Int. J. Advance Soft Compu. Appl, Vol. 12, No. 1, March 2020
ISSN 2074-8523

Segregation of Code-Switching Sentences using Rule-Based Technique

Emaliana Kasmuri, Halizah Basiron

Fakulti Teknologi Maklumat dan Komunikasi,
Universiti Teknikal Malaysia Melaka, 76100 Durian Tunggal Melaka
email: emaliana@utem.edu.my

Fakulti Teknologi Maklumat dan Komunikasi,
Universiti Teknikal Malaysia Melaka, 76100 Durian Tunggal Melaka
email: halizah@utem.edu.my

Abstract

Code-switching sentence contains a mixture of two or more languages within a single constructed sentence. Code-switching is a new trend of language that is widely used in open platform such as blogs and social medias. Consequently, code-switching which has become a new challenge to natural language processing (NLP). The challenge is due to the limitation of the existing NLP systems which were designed for mono-lingual system. Therefore, a new NLP system is needed to deal with code-switching sentences. However, system that segregate code-switching sentences from mono-lingual sentences must be developed prior to the code-switching sentences are used in the NLP systems. This paper considers the segregation is essential because firstly the current NLP systems deals only with mono-lingual sentences. Secondly the current NLP systems treats switching words as meaningless thus will lead to inaccurate result. This paper segregates code-switching sentences from mono-lingual sentences using rule-based technique and dictionaries. This paper used the ratio of word presence to segregate the sentences. The rule-based technique performed with accuracy of more than 87.00% for Malay-English code-switching (MY-EN-CS) sentences.

Keywords: *code-switching sentence, mono-lingual sentence, rule-based technique, sentence segregation*

1 Introduction

Mixing two or more languages in the construction of a sentence is common among bi-lingual speakers especially in multi-ethnic community. The mixing has produced a new kind of language known as code-switching. It has become a trend to use code-switching in spoken and written in open platform such as in blog and social media postings.

Code-switching is not a random process. However, it happened at the convenient of the speaker. This gives a new challenge to the NLP and the need to computationally process the code-switching text is emergent. A systematic computational analysis is necessary to develop better language model suitable for code-switching. Thus, a better result can be produced. The traditional machine translation is not a feasible solution as it would treat the code-switching data as noise or just a regular sentence [1]. The result produced from machine translation for the code-switching text could be misleading.

This paper segregates code-switching sentences from mono-lingual sentences using rule-based technique incorporated with dictionaries. Five dictionaries and two sets of rules were used in this paper for the segregation of the sentences. The first set of the rules was designed to identify the name-entity due to the unavailability of the named-entity identifier for the code-switching text. The second set of rules was designed to segregate the sentences into the categories defined in this paper.

The rest of the paper is structured in the following sections. The previous studies on computational processing of code-switching textual document is discussed in Section 2. The preparation of Malay-English code-switching (MY-EN-CS) dataset is described in Section 3. The pre-processing of the dataset is described in Section 4. The proposed rule-based technique is introduced at Section 5. The results are presented at Section 6. Finally, Section 7 concludes this paper and outlined the direction for the future work.

2 Previous Studies

The computational study of code-switching text can be considered as relatively new. Even though it was first mentioned in the early of 1980s, the study of computational code-switching shown relatively small progress until 2008. From 2008 the computational study of code-switching slowly gain momentum. This paper assumed the trend of code-switching before 2008 was not as apparent as today due to the accessibility to the data and multi-lingual speaker involvement on open platform such as blogs and social medias. Table 1 gives an overview of for the computational study of code-switching text. Furthermore, the study of code-switching text has been applied in information retrieval[2], speech-recognition [3][4], sentiment analysis [5][6], emotion analysis [7] and topic identification [8].

Table 1: Areas of study for computational code-switching text

Area of study	**Reference Articles**	**Total**
Language identification	[9][10][11][12][13][14][15]	7
Code-switching point prediction	[16][17]	2
Code-switching identification	[18][19]	2
Corpus creation	[20]	1
Automated code-switching text generation	[21]	1
Code-switching identification within word	[22]	1

The issue of code-switching was first addressed by NLP computing in 1982 [23]. The study introduced a model to formalize the characteristics of code-switching sentences. The model introduced the concept of matrix language and embedding language. Matrix language is defined as the primary language used by the speaker, which usually is the local language, whereas embedding language is defined as foreign language or the second language.

The most studied computing area of code-switching is language identification as shown in Table 1. Language identification in code-switching sentence is one of the sub-tasks at document level language identification task [14]. In the computing study of code-switching, language is identified at word level. Each word in the sentences are assessed to determine its language. The majority of the study incorporated dictionary and learning-based techniques.

