Disambiguation of Thai Personal Name from Online News Articles

Phaisarn Sutheebanjard
Graduate School of Information Technology
Siam University
Bangkok, Thailand
mr.phaisarn@gmail.com

Wichian Premchaiswadi
Graduate School of Information Technology
Siam University
Bangkok, Thailand
wichian@siam.edu

Abstract— **Since online news articles are updated daily, hourly and sometimes every minute, therefore the data from online news articles are glowing rapidly. These data seem like a large corpus of text mining. This research focuses on Thai personal names that appear in the online news which sometimes have slightly different spelling but they actually refer to the same person. From the news data that were collected during 30 July 2009 to 5 November 2009, there are a lot of name variations. The objective of this paper is to disambiguate Thai personal names by applying string matching techniques which are Guth, Levenshtein, Damerau-Levenshtein, Longest Common Substring and Longest Common Subsequence. The experimental results show that the Longest Common Subsequence was the most efficient technique for matching Thai personal name with the F-Score of 94.43%. After that, the two-scan labeling technique was used to identify the unique full Thai personal name. The results show that it can reduce the 6,884 distinct personal names to 830 unique personal named entities which equals to 12.057% reduction.**

Keywords; personal name; string matching; online news; two-scan labeling

I. INTRODUCTION

Nowadays, online news is wildly used in data mining and text mining because an enormous amount of data is updated everyday. Online news seems like a large corpus of data. There are many full personal names appear in online news, but it is difficult to identify the unique full personal name. One of the most common problems is spelling variations such as transcription differences and misspellings (mistyping). Misspellings can be categorized as insertion, deletion or omission, substitution and transposition [1]. Generally, such variations do not affect the phonetic structure of the name but still cause problems in matching names, this problem is called string matching problem.

String matching has been extensively studied for the past 30 years. It is an everlasting interesting topic in computer science. String matching is a fundamental issue in computer science and is widely used in many applications. String matching attempts to measures the similarity between strings. This is useful for applications ranging from database de-duplication and record linkage to terminology extraction and spell checking.

This paper proposed the method to extract disambiguation of Thai personal named entity from online news articles. The proposed method composes of three processes. Firstly, extract personal named entity from online news articles. Secondly, matching personal name with five strings matching techniques and compare their performances. Thirdly, identify the unique personal name by using two-scan labeling technique.

II. VARIATION OF FULL THAI PERSONAL NAMES

Name variation is one of the major problems in identifying people, because it is not easy to determine whether a name variation is a different spelling of the same name or a name for a different person [2]. The same problem occurs with personal names in Thai language as well. This research investigates full Thai personal names that were appeared in the online news articles during 30 July 2009 to 5 November 2009. Apparently, there were many variously spelled personal names that referred to the same person. Table I shows the variants of personal name called "อภิสิทธิ์ เวชชาชีวะ" and the number of times that the name appeared in the online news articles during that period.

TABLE I. VARIANTS OF PERSONAL NAME "อภิสิทธิ์ เวชชาชีวะ"

Variation	No of appearance	Variation	No of appearance
อภิสิทธิ์ เวชชาชีวะ	4,418	อภิสทิธิ์ เวชชาชีวะ	1
อภิสิทธิ์ เวชาชีวะ	29	อภิสิท▮ เวชชาชีวะ	1
อภิสิทธ์ เวชชาชีวะ	15	อภิสิทธิ์ เวชชาชะ	1
อภิสิทธิ เวชชาชีวะ	14	อภิสิทธิ์ เวชชาชีรวะ	1
อภิสิทธิ์ เวชชีวะ	9	อภิสิทธิ์ เวชชาชีว	1
อภสิทธิ์ เวชชาชีวะ	4	อภิสิทธิ์ เวชชาวะ	1
อภิสิทธิ์ เวชชชาชีวะ	3	อภิสิทธิ์ เวชช่าชีวะ	1
อภิสทธิ์ เวชชาชีวะ	2	อภิสิทธ์ เวชชาวีวะ	1
อภิสิทธิ์ เวชาชาชีวะ	2	อภิสิทธ์ เวชาชีวะ	1
อภภิสิทธิ์ เวชชาชีวะ	1		

After applying name matching technique, we found some reasonable alternatives of the name, "อภิสิทธิ์ เวชชาชีวะ" as shown in table 1. Consequently, all the alternative names of "อภิสิทธิ์ เวชชาชีวะ" can be assumed that they refer to the same name.

