rest/.

Table I shows the available methods and examples for inputs and outputs formats. We have also defined the outputs for invalid input and invalid api-key. First, the service will check if the input is a JSON encoded message. Then it will check registered api-key. If the key is valid, it will check the method and parameters. An input message contains an api-key, a method, and list of parameters. An output message contains status and a result list.

TABLE I
JSON HTTP Web Service Method and Example

Method	SWATH
Example Input	{’api_key’: ’YOUR API KEY’, ’method’: ’SWATH’, ’params’: [’unicode strings’], }
Example Output	{”status”: 0, ”result”: [’list’,’of’, ’segmented’, ’words’], }
Method	ORCHID
Example Input	{’api_key’: ’YOUR API KEY’, ’method’: ’ORCHID’, ’params’: [’unicode strings’], }
Example Output	{”status”: 0, ”result”: [[”list”,”POS”], [”tagged”,”POS”]], }
Wrong API-Key Output	{”status”: 1, ”result”: [”Wrong API key.”] }
Wrong Input Format Output	{”status”: -1, ”result”: [”Unkown request”] }

D. JSON Web Service Client Example

We have created an example to simply call the JSON service. In Python version 2.6, we use its standard API to create a JSON document according to the formats. Figure 3 and Figure 4 show results of examples in calling the SWATH and ORCHID methods in the proposed web service respectively.

JSON web service call instruction:

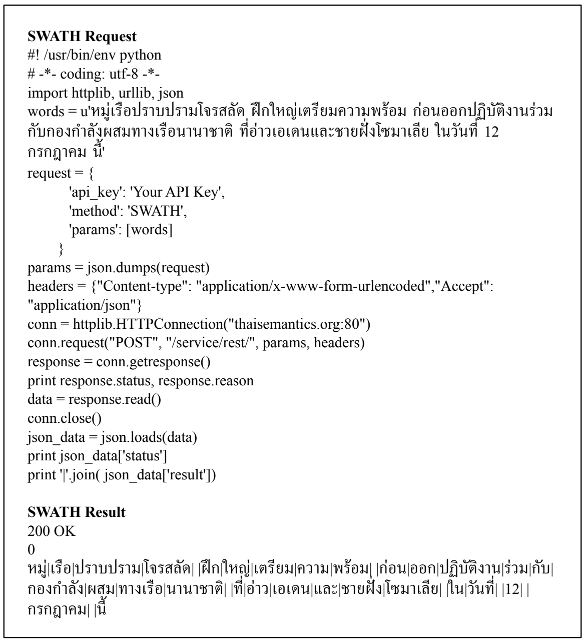
1) Register with email validation. After validating through the email and logging on the web, on the profile page, the user can then copy the API-KEY to use for creating a message to call the service.
2) Create a JSON encoded message according to its method as described in table I.
3) Create a HTTP or a HTTPS connection and then post the JSON encoded message to the service endpoint.
4) Retrieve the result.

V. Experiments and Results

In the experiment, we propose to test the segmentation service for our REST JSON web service. The evaluation result are compared between our service and the existing TLexs service. There are some differences between our service and TLexs that might not be directly compared against each other such as some of them are offline applications. However, the objective of this experiment is to show that this service performs functions as expected.

We tested the word segmentation system using BEST corpus. The BEST corpus is the only one collection of Thai word segmentation solution corpus which is available for public usage.

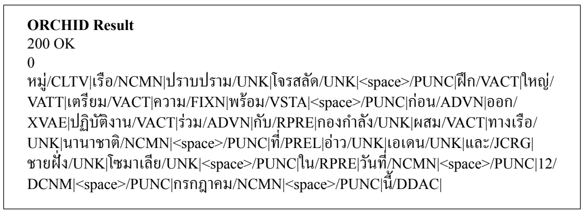
First, we prepared a testing data by using BEST corpus multiple files for each document resource. We have generated the testing data from BEST training data by removing segmented marks in each file. The segmentation results are the segmented



```
SWATH Request
#! /usr/bin/env python
# -*- coding: utf-8 -*-
import httplib, urllib, json
words = u'หมู่เรือปราบปรามโจรสลัด ฝึกใหญ่เตรียมความพร้อม ก่อนออกปฏิบัติงานร่วม
กับกองกำลังผสมทางเรือนานาชาติ ที่อ่าวเอเดนและชายฝั่งโซมาเลีย ในวันที่ 12
กรกฎาคม นี้'
request = {
         'api_key': 'Your API Key',
         'method': 'SWATH',
         'params': [words]
       }
params = json.dumps(request)
headers = {"Content-type": "application/x-www-form-urlencoded","Accept":
"application/json"}
conn = httplib.HTTPConnection("thaisemantics.org:80")
conn.request("POST", "/service/rest/", params, headers)
response = conn.getresponse()
print response.status, response.reason
data = response.read()
conn.close()
json_data = json.loads(data)
print json_data['status']
print '|'.join( json_data['result'])

SWATH Result
200 OK
0
หมู่|เรือ|ปราบปราม|โจรสลัด| |ฝึก|ใหญ่|เตรียม|ความ|พร้อม| |ก่อน|ออก|ปฏิบัติงาน|ร่วม|กับ|
กองกำลัง|ผสม|ทางเรือ|นานาชาติ| |ที่|อ่าว|เอเดน|และ|ชายฝั่ง|โซมาเลีย| |ใน|วันที่| |12| |
กรกฎาคม| |นี้
```