Incorporating dictionary with machine learning technique was used to identify language at word-level in code-switching text [11]. In the proposed approach, the study used three techniques. The first technique was to identify the language of the analyzed word using unsupervised dictionary-based. The second technique was supervised word-level classification with and without contextual clues with dictionary. Lastly, the third technique was sequence labelling using Conditional Random Fields. The results from the study concluded that dictionary-based approach has surpassed the supervised classification technique.

Another technique to identify language at word level was the use of unsupervised learning technique [10]. The technique used a combination of character n-gram that consists unigram, bigram, trigram, 4-grams and 5-grams as features with logistic regression classifier. The classifier performed above 88.00% accuracy with the combination of all character n-gram and analyzed word.

A combination of dictionary-based technique, supervised machine learning and neural-network was used to identify and classify language at word level in code-switching text [15]. The study used character n-gram technique, dictionary-based labels, length of words and word capitalization as features. Two pairs of datasets that contains tweets for Nepali-English and Spanish-English was used to train the classifier using Support Vector Machine (SVM) and k-nearest neighbor (KNN).

The classifier achieved 96.30% and 84.40% accuracy for Nepali-English and in Spanish-English respectively. The study reported that the accuracy of the system is improved with the addition of neural network features.

A simple technique of incremental char n-gram was used to identify language at word level [9]. The incremental char n-gram technique was used to search the analyzed word in two dictionaries. The study started with 4-grams for words that contains three and more letters. The technique continued to increase the character until reach to the last character. The incremental technique was to address the variant length of word in the processed text. Scoring technique was used to determine the language of the analyzed word. The language was assigned to the bigger computed scoring value. The proposed technique performed at accuracy 95.1% for Tweets dataset and 79.4% accuracy in Facebook posting dataset.

Besides identifying languages in code-switching data, predicting code-switching point has become an important computational study for code-switching text. Code-switching point is the point where the speaker changed from one language to another before he ends his utterance or writings. A better technique can be developed to process code-switching text effectively with these predicted points [16]. Machine learning technique was used to predict the code-switching point [16]. The study reported that Naive Bayes and VFI can predict code-switch point with an acceptable F-measures in Spanish-English data even though the task was challenging. Another study used pattern-based approach to identify code-switching point [17]. The evaluation of the study has shown that pattern-based approach performed at 94.51% accuracy.

Another interesting study has generated code-switching text automatically [21]. The automated generation of code-switching text was an expert-based system that selects the word to be changed to another language. The technique incorporates knowledge content and a sequence of words to be changed to another language. A discriminative model was used to facilitate the selection of code-switching points. The model learned the code-switching point from the corpus created by this study.

The majority of code-switching studies in NLP have involves a pair of languages in which English was the embedding language. For example, Marathi-English [23], Spanish-English [16], Korean-English [2], Chinese-English [2], Nepali-English [9], Mandarin-English [3][18], Tagalog-English [17], Urdu-English [5] and French-English [24]. There were also studies of code-switching which English was not the embedding language. For example Arabic-French [20] and Turkish-Dutch [15]. Besides that, there was a study that analyzed mixture of three languages, Bengali-English-Hindi within a text [11].

Based on the aforementioned studies, this paper concludes that less attention is given to the segregation between code-switching text and mono-lingual text. The segregation is essential due to the different characteristics that exists in code-switching text and mono-lingual text. Furthermore, the segregation is also essential due the inadequacy of the existing system to process code-switching text. Therefore, this paper proposed a rule-based technique for the segregation.

3 Dataset Preparation

This paper has gathered data about Malaysia from blogs. The content of the blogs was downloaded using Python script and separated into individual sentences for annotation process. Annotation process is a process to label the sentences using a set of predefined labels. The annotated sentences are regard as a gold-standard dataset that will be as a comparison to the result produced from the rule-based technique proposed in this paper.

After the content of the blogs are separated into individual sentences, the sentences that contains between three (3) to 20 words are selected to be annotated by the appointed annotators. The sentences that contains less than three words are considered as non-informative by this paper and the sentences that contains more than 25 words contains excessively overwhelming information. Furthermore, this paper has found that sentences that contains more than 20 words were poorly constructed sentence. These sentences expressed the discussed subject matter in an unorganized manner and improper use of punctuations.

The selected sentences were labelled either as *my* referring to the Malay sentences, *en* referring to the English sentences and *my-en-cs* referring to the Malay-English code-switching sentences by two appointed annotators. These annotators were undergraduate students who were proficient in spoken and written using Malay and English languages.