III. THAI PERSONAL NAMED ENTITY EXTRACTION

Named entity (NE) extraction is the task that identifies expressions such as entity names (person, organization, and location name), temporal expressions (date and time), and numerical expression (monetary values and percentages) [3]. Especially on person named entity, it is an important in some kinds of applications like a people search engine such as pipl.com.

In 1998 Charoenpornsawat et al. [4] proposed an approach to identify Thai proper names partially composed of known and unknown substrings using probabilistic

978-1-4244-6350-3/10/$26.00 ©2010 IEEE

trigram models and the Winnow algorithm. The features used in the method were context words, collocations and part of speech (POS) tag, as well as heuristics information from dictionary and POS to generate named entity candidates to solve named entity boundary problem. The accuracy presented in the system was 92.17%. In 2004 Chanlekha and Kawtrakul [3] proposed an approach to extract Thai multiword named entity by using combination of rule-based, dictionary-based and statistical-based models to predict boundaries of named entities. The method applied Maximum Entropy model and incorporate knowledge, which are rules and dictionary to named entity extraction system such as extracting personal names, locating personal names and organization names. In 2009 Sutheebanjard and Premchaiswadi [5] proposed the method that uses front and rear context to automatically extract Thai personal named entity from plain text of online political, financial and sport news articles. This method uses neither word segmentation nor POS tagging process which can significantly reduce the effort and speed up the process in building the training corpus. Moreover, it also eliminates the effects on the efficiency of using word segmentation and POS tagging on named entity finding. The accuracy presented in the system was 91.720%. Thai named entity extraction method can be divided into three groups as follows:

- Using word segmentation such as [3].
- Using word segmentation and POS tagging such as [4], [6-7].
- Neither used word segmentation nor POS tagging such as [5].

IV. String Matching Techniques

String matching is a very important subject in the wider domain of text processing. String matching is the algorithms that try to find a place where one or several strings (also called patterns) are found within a larger string or text. There are many strings matching methods such as Guth, Levenshtein, Damerau-Levenshtein, Longest Common Substring and Longest Common Subsequence.

A. *Guth*

Guth's algorithm [8] is a letter-by-letter matching algorithm. The algorithm is left to right sequence driven, and is essentially alphabetic but is independent of language and ethnic issues. Guth's algorithm does not depend on recognition of phonetic similarity. It is able to identify variant spellings through the position of letters in names. It is, however, weak when comparing short names where one or two common vowels can produce a mismatch [10].

The algorithm first checks for the case that two names are identical, considering each name as a single character string. If this fails, the algorithm proceeds to compare the names letter-by-letter and when the program encounters different letters in the same position it then searches for matching letters in other positions. The comparison pattern is illustrated in table II [8-9].

TABLE II. GUTH MATCHING COMPARISONS

	Position of Name 1	Position of Name 2
Test 1	X	X
Test 2	X	X+1
Test 3	X	X+2
Test 4	X	X-1
Test 5	X-1	X
Test 6	X+1	X
Test 7	X+2	X
Test 8	X+1	X+1
Test 9	X+2	X+2

B. *Levenshtein or edit distance*

Levenshtein or edit distance [11] is a measure of the similarity between two strings. It is obtained by finding the cheapest way to transform one string into another. Transformations are the one-step operations of insertion, deletion and substitution. It is defined to be the smallest number of edit operations (insertions, deletions and substitutions) required to change one string into another.

For example, define the original strings as s_1 and s_2. The distance is the number of deletions, insertions, or substitutions required to transform s_1 into s_2.

- If s_1 is “test” and s_2 is “test”, then Distance(s_1,s_2)=0, because no transformations are needed. The strings are already identical.
- If s_1 is “test” and s_2 is “tent”, then Distance(s_1,s_2)=1, because one substitution (change “s” to “n”) is sufficient to transform s_1 into s_2.