Fig. 3. JSON Service Client Call and Output Example for SWATH



```
ORCHID Result
200 OK
0
หมู่/CLTV|เรือ/NCMN|ปราบปราม/UNK|โจรสลัด/UNK|<space>/PUNC|ฝึก/VACT|ใหญ่/
VATT|เตรียม/VACT|ความ/FIXN|พร้อม/VSTA|<space>/PUNC|ก่อน/ADVN|ออก/
XVAE|ปฏิบัติงาน/VACT|ร่วม/ADVN|กับ/RPRE|กองกำลัง/UNK|ผสม/VACT|ทางเรือ/
UNK|นานาชาติ/NCMN|<space>/PUNC|ที่/PREL|อ่าว/UNK|เอเดน/UNK|และ/JCRG|
ชายฝั่ง/UNK|โซมาเลีย/UNK|<space>/PUNC|ใน/RPRE|วันที่/NCMN|<space>/PUNC|12/
DCNM|<space>/PUNC|กรกฎาคม/NCMN|<space>/PUNC|นี้/DDAC|
```

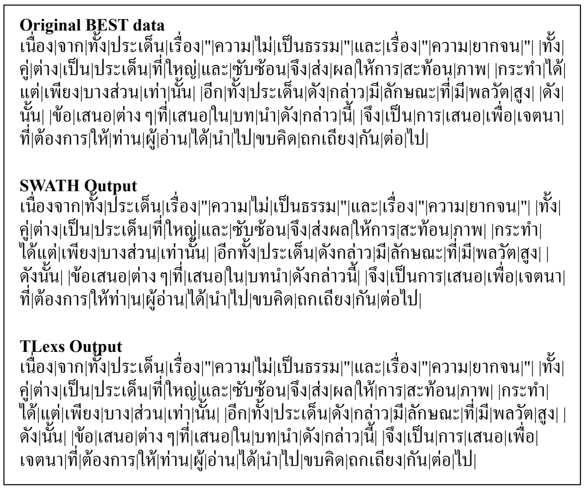
Fig. 4. JSON Service Output Example for ORCHID

words and separated by segmented masks for each file. Thus if we compare the segmentation results with the original BEST training data, we will know the segmentation correctness.

Next, we applied the testing data into each service. Tlexs has input limitation by being able to accept text at most 3,000 characters To avoid the error that might occur, we have splitted the test data into multiple of parts of text. These parts of texts were passed into the TLexs web service client as an input argument for segmentation. To measure the performance of TLexs, We have implemented a SOAP web service client to call the web service and then applied the testing data to get the evaluation results. After that, we also apply these parts of text in to our JSON web service. The result example is shown in Figure 5.

We have applied all testing data and collected the evaluation results for each service. We have evaluated the system effectiveness in the term of precision, recall and f-score [23]. Table 2 is the evaluation result using BEST corpus.