This paper has annotated 6,543 sentences with reliability of 0.83 Cohen-Kappa. Table 2 shows the distribution of the sentences labelled by this paper. The distribution of data shows that code-switching was highly used by the bloggers. Although the number of *my*-labelled sentences are low compared to *en*-label and *my-en-cs*-labelled sentences, *my*-labelled sentences were not discarded from the dataset. This paper used *my*-labelled sentences as a control variable to validate the result from the proposed rule-based technique.

Table 2. Number of sentences per language label

Language	**Label**	**Number of sentences**	**Ratio**
Malay	my	415	6.00%
English	en	3,132	48.00%
Malay-English code-switching	my-en-cs	2,996	46.00%
Total		**6,543**	**100.00%**

4 Text Pre-Processing

This paper deemed *my*-labelled and *en*-labelled sentences as mono-lingual sentences and *my-en-cs*-labelled sentences as code-switching sentences. The type of word was used to segregate code-switching sentences from mono-lingual sentences. The sentences were broken into individual unit known as token and these tokens were processed individually.

The sentences are tokenized into 1-gram token. The 1-gram token represents the individual unit of words in the sentence. Each token was validated using a set of rules. Only selected tokens were processible by this paper. This paper deemed valid token as a token that was composed with alphabets only. However, alphanumeric tokens and tokens that contains mixture of alphabets and punctuations were present in the dataset. Alphanumeric tokens were ignored by this paper because it does not have an indicator for any analyzed languages. Furthermore, considering it might leads to inaccurate result. Besides that, hyphens (-) were present in the middle of some tokens. This paper replaced these hyphens with a space to increase the processability of the token and to be labelled into one of the analyzed languages. Only selected punctuations such as double quotes ("), comma (,), question mark (?), exclamation mark (!) and full stop (.) that were present at the beginning or the end of the token was removed by this paper for the same reason. The process is illustrated in Fig.1.

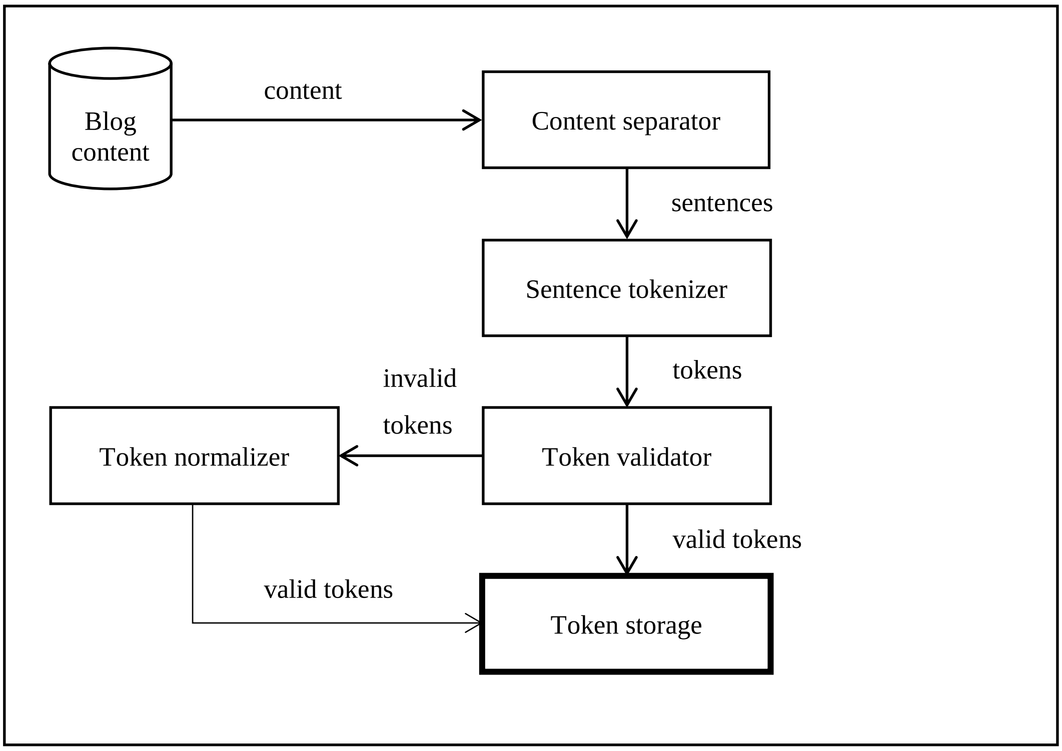

Fig.1 Flow of blog's content pre-processor

5 Rule-Based Technique

This paper used five dictionaries for the rule as look-up resources to determine the categories. The categories of the tokens are described in Table 3. The dictionaries used in this paper are interjections, English stop words derived from Python's NLTK's library, English WordNet, Malay stop words [25] and Malay WordNet [26].