C. *Damerau-Levenshtein distance*

Damerau-Levenshtein distance was introduced by Damerau [12] and Levenshtein [11]. A Damerau-Levenshtein distance function is a variant of the Levenshtein distance function where the transposition operation counts as a single operation (in the Levenshtein distance, a transposition corresponds to two edits: one insert and one delete or two substitutions).

D. *Longest Common Substring*

The longest common substring is based on the notion of likeness measure between two strings. The likeness measure is obtained using a procedure which iteratively finds and removes the longest common substring between two strings. The likeness measure is based on the total length of the common portions of the name pairs compared to the length of the actual names [13]. This algorithm repeatedly finds and removes the longest common sub-string in the two strings compared, up to a minimum length (normally set to 2 or 3). For example, the two name strings ‘gail west’ and ‘vest abigail’ have a longest common sub-string ‘gail’. After it is removed, the two new strings are ‘ west’ and ‘vest abi’. In the second iteration the sub-string ‘est’ is removed, leaving ‘ w’ and ‘v abi’. The total length of the common sub-strings is now 7. If the minimum common length would be set to 1, then the common whitespace character would be counted towards the total common sub-strings length as well [2].

E. *Longest Common Subsequence*

The longest common subsequence problem is to find a longest common subsequence of two given strings. Given two sequences of strings, find the longest common subsequence present in both of them. A subsequence is a sequence that appears in the same relative order, but not necessarily contiguous. For example, in the string abcdefg, "abc", "abg", "bdf", "aeg" are all subsequences.

The longest common subsequence problem has what is called an "optimal substructure"; the problem can be broken down into smaller, simple sub-problems, which can be broken down into yet simpler sub-problems until the solution becomes trivial. There are two general approaches to the longest common subsequence problem. The dynamic programming approach takes quadratic time but linear space, while the nondynamic-programming approach takes less time but more space. Recently, XuYu xiang et al. [14] proposed the nondyamic-programming implementation with efficient in both time and space.

V. The Proposed Method

The proposed system is divided into 3 processes. The first process extracts Thai personal named entity from online news articles. The second process compares five string matching techniques in order to find the most efficient method for matching Thai personal name. The final process disambiguates Thai personal names by using two-scan labeling technique.

A. *Thai personal named entity extraction*

To reduce times and efforts in building training corpus and eliminate the effects on the efficiency of using word segmentation and POS tagging, the method of Sutheebanjard and Premchaiswadi [5] was used to extract Thai personal named entity from online news articles. This method used plain text as the input of the system and applied contextual environment of Thai personal name to compare against the word list. The word list is a list of words that could possibly have the same contextual as Thai personal name, and then apply a simple rule base to recognize Thai personal name from plain text as shown in Fig. 1.

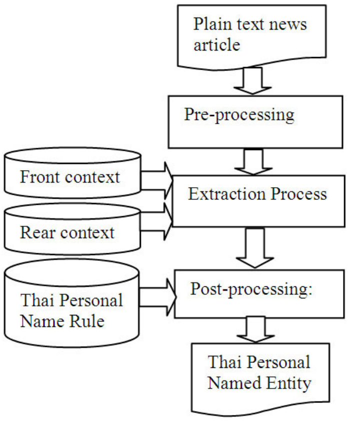

Figure 1. Thai personal named entity extraction.

B. *The comparison of five string matching techniques*

In online news articles, there are variously spelled personal names that refer to the same person. The use of exact matching leads to poor results. This process compares the following five string matching techniques:

- Guth
- Levenshtein
- Damerau-Levenshtein distance
- Longest Common Substring
- Longest Common Subsequence

In order to measure string similarity among those methods, this experiment used string distances obtained from the Levenshtein and Damerau-Levenshtein techniques to calculate string similarity by using (1), and string distances obtained from Guth, Longest Common Substring and Longest Common Subsequence techniques to calculate string similarity by using (2).