In addition, we have collected the time usage when calling the service for each service. The usage time is measured by simply using 'time' command in GNU/Linux box. The command runs programs and summarize system resource usage in the term of time. We notice that time usage measurement has some complex parameters especially in network latency. The result in Table 3 shows the result of usage time for calling SWATH method for our web service and TLexs service. The result is an average service calling time usage across all testing documents.



```
Original BEST data
เนื่อง|จาก|ทั้ง|ประเด็น|เรื่อง|"|ความ|ไม่|เป็นธรรม|"|และ|เรื่อง|"|ความ|ยากจน|"| |ทั้ง|
คู่|ต่าง|เป็น|ประเด็น|ที่|ใหญ่|และ|ซับซ้อน|จึง|ส่ง|ผล|ให้การ|สะท้อน|ภาพ| |กระทำ|ได้|
แต่|เพียง|บางส่วน|เท่า|นั้น| |อีก|ทั้ง|ประเด็น|ดัง|กล่าว|มี|ลักษณะ|ที่|มี|พลวัต|สูง| |ดัง|
นั้น| |ข้อ|เสนอ|ต่าง ๆ|ที่|เสนอ|ใน|บท|นำ|ดัง|กล่าว|นี้| |จึง|เป็น|การ|เสนอ|เพื่อ|เจตนา|
ที่|ต้องการ|ให้|ท่าน|ผู้|อ่าน|ได้|นำ|ไป|ขบคิด|ถกเถียง|กัน|ต่อ|ไป|

SWATH Output
เนื่องจาก|ทั้ง|ประเด็น|เรื่อง|"|ความ|ไม่|เป็นธรรม|"|และ|เรื่อง|"|ความ|ยากจน|"| |ทั้ง|
คู่|ต่าง|เป็น|ประเด็น|ที่|ใหญ่|และ|ซับซ้อน|จึง|ส่งผล|ให้การ|สะท้อน|ภาพ| |กระทำ|
ได้แต่|เพียง|บางส่วน|เท่านั้น| |อีกทั้ง|ประเด็น|ดังกล่าว|มี|ลักษณะ|ที่|มี|พลวัต|สูง| |
ดังนั้น| |ข้อเสนอ|ต่าง ๆ|ที่|เสนอ|ใน|บทนำ|ดังกล่าวนี้| |จึง|เป็นการ|เสนอ|เพื่อ|เจตนา|
ที่|ต้องการ|ให้ท่า|น|ผู้อ่าน|ได้|นำ|ไป|ขบคิด|ถกเถียง|กัน|ต่อไป|

TLexs Output
เนื่อง|จาก|ทั้ง|ประเด็น|เรื่อง|"|ความ|ไม่|เป็นธรรม|"|และ|เรื่อง|"|ความ|ยากจน|"| |ทั้ง|
คู่|ต่าง|เป็น|ประเด็น|ที่|ใหญ่|และ|ซับซ้อน|จึง|ส่ง|ผล|ให้|การ|สะท้อน|ภาพ| |กระทำ|
ได้|แต่|เพียง|บาง|ส่วน|เท่า|นั้น| |อีก|ทั้ง|ประเด็น|ดัง|กล่าว|มี|ลักษณะ|ที่|มี|พลวัต|สูง| |
ดัง|นั้น| |ข้อ|เสนอ|ต่าง ๆ|ที่|เสนอ|ใน|บท|นำ|ดัง|กล่าว|นี้| |จึง|เป็น|การ|เสนอ|เพื่อ|
เจตนา|ที่|ต้องการ|ให้|ท่าน|ผู้|อ่าน|ได้|นำ|ไป|ขบคิด|ถกเถียง|กัน|ต่อ|ไป|
```

Fig. 5. Web Service Call Output Example

TABLE II
EVALUATION RESULTS

Calling Service	Precision	Recall	F-Score
SWATH	85.93	77.62	81.57
TLexs	95.65	97.45	96.54

We have implement ORCHID POS tagging service. There is no POS tagging service in Thai language to compare. There is no the benchmark tools for tagging service and tagging corpus, expect ORCHID we use. We have applied the prepared training data sets as we described previously. From observation, it has about 60-70 percent correctness. We have collected average time usage for calling this service. The result is shown in the Table 3.

TABLE III
AVERAGE SERVICE CALLING USAGE TIME

Calling Service	real	user	sys
SWATH	1m39.293s	0m5.172s	0m1.176s
ORCHID	4m50.377s	0m11.077s	0m1.212s
TLexs	3m17.045s	0m6.632s	0m1.048s

The result in Table 2 shows that TLexs gives better performance in words segmentation than SWATH because SWATH and TLexs use different methods in words segmentation. SWATH uses maximal matching algorithms but TLexs uses conditional random field method for segmentation. In the term of service calling average usage time, SWATH has a better result. We should not really compare TLexs with SWATH because there are some of testing parameters such as networking delay issue. It nevertheless shows the overview of usage time for calling the service.

VI. Conclusion

We have developed a Thai word segmentation web application and web service with part of speech tagging. Our system has been implemented using Python programming language and Django framework. The system has an web application interface and REST HTTP JSON web service. It includes access API key and registration system. We have evaluated the system performance by comparing the proposed service with the existing TLexs service. The system has lower scores in precision, recall and f-score because we use existing words segmentation algorithm. In the future, we will include more free language software and resources to extend the service functions in our proposed web service.