Table 3. Categories of tokens

Category	**Description**
length	The number of words used in a sentence.
sw-en	English stop words
fn-en	English functional words are words that are registered in the English dictionary used in this paper.
total_en	Total of English words in a sentence
percent_en	Percentage of English words in a sentence
sw-my	Malay stop words
fn-my	Malay functional words are words that are registered in the Malay dictionary used in this paper.
total_my	Total of Malay words in a sentence
percent_my	Percentage of Malay words in a sentence.
ne	Words that are used to described entity such as a person name, organization and events.
ood	Out-of-dictionary (ood) are words that are not registered in the English and Malay dictionary used in this paper.
intj	Interjections are words that are used to describe the sounds of emotions made by human. These words that are not registered in the dictionaries used in this paper.

Named-entity was identified using simple rule-based technique as shown in Fig.2. This rule-based technique was used due to unavailability of named-entity identifier designed for code-switching. Therefore, this paper used capitalization (words that start with capital letters that are located in the middle of the sentence) was used to identify the named-entity.

```
for analyzed_token tokens:
    first_letter = first letter of analyzed_token
    if index of analyzed_token > 0 AND first_letter is upper case
        analyzed_token is named-entity
        update frequency of named-entity
```

Fig.2 Rule design for named-entity

This paper has compiled a list of interjections from the Internet[1] and the annotated dataset. The interjections were categorized according to their language and polarity. Each interjection was manually labelled either as *my* (Malay) or *en* (English) for the language label and p (positive), n (negative) or *x* (neutral) for the polarity label. A total of 179 English and 18 Malay interjections were labelled by this paper.

Each token is assigned to one of the categories described in Table 3 and the count of each categories are updated. The focus of this paper is to segregate code-switching sentences from mono-lingual sentences. Therefore, the presence of words from any of the analyzed languages are the determinant. This paper found that the length of the sentence is not feasible to be used as the measure to determine the sentences either as mono-lingual or code-switching. The length of a sentence is a misleading measurement since the length includes the non-processible tokens. Therefore, length of the sentence is discarded by this paper.

This paper did not include interjections, named-entity and out-of-dictionary tokens into the count of total words used in a sentence. The used of interjections and named-entity were not constrained to a certain language and it has become a global use. Furthermore, named-entity is not supposed to be different when it used across language.

This paper defined mono-lingual sentence as $t_l = \{w_{kl}\}$ where

$w_{kl}$is element of t_l

k is 1…n.

Two languages were used to construct code-switching sentences. This paper defined code-switching as $t_{l1l2} = \{w_{kl1}, w_{kl2}\}$ where

w_{kl1} ,w_{kl2} are elements of t_{l1l2}

k is 1…n.

The probability of word presence was used to determine the category of the sentence instead of the total number of words. The probability normalized the variant number of total words in each sentence. The probability of word presence from l_i given by $Pb(l_i) = \frac{n(l_i)}{n(t)}$.

5.1. Analysis of Processible Words

This paper has processed 6,865 words from 6,543 sentences. A total of 6,165 words from 6,865 words were processible words, deemed as valid token as described in

[1] https://www.vidarholen.net/contents/interjections/ and https://surveyanyplace.com/s/interjectionspdf

Table 3. These valid tokens (known as dictionary words) were words that contained in the dictionaries used by this paper. The remaining of the 700 words were known as out-of-dictionary (OOD) words. OOD words were not registered in any of the dictionaries used by this paper. The distribution of the words is shown in Table 4. This paper ignored OOD words due their low percentage in annotated dataset.

Table 4. Distribution of processed words

Type of words	Number of words	Percentage
Dictionary words	6,165	89.80%
OOD words	700	10.20%
Total	**6,865**	**100.00%**

5.2. Analysis of Threshold Segregation

The presence of categorized words in a sentence plays a major role in determining whether the sentence is mono-lingual or code-switching. This paper put a threshold of word presence for the analyzed languages. The results of the rule-based technique which are shown in Fig.3, Fig.4 and Fig. 5 confirmed the selection of the threshold.

Fig.3 shows the distribution of *my*-labelled sentences based on the presence of Malay words. It is logical to consider Malay word presence in this analysis because the objective of this analysis is to categorize a sentence either as Malay or otherwise. The score of the Malay word presence were group in the interval of 10.00% in Fig.3. Based on the distribution of the sentences, this paper sets the threshold value for presence of Malay words between 91.00% to 100.00% from the total words in the sentence.