$$sim(s_1,s_2)=1.0-\frac{dist(s_1,s_2)}{\max(|s_1|,|s_2|)} \qquad (1)$$

$$sim(s_1,s_2)=\frac{dist(s_1,s_2)}{\max(|s_1|,|s_2|)} \qquad (2)$$

Whereas:

- $sim(s_1,s_2)$ denotes similarity of string1 and string2; result value between 0.0 and 1.0
- $dist(s_1,s_2)$ denotes distance between string1 and string2
- $max(|s_1|,|s_2|)$ denotes maximum length between string1 and string2

In addition, Precision (P), Recall (R), and F-measure (F) were computed to evaluate the performance of the proposed method by using (3), (4), and (5) respectively.

$$P=\frac{TP}{TP+FP} \qquad (3)$$

$$R=\frac{TP}{TP+FN} \qquad (4)$$

$$F=\frac{2PR}{P+R} \qquad (5)$$

Whereas:

TP (True positives) refers to positive examples correctly labeled as positives. FP (False positives) refers to negative examples incorrectly labeled as positive. TN (True negatives) corresponds to negatives correctly labeled as negative. Finally, FN (false negatives) refers to positive examples incorrectly labeled as negative [15], as summarized in table III.

TABLE III. Confusion Matrix

	Actual positive	**Actual negative**
Predicted positive	TP	FP
Predicted negative	FN	TN

By using string matching technique, every personal name was compared with one another. Therefore, it produced a bunch of many-to-many mappings pairs of personal names. After that, the following technique was used in order to identity unique full personal names.

C. Disambiguation of Thai personal name by using two-scan labeling technique

Identifying the occurrences of personal names is a difficult task because of the many-to-many names mapping process. To identify the unique personal name, this research applied the two-scan labeling technique which is widely used for labeling connected components in a binary image [16]. The two-scan labeling technique applied in this research is composed of two scans. The first scan was done in order to store the label equivalences of each personal name in a 2-D array. Then the provisional labels are replaced by the smallest equivalent label with use of the 2-D array during the second scan. For example, some variants of two personal names, "A" and "B" was shown in table IV.

TABLE IV. THE RESULTS OF FIRST SCAN AND SECOND SCANLABELING

Name1	**Name2**	**First Scan**		**Second Scan**	
		Label of Name1	***Label of Name2***	***Label of Name1***	***Label of Name2***
A1	A2	1	2	1	1
A1	A3	1	3	1	1
A2	A3	2	3	1	1
A2	A4	2	4	1	1
A5	A6	5	6	1	1
A1	A5	1	5	1	1
B1	B2	7	8	7	7
A7	A1	9	1	1	1
A7	A8	9	10	1	1
B2	B3	8	11	7	7

VI. EXPERIMENTAL RESULTS

In this experiment, Thai online news articles were collected from 30 July 2009 to 5 November 2009, 99 days in totals. During that time frame, there were 26,797 online news articles consisting of 9,231 economic news articles, 9,468 political news articles, 11,213 breaking news articles and 2,747 stock news articles.

The experimental results can be categorized into three parts. The first part used the method of Sutheebanjard and Premchaiswadi [5] to extract full personal name from 26,797 online news articles, and a total of 57,361 personal names were found. And among these 57,361 personal names, there were distinct 6,884 personal names.

The second part is to evaluate the efficiency of the five string matching techniques from the distinct 6,884 personal names obtained from the first part. The experimental results from computing the Precision, recall and F-Score at different similarity levels are shown in Fig. 2-4 and table V-VII.

The final part was to find the unique personal named entity by applying two-scan labeling technique on the pairs of personal names. Among the 2,609 correct pairs of personal names, there were 830 unique personal named entities. The top ten personal names that have the most number of pairs and name variation are shown in table VIII.