Acknowledgment

We would like to thank Dr. Choochart Haruechaiyasak for his suggestion about including performance evaluation in this paper.

References

[1] T. Karoonboonyanan, C. Silpa-Anan, P. Kiatisevi, P.Veerathanabutr and V. Ampornaramveth, ”libthai Library”. Available at: http://linux.thai.net/projects/libthai.

[2] P. Charoenpornsawat, ”SWATH (Smart Word Analysis for THai)”. Available at: http://www.cs.cmu.edu/~paisarn/software.html.

[3] T. Karoonboonyanan, ”swath 0.4.1 Released”. Available at: http://linux.thai.net/svn/software/swath.

[4] The Royal Institute of Thailand 2525, ”Thai dictionary words from the royal institute of Thailand 2525”. Available at: http://thailang.nectec.or.th/.

[5] V. Sornlertlamvanich, T.Charoenporn and H.Isahara, ”ORCHID: Thai Part-Of-Speech Tagged Corpus”. National Electronics and Computer Technology Center. Technical Report: TR-NECTEC-1997-001, 1997.

[6] National Electronics and Computer Technology Center (NECTEC), ”BEST Corpus”. Available at: http://thailang.nectec.or.th/best/.

[7] C. Haruechaiyasak and S. Kongyoung, ”TLex: Thai Lexeme Analyser Based on the Conditional Random Fields”, *Proc. 8th International Symposium on Natural Language Processing*, 2009.

[8] National Electronics and Computer Technology Center (NECTEC), ”TLex”. Available at: http://sansarn.com/tlex/.

[9] National Electronics and Computer Technology Center (NECTEC), ”TLexs”. Available at: http://www.sansarn.com/WSeg/wsdl/BnSeg.wsdl.

[10] K. Toutanova and C. D. Manning, ”Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger”. *Proc. the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pp. 63-70, 2000.

[11] D. Roth and D. Zelenko, ”Part of Speech Tagging Using a Network of Linear Separators”, *The 17th International Conference on Computational Linguistics (1998)*, pp. 1136–1142, 1998.

[12] The World Wide Web Consortium. ”Extensible Markup Language (XML) 1.0 (Fifth Edition)”. Available at : http://www.w3.org/TR/REC-xml/.

[13] The World Wide Web Consortium. ”SOAP Version 1.2 Part 1: Messaging Framework (Second Edition)”. Available at : http://www.w3.org/TR/soap12-part1/.

[14] R. T. Fielding. ”Representational State Transfer (REST)”. Available at : http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm.

[15] Ecma International, ”Introducing JSON”. Available at: http://www.json.org/

[16] Ecma International, ”Standard ECMA-262 5.1 Edition / June 2011 : ECMAScript Language Specification”. Available at: http://www.ecma-international.org/publications/files/ecma-st/ECMA-262.pdf

[17] Wikipedia, ”Comparison of web browsers”. Available at: http://en.wikipedia.org/wiki/Comparison_of_web_browsers

[18] C. Haruechaiyasak, S. Kongyoung and M. N. Dailey. ”A Comparative Study on Thai Word Segmentation Approaches”, *Proc. 5th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (2008)*, pp.125–128, 2008.

[19] G. van Rossum, ”Python tutorial, Technical Report CS-R9526”, *Centrum voor Wiskunde en Informatica (CWI)*, Amsterdam, May, 1995.

[20] S. Bird and E. Loper, ”NLTK: The Natural Language Toolkit”, *Proceedings of the ACL Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics*, pp. 62-69, 2002.

[21] Django Software Foundation, ”The Django framework”. Available at : https://www.djangoproject.com/.

[22] ”LigHTTPD fly light”. Available at : http://www.lighttpd.net/.

[23] C. D. Manning, P. Raghavan and H. Schutze. ”An Introduction to Information Retrieval”, *Cambridge University Press, Cambridge, England*, pp.155, 2009.

[24] G. Mulligan and D. Gračanin. ”A Comparison of SOAP And REST Implementation of a Service Based Interaction Independence Middleware Framework”, *Proceedings of the Winter Simulation Conference*, pp.1423-1432, 2009.

[25] G. Wang. ”Improving Data Transmission in Web Applications via the Translation between XML and JSON”, *Third International Conference on Communications and Mobile Computing*, pp.182-185, 2011.