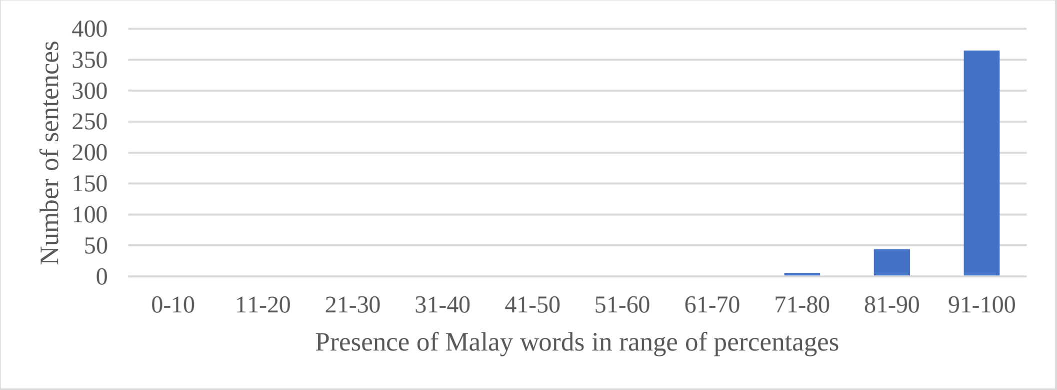

Fig.3 Distribution of *my*-labelled sentences based on the presence of Malay words

The *en*-labelled sentences have similar pattern of distribution as *my*-labelled sentences as shown in Fig.4. The threshold value for English word presence is between 91.00% to 100.00% due to high distribution shown in Fig.4.

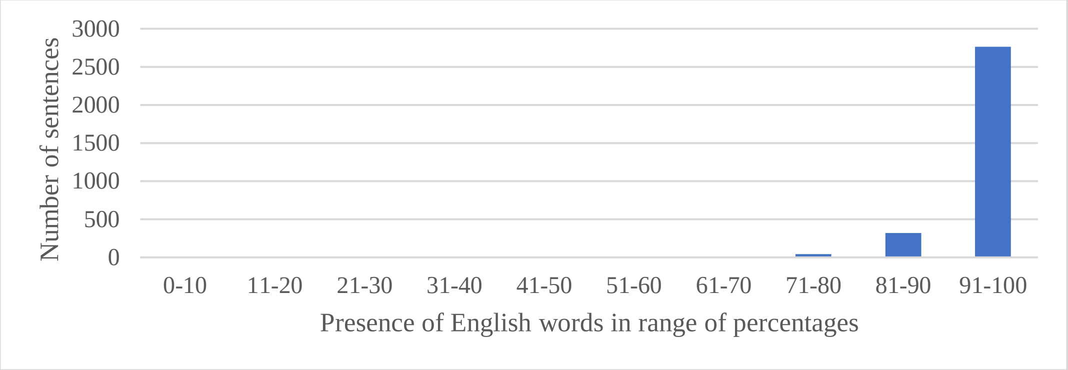

Fig.4 Distribution of *en*-labelled sentences based on the presence of English words

The value other than the said threshold value in this paper is used to categorize the sentence as MY-EN-CS sentences. A combination of two threshold values is used to categorize a MY-EN-CS. The combination is defined as the ratio of Malay words that is less than 90.00% and the presence of English word that is more than 10% or the presence of English words that is less than 90.00% and the presence of Malay word is more than 10.00%. The result in Fig. 5 confirmed this conclusion and the A number of sentences falls in the percentage of 91-100/0-10 Malay/English word presence is due to shared words. Shared words are the words that exists in Malay and English dictionaries. The flow of the segregator is shown in Fig.6.

