ASTES

Advances in Science, Technology and Engineering Systems Journal
Vol. 5, No. 6, 363-369 (2020)
www.astesj.com
Special Issue on Multidisciplinary Sciences and Engineering

ASTES Journal
ISSN: 2415-6698

Vietnamese Text Classification with TextRank and Jaccard Similarity Coefficient

Hao Tuan Huynh[1], Nghia Duong-Trung[2], Dinh Quoc Truong[1], Hiep Xuan Huynh*[1]

[1]*College of Information & Communication Technology, Can Tho University, Can Tho city, 94000, Vietnam*

[2]*Software Engineering Department, FPT University, Can Tho city, 94000, Vietnam*

ARTICLE INFO

Article history:
Received: 03 September, 2020
Accepted: 28 October, 2020
Online: 20 November, 2020

Keywords:
TextRank
Vietnamese
Jaccard Similarity
Representative Keywords
Text Classification

ABSTRACT

Text classification is considered one of the most fundamental and essential problems that deal with automatically classifying textual resources into pre-defined categories. Numerous algorithms, datasets, and evaluation measurements have been proposed to address the task. Within the era of information redundancy, it is challenging and time-consuming to engineering a sizable amount of data in multi-languages manually. However, it is time-consuming to consider all words in a text, but rather several key tokens. In this work, the authors proposed an effective method to classify Vietnamese texts leveraging the TextRank algorithm and Jaccard similarity coefficient. TextRank ranks words and sentences according to their contribution value and extracts the most representative keywords. First, we collected textual sources from a wide range of Vietnamese news websites. We then applied data preprocessing, extracted keywords by TextRank algorithm, measured similarity score by Jaccard distance and predicted categories. The authors have conducted numerous experiments, and the proposed method has achieved an accuracy of 90.07% on real-world datasets. We have proved that it is entirely applicable in practice.

1 Introduction

This paper is an extension of work initially presented in IEEE RIVF 2020 [1] as an invited paper.

Within the era of information redundancy, it is challenging and time-consuming to manually engineer a sizable amount of multilingual data. For example, an electronic library might quickly identify documents for archiving and management. Besides, it is often less accurate and overwhelming if humans are involved in the process. Therefore, apply machine approaches to automate the procedure of text classification is mandatory. It makes the classified results more reliable and less subjective. It helps alleviate human involvement and information overload and enhances knowledge retrieval efficiency. There are many text classification studies such as, e.g. Bayes [2], decision trees [3], K-nearest neighbor [4], and neural network [5]. The literature shows that automated text classification is one mainstream research in natural language processing [6, 7]. Many research papers have been conducted to solve such problems as email-messages filtering [8], topic modeling [9], geo-localization [10], and document categorization [11]. The flowchart of standard text classification is presented in Figure 1.

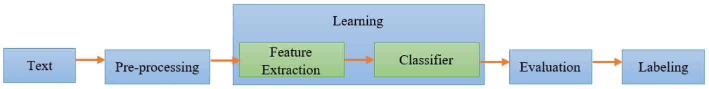

Figure 1: The flowchart of regular text classification where pre-processing and featuring perform an essential role.

The curse of dimensionality affects accuracy and computing time that arise when analyzing and organizing textual data in high-dimensional spaces [9]. However, machine learning models do not need to learn all tokens in the texts to categorize them. Instead, the text's label can be identified through its necessary tokens, which contribute the most to the text's meaning. Consequently, if we can extract the text source's main keywords, we might accurately classify it into an appropriate topic. The TextRank algorithm [12, 13] allows extracting the list of representative keywords of textual contents. On that consideration, the authors propose the method of automatic Vietnamese text classification based on the representative keyword analysis of the text. Textual datasets have been downloaded from several news websites with 15 main topics. Then, keyword sets that represent each topic are built. The system will extract that text's specific keywords when it is necessary to identify the topic

[*] Corresponding Author: Hiep Xuan Huynh, College of Information & Communication Technology, Can Tho University, Can Tho city, 94000, Vietnam, Email: hxhiep@ctu.edu.vn

https://dx.doi.org/10.25046/aj050644

of a text. The class of the text will be determined by computing the coverage of the representative keywords. It would be the right way for text classification applications in areas such as electronic library management. Although many Vietnamese documents are electronically available, no previous research has applied keywords-based text classification to them.

This work portrays our endeavor to construct a practical framework for addressing Vietnamese texts' classification task on the news. To the best of our knowledge, in this extended article, the authors have made several contributions. First, we discuss the paper [1] in a lot more detail by extending the related works and technical background. Second, we extend the experiments by investigating more scenarios.