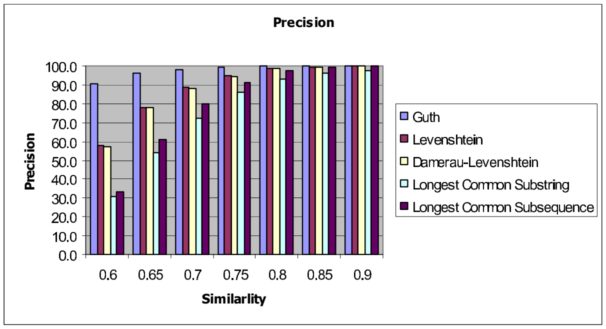

Figure 2. Precision at different similarity measure level.

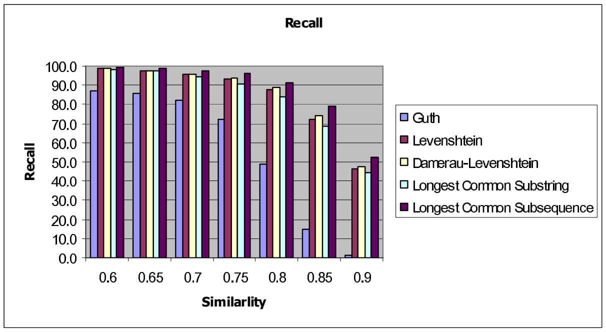

Figure 3. Recall at different similarity measure level.

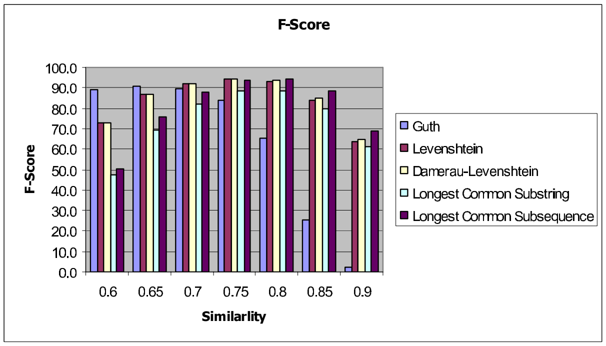

Figure 4. F-score at different similarity measure level.

VII. CONCLUSION

This research proposed an efficient method to disambiguate Thai personal name entity in online news articles. This research also compared the performance of five string matching techniques in matching full Thai personal names. The results show that the Longest Common Subsequence technique yields the best results in matching full Thai personal names at the similarity level of 0.8 with F-Score of 94.43%. After that, the two-scan labeling technique was used to identity unique full personal names. With the success of this research, the 6,884 distinct personal names were reduced to 830 unique personal named entities, which equals to 12.057% reduction. Therefore, it can be concluded that the proposed method can reduce Thai personal name variations in the online news articles.

TABLE V. PRECISION AT DIFFERENT SIMILARITY MEASURE LEVEL

Similarity Level	**0.60**	**0.65**	**0.70**	**0.75**	**0.80**	**0.85**	**0.90**
Guth	90.79	96.25	98.30	99.42	99.84	99.74	100
Levenshtein	57.56	78.14	88.42	94.67	98.49	99.58	100
Damerau-Levenshtein	57.41	78.06	88.34	94.65	98.51	99.59	100
Longest Common Substring	31.00	54.06	72.08	86.01	93.06	96.23	97.57
Longest Common Subsequence	33.58	61.28	79.70	91.13	97.43	99.47	100

TABLE VI. RECALL AT DIFFERENT SIMILARITY MEASURE LEVEL

Similarity Level	**0.60**	**0.65**	**0.70**	**0.75**	**0.80**	**0.85**	**0.90**
Guth	86.93	85.51	82.06	72.29	48.49	14.53	1.03
Levenshtein	98.54	97.55	95.75	93.25	87.70	72.33	46.34
Damerau-Levenshtein	98.62	97.62	95.82	93.60	88.92	74.28	47.80
Longest Common Substring	98.31	97.24	94.60	90.95	84.25	68.53	44.54
Longest Common Subsequence	99.20	98.70	97.51	96.13	91.61	79.30	52.59