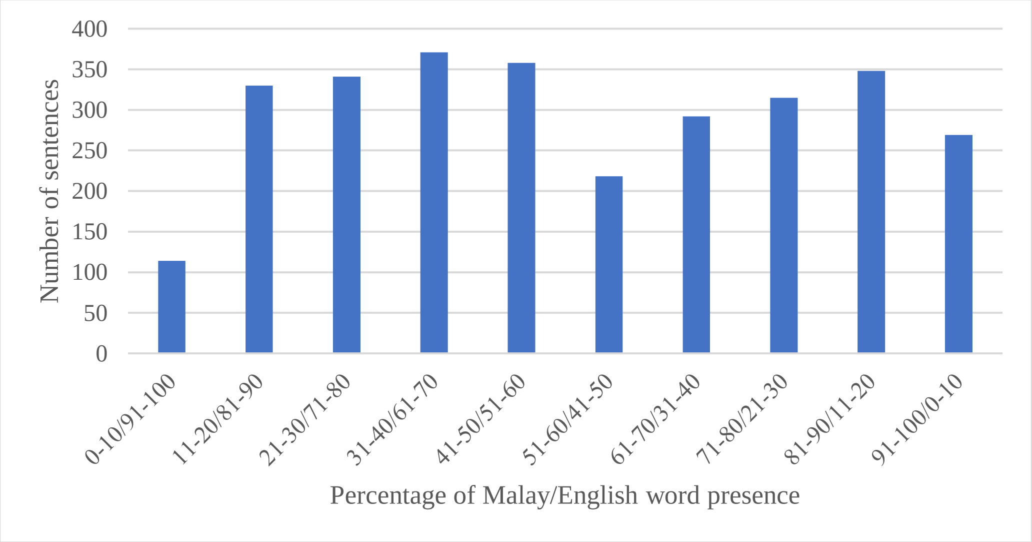

Fig. 5 Distribution of *my-en-cs*-labelled sentences according to the ratio of percentage of Malay and English word presence

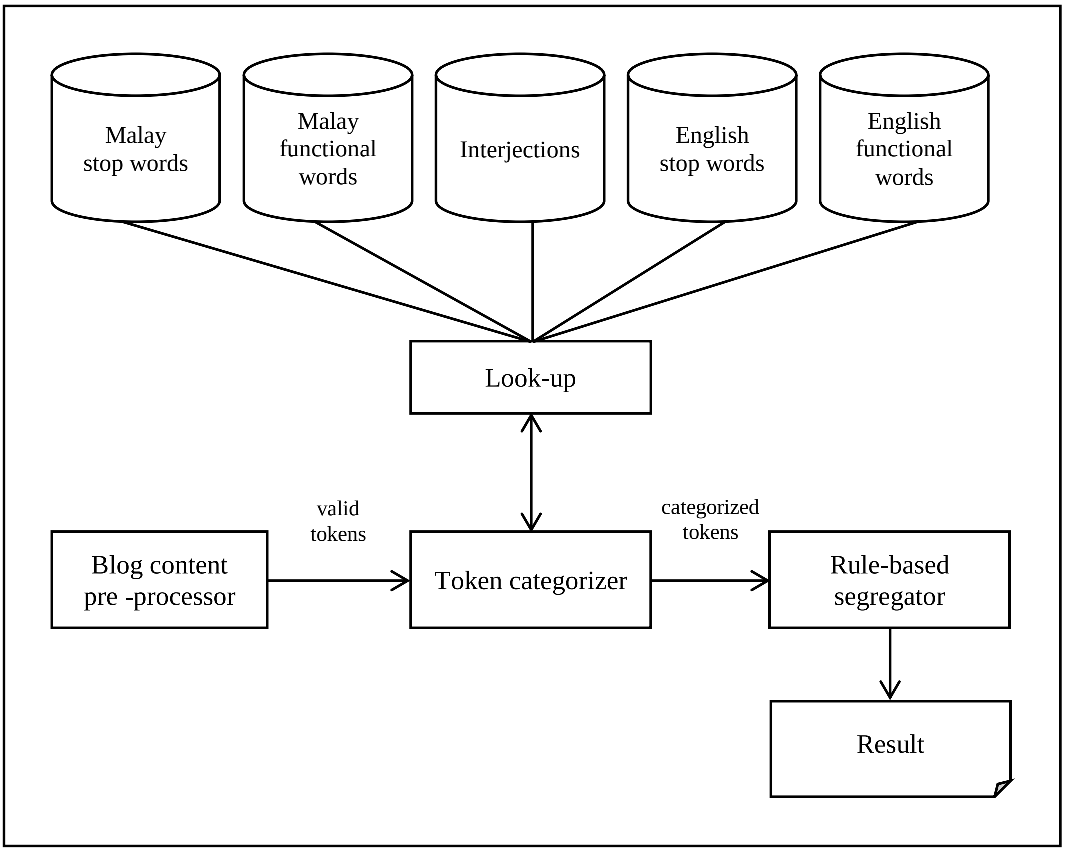

Fig.6 The flow of code-switching sentence segregator using rule-based technique

Based on the result, the rule-based segregator is defined as,

$$s = \begin{cases} t_{l_i l_j} & Pb(l_i) < 0.9 \; or \; Pb(l_j) < 0.9 \\ t_{l_i} & Pb(l_i) \geq 0.9 \end{cases}$$

where $i \neq j$ and i, j are integers.