2 Related Work

An early effort to address the task of Vietnamese text classification was conducted more than a decade ago [14]. In that paper, the authors solved automatically categorizing the problem, given textual sources into predefined categories. A comparison between statistical N-Gram language modeling and bag of words approaches has been investigated on their collected dataset. Several researchers have applied the idea of spam filtering into Vietnamese text sources [15]. Short messages such as conversational texts have also been exploited by addressing the task of suggestion intents [16]. The authors proposed a user suggestion intent definition in general from conversational texts at a functional segment unit. The task of automatic text categorization has been studied by comparing the performance of several term weighting schemes rather than analyzing the actual classification task [17]. Regarding Vietnamese sources, full-text representation has been exploited by many other research papers [18, 19, 20]. Thoughtfully learning the literature, we could claim that this is the first attempt to featuring Vietnamese texts using the idea of representative keywords.

3 Proposed System

3.1 Design Concept

The Vietnamese text classification system proposed by the authors includes two main components. One is the keyword extraction module, and the other is the comparison with the training set to identify the topic of a new document. We present the overall design of the system in Figure (**??**). The process begins with the text sources featured in the representative keyword vectors. For the training set, the vectors will be marked with the topic label. For the test data, the keyword extraction module is applied to convert the original text data into its keyword-based representation. Next, the system compares it to training data by calculating similarity scores. Finally, the prediction is assigned to the test data.

3.2 Text Pre-processing

Vietnamese is the only language in which every syllable is pronounced separately and is represented by a written word. This feature is evident in all aspects of phonetics, vocabulary and grammar. Data pre-processing is the first important step of any data mining process. It makes data in its original form easier to observe and explore. For the problem of text classification, due to specific characteristics, each language has its own challenges. The preprocessing process will help improve sorting efficiency and reduce the complexity of the training algorithm. Depending on the purpose of the classifier, we will have different preprocessing methods, such as

- Convert text to lowercase and correct spelling errors.
- Remove punctuation marks (if no sentence separation is performed).
- Remove special characters ([], [.], [,], [:], [”], [”], [;], [/], [[]], []̃, []́, [!], [@], [#], [$], [%], []̂, [&], [*], [(], [)]).
- Separate tokens by compound words (Vietnamese).
- Remove the stopwords, e.g. the words that appear most in the text that are not meaningful when participating in text classification. We utilize a list of 1942 Vietnamese stopwords[1] in our data processing.
- For the tokenization step, we utilize vnTokenizer [21] in our research. The comparison of tokenization accuracy achievable with different software is beyond the scope of this research paper.

3.3 Vietnamese Text Tokenization

Phonetic characteristics In Vietnamese, there is a special type of unit called ”tieng” or a sound of the thing. Phonetically, each ”tieng” is a syllable. For example, the word ”student” is translated into two syllables ”sinh vien” which are two separate words. As a result, these two words should come together to form a meaning token.

Vocabulary characteristics Each ”tieng”, in general, is a meaningful element. Continuing the previous example, the words ”sinh” and ”vien” have their own meaning when coming alone. But when they come together to form a single word ”sinh vien”, it has the meaning of student as in English. The vocabulary of Vietnamese is based on single words (one syllable) and the countless combination of them. Creating new words is very easy and flexible. If we pronounce a sound, we could write it down as a word.

Grammatical characteristics The Vietnamese words do not change morphology. For example, verbs in Vietnamese do not have -ed, -s, -ing forms. This feature will dominate other grammatical characteristics. When words combine with words into sentences, it is important to know the word order, the word phrase, and keywords for tenses recognition. Sorting words in a certain order is one of the main processes to express syntax relations.

[1] https://github.com/stopwords/vietnamese-stopwords/blob/master/vietnamese-stopwords.txt