TABLE VII. F-SCORE AT DIFFERENT SIMILARITY MEASURE LEVEL

Similarity Level	**0.60**	**0.65**	**0.70**	**0.75**	**0.80**	**0.85**	**0.90**
Guth	88.82	**90.56**	89.45	83.71	65.27	25.36	2.05
Levenshtein	72.67	86.77	91.94	**93.96**	92.78	83.79	63.33
Damerau-Levenshtein	72.57	86.75	91.93	**94.12**	93.47	85.09	64.68
Longest Common Substring	47.14	69.49	81.82	88.41	**88.43**	80.05	61.16
Longest Common Subsequence	50.17	75.61	87.71	93.56	**94.43**	88.25	68.93

TABLE VIII. TOP TEN OF UNIQUE PERSONAL NAMES

No	**Personal Name**	**No. of pairs of personal name**	**No. of variance**
1	อภิสิทธิ์ เวชชาชีวะ	190	19
2	พัชวาท วงษ์สุวรรณ	91	13
3	กรณ์ จาติกวณิช	78	12
4	ชินวรณ์ บุณยเกียรติ	66	11
5	สุเทพ เทือกสุบรรณ	55	10
6	ชวรัตน์ ชาญวีรกูล	48	10
7	ประทีป ตันประเสริฐ	45	10
8	ปานเทพ พัวพงศ์พันธ์	36	8
9	จตุพร พรหมพันธุ์	28	7
10	ปิยะพันธ์ นิมมานเหมินทร์	28	7

REFERENCES

[1] C. Snae, “A comparison and analysis of name matching algorithms,” Proceedings of world academy of science, engineering and technology vol. 21, January 2007.

[2] P. Christen, “A comparison of personal name matching techniques and practical issues,” ICDM Workshops, December 2006.

[3] H. Chanlekha, A. Kawtrakul, “Thai named entity extraction by incorporating maximum entropy model with simple heuristic information,” 1st International Joint Conference on Natural Language Processing, 2004.

[4] P.Charoenpornsawat, B. Kijsirikul, and S. Meknavin, “Feature-based proper name identification in Thai,” National Computer Science and Engineering Conference: NCSEC’98, Thailand, 1998.

[5] P. Sutheebanjard and W. Premchaiswadi, “Thai personal named entity extraction without using word segmentation or POS tagging,” . Natural Language Processing, SNLP'09, pp. 221 – 226, October 2009.

[6] H. Chanlekha, A. Kawtrakul, P. Varasrai and I. Mulasas, “Statistical and heuristic rule based model for Thai named entity recognition,” in Natural Language Processing, SNLP'02, 2002.

[7] B. Kijsirikul, “Comparing Winnow and RIPPER in Thai named-entity identification,” Natural Language Processing Pacific Rim Symposium 1999 (NLPRS’99), Bejing, China, 1999.

[8] G.J.A. Guth, "Surname spellings and computerized record linkage," Historical Methods Newsletter, vol. 10, pp.10-19, 1976.

[9] A. J. Lait and B. Randell, “An assessment of name matching algorithms,” Society of Indexers Genealogical Group, Newsletter Contents, SIGGNL issues 17, 1998.

[10] G. De Brou and M. Olsen, “The Guth algorithm and the nominal record linkage of multi-ethnic populations,” Historical Methods, vol. 19, pp.20-24, 1986.

[11] V. I. Levenshtein, “Binary codes capable of correcting deletions, insertions and reversals,” Sov. Phys. Dokl., vol. 6, pp. 707-710, 1966.

[12] Fred J. Damerau, “A technique for computer detection and correction of spelling errors,” Communications of the ACM 7, 1964.

[13] C. Friedman and R. Sideli, “Tolerating spelling errors during patient validation,” Computer an dbiomedical research 25, 486-509, 1992.

[14] X. Xiang, D. Zhang and J. Qin, “An improve algorithm for the longest common subsequence problem,” International Conference on Convergence Information Technology, pp.637-639, November 2007.

[15] J. Davis and M. Goadrich, “The relationship between precision-recall and ROC curves,” 23rd international conference on Machine learning, pp.233-240, Pittsburgh, Pennsylvania, June 2006.

[16] L. He, Y. Chao and K. Suzuki, “A run-based two-scan labeling algorithm, ” IEEE Transaction on Image Processing vol. 17, 2008.