6 Result

This paper measured the performance of the proposed rule-based technique using precision, recall, accuracy and F1. These measurements are commonly used in many text processing experiments. The metrics to calculate these measurements are extracted from three different confusion matrices, where each matrix is for different language labels. Four types of metrics are used plot the data into the confusion matrix as shown in Fig. 7. The result from the rule-based segregator were plotted into this matrix. The performance measurements are described in Table 6.

		Predicted Label	
		Analyzed language	**Others**
Actual Label	**Analyzed language**	True-Positive	False-Negative
	Others	False-Positive	True-Negative

Fig. 7 Confusion matrix

Table 5. Definition of performance measurement

Measurement	Definition	Model
Precision	The proportion of positive sentences that was correctly identified by the segregator	$Precision = \frac{True\ Postive}{True\ Positive + False\ Positive}$
Recall	The proportion of actual positive sentences that was correctly identified by the segregator	$Recall = \frac{True\ Positive}{True\ Postive + False\ Negative}$
Accuracy	The proportion of sentences that was correctly identify by the segregator	$Accuracy = \frac{True\ Postive + True\ Negative}{Total\ Data}$
F1	The weighted average of the recall and precision.	$F1 = \frac{2 * Recall * Precision}{Recall + Precision}$

The rule-based technique performed well with accuracy of 88.11% for *my-en-cs*-labelled, 93.89% for *my*-labelled and 94.19% for *en*-labelled sentences. In terms of precision, *my*-labelled sentences performed satisfactorily with 50.99% as compared to other labelled sentences. This is due to the low number of *my*-labelled sentences as compared to other labelled sentences in the dataset. The precision of *my*-labelled can be improved by adding more *my*-labelled sentences to the dataset. Other metrics of performances are shown in Table 6.

Table 6. Performance of rule-based technique

Language	Precision (%)	Recall (%)	Accuracy (%)	F1(%)
my	50.99	93.49	93.89	51.42
en	93.19	94.80	94.19	93.99
my-en-cs	92.72	80.34	88.11	86.09

7 Conclusion and Future Works

Code-switching sentence contains a mixture of words from two or more languages. Code-switching issue has been studied since 1970s by the linguists. The issue of code-switching is considered new in NLP. Currently, the availability and the accessibility to the code-switching data has made it possible for the NLP to address the issue of code-switching. The existing mono-lingual NLP systems are not suitable to process code-switching data because mono-lingual NLP system deals with only one language system. The results produce from such system could be misleading for code-switching data. Therefore, different NLP systems that process the presence of more than one language in a sentence is needed.

Code-switching sentences need to be segregated from mono-lingual sentences due different traits posed by these sentences. This paper proposed a rule-based technique to segregate these sentences. The proposed rule-based technique used ratio of word presence in a sentence with the threshold of 90.00% for mono-lingual and a combination of ratios for code-switching. The rule-based technique performed well with accuracy of more than 87.00%. The technique has successfully segregated code-switching sentences from mono-lingual sentences. The advantage of the rule-base technique is it does not require a huge amount of data as compared to learning-based technique. However, additional rules need to be added to the existing rules if the system needs to deal with three or more languages, which eventually will increase the complexity of the systems.

The segregation of code-switching and mono-lingual sentences is also usable in other text classification problem such as subjectivity classification, polarity classification, sarcasm detection and emotional classification. Even though this paper has viewed proposed approach as has performed well, this paper observed that the segregation approach needs to be experimented with learning-based techniques, which will be the future work of this paper. Furthermore, more studies on shared words are needed to improve the accuracy of code-switching results. The proposed segregation approach also needs to be experimented with other mixture of languages to ensure language independence.

References

[1] Huang, F., & Yates, A. (2014). Improving Word Alignment Using Linguistic Code Switching Data. In *EACL*, (pp. 1–9).

[2] Li, Q., Chen, Y. P., Myaeng, S.H., Jin, Y., & Kang, B.Y. (2009). Concept Unification of Terms in Different Languages via Web Mining for Information Retrieval. *Information Process & Management, 45*(2), 246–262.

[3] Li, Y., & Fung, P. (2012). Code-Switch Language Model with Inversion Constraints for Mixed Language Speech Recognition. In *24th International Conference of Computer Linguistic,* (pp. 1671–1680).

[4] Adel, H., Vu, N. T., Kirchhoff, K., Telaar, D., & Schultz, T. (2015). Syntactic and Semantic Features for Code-Switching Factored Language Models. *IEEE Transaction Audio, Speech & Language Processing, 23*(3), 431–440.

[5] Mukund, S., & Srihari, R. K. (2012). Analyzing Urdu Social Media for Sentiments Using Transfer Learning with Controlled Translations. In *Second Workshop on Language in Social Media,* (pp. 1–8).

[6] Vilares, D., Alonso, M. A., & Gómez-Rodrıguez, C. (2015). Sentiment Analysis on Monolingual, Multilingual and Code-Switching Twitter Corpora. In *6th Workshop on Computational Approaches to Subjectivity, Sentiment And Social Media Analysis 2015,* (pp. 2).

[7] Wang, Z., Lee, S. Y. M., Li, S., & Zhou, G. (2017). Emotion Analysis in Code-Switching Text With Joint Factor Graph Model. *IEEE/ACM Transaction Audio, Speech and Language Processing*, *25*(3), 469–480.

[8] Lo, S. L., Chiong, R., & Cornforth, D. (2017). An Unsupervised Multilingual Approach for Online Social Media Topic Identification. *Expert Systems with Application.*, 81, 282–298.

[9] Shrestha, P. (2012). Incremental N-gram Approach for Language Identification in Code-Switched Text. In *First Workshop on Computational Approaches to Code Switching*.

[10] King, B., & Abney, S. P. (2013). Labeling the Languages of Words in Mixed-Language Documents using Weakly Supervised Methods. In *2013 Conference of North American Chapter of the Association for Computational Linguistic: Human Language Technology,* (pp. 1110–1119).

[11] Barman, U., Das, A., Wagner, J. & Foster, J. (2014). Code Mixing : A Challenge for Language Identification in the Language of Social Media. In *First Workshop on Computational Approaches to Code Switching*.

[12] King, L., Baucom, E., Gilmanov, T., Sandra, K., Maier, W. & P. Rodrigues. (2014). The IUCL + System : Word-Level Language Identification via Extended Markov Models. In *First Workshop on Computational Approaches to Code Switching,* (pp. 102–106).

[13] Lui, M., Lau, J. H., & Baldwin, T. (2014). Automatic Detection and Language Identification of Multilingual Documents. *Transaction of the*

Association Computational Linguistics, 2, 27–40.

[14] Jain, N., LTRC I. H., & Bhat, R. A. (2014). Language Identification in Code-Switching Scenario. In *2014 Empirical Method in Natural Language Processing,* (pp. 87).

[15] Barman, U., Wagner, J., Chrupa, G. & Foster, J. (2014). DCU-UVT : Word-Level Language Classification with Code-Mixed Data. In *First Workshop on Computational Approaches to Code Switching* (pp. 127–132).

[16] Solorio, T. & Liu, Y. (2008). Learning to Predict Code-Switching points. In *Conference on Empirical Methods in Natural Language Processing* (pp. 973–981).

[17] Oco, N. & Roxas, R. E. O. (2012). Pattern Matching Refinements to Dictionary-Based Code-Switching Point Detection. In *26th Pacific Asia Conference on Language, Information and Computation,* (pp. 229–236).

[18] Li, Y., Yu, Y., & Fung, P. (2012). A Mandarin-English Code-Switching Corpus. In *Language Resources and Evaluation 2012*, (pp. 2515–2519).

[19] Papalexakis, E. E. (2014). Predicting Code-Switching in Multilingual Communication for Immigrant Communities. In *First Workshop of Computing Approaches to Code Switch,* (pp. 42–50).

[20] Cotterell, R., Renduchintala, A., Saphra, N., & Callison-Burch, C. (2014). An Algerian Arabic-French Code-Switched Corpus. In *Workshop on Free/Open-Source Arabic Corpora and Corpora Processing Tools Workshop Programme,* (pp. 34).

[21] Labutov, I., & Lipson, H. (2014). Generating Code-switched Text for Lexical Learning. In *52nd Annual Meeting of the Association for Computational Linguistic,* (pp. 562–571).

[22] Nguyen, D., & Cornips, L. (2016). Automatic Detection of Intra-Word Code-Switching. In *14th Annual SIGMORPHON Workshop on Computational Research in Phonetics, Phonology, and Morphology,* (pp. 82–86).

[23] Joshi, A. K. (1982). Processing of Sentences with Intra-Sentential Code-Switching. In *9th conference on Computational Linguistics,* (pp. 145–150).

[24] Carpuat, M. (2014). Mixed-Language and Code-Switching in the Canadian Hansard. In *First Workshop on Computational Approaches to Code Switching,* (pp. 107–115).

[25] Chekima, K., & Alfred, R. (2016). An Automatic Construction of Malay Stop Words Based on Aggregation Method. *Soft Computing in Data Science*, *652*, 180–189.

[26] Noor, N. H. B. M., Sapuan, S., & Bond, F. (2011). Creating the Open

Wordnet Bahasa. In *25th Pacific Asia Conference on Language, Information and Computation* (pp. 255–264).