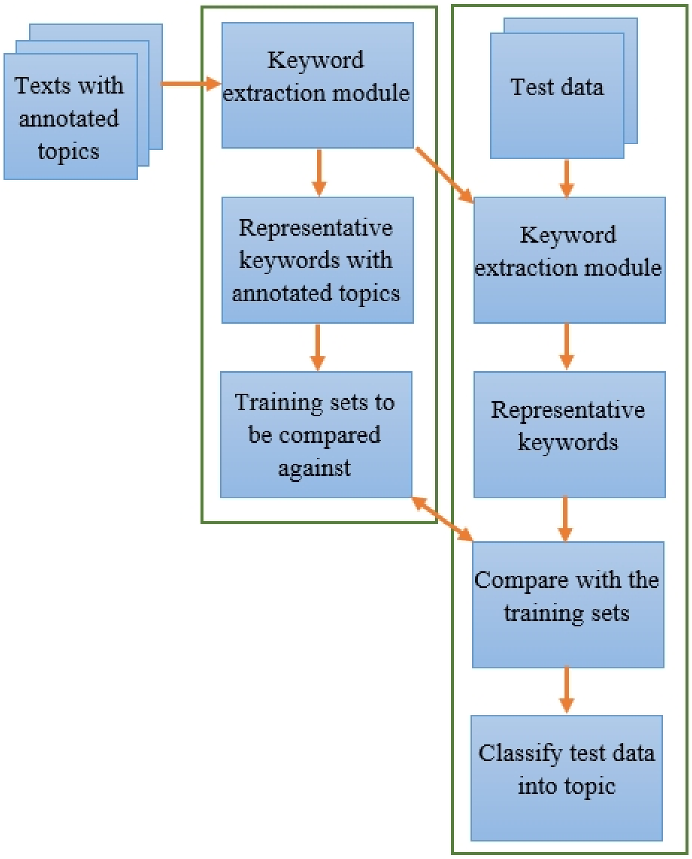
Texts with annotated topics
Keyword extraction module
Representative keywords with annotated topics
Training sets to be compared against
Test data
Keyword extraction module
Representative keywords
Compare with the training sets
Classify test data into topic

Figure 2: Our proposed Vietnamese text classification procedure.

3.4 Keyword Extraction by TextRank

Algorithm TextRank [22] was developed based on the main idea of the PageRank algorithm that Google search engine uses to rank website cite zhou2019chip, langville2008google, berkhout2016google. The bottom line of the TextRank algorithm is to use graphs to represent text and to score important information about the structure in which the text is represented by keywords. In other words, the TextRank algorithm processes a group of keywords representing the entire text. TextRank ranks words by their importance, arranges them in descending order of computed value, and extracts the most important words. The number of important words is a hyperparameter which is determined by the user prior to the TextRank algorithm execution. This algorithm is successfully applied for keyword extraction based on key value from a single text and this is also the advantage of TextRank.

The TextRank algorithm represents the textual source as a graph $G = (V, E)$ where V is the set of vertices and E is the set of edges in the graph. E is derived from a subset of $V\ x\ V$. Each vertex of the graph G corresponds to one word extracted from the text. An edge between any two vertices is created when their words appear in the text at any position between 2 and N. The value for the importance of the vertex V_i is calculated using the following formula:

$$S(V_i) = (1-d) + d * \sum_{j\in \mathrm{In}(V_i)} \frac{1}{|\mathrm{Out}(V_j)|} S(V_j)\,, \qquad (1)$$

where $d = [0, 1]$. In our experiments, d is set to 0.85 by default [22]. $\mathrm{In}(V_i)$ is the collection of vertices that point to it, and $\mathrm{Out}(V_i)$ is the collection of vertices that vertex V_i points to it. The TextRank algorithm can be presented in the Algorithm (1).

Algorithm 1 Implementation of TextRank algorithm.

Input: Textual data
Output: Extracted keywords K

```
1: Build the graph $G = (V, E)$
2: Compute the edges' importance score by Equation (1)
3: Sort the edges by their scores
4: Select top $K$ edges
```

3.5 Keyword Extraction for Topics

At this stage, we build a sample set of keywords for each topic from a set of keywords extracted from the subject-labeled texts in the training set. The model of building a set of keyword samples for topics can be summarized as Figure (3). The keywords of the topic are calculated by the statistical method of the number of occurrences of each word in the list of keyword sets of training text. Words that are keywords of one topic cannot be keywords of another topic.

3.6 Similarity Measurement by Jaccard Distance

Mathematically, there are many ways to calculate the similarity between any two keyword lists R_i and any R_j, provided they are of the same length. However, in the context of the similarity between two documents, we do not need to include all words in the text but only the representative keywords T. The number of keywords represented will be a lot less than the entire word in the text. Then, a weighted version of the Jaccard [23] distance is determined as follows:

$$\mathrm{Jaccard}(R_i, R_j) = \frac{1}{T} \sum_{d=1}^{T} \frac{|R_{i,d} \cap R_{j,d}|}{|R_{i,d} \cup R_{j,d}|}, \qquad (2)$$

4 Experiments

4.1 Datasets

Data are collected from highly reputable Vietnamese websites. We used Teleport pro [2] software for automated data collection. The downloaded data is converted to plain text file and saved to the corresponding folders with the folder name as predefined theme name. Specifically, data is downloaded from the website [3] [4] [5] [6] with 15 main topics. These topics are summarized in Table (1). The

[2] http://tenmax.com
[3] http://vnexpress.net
[4] http://dantri.vn
[5] http://tuoitre.vn
[6] http://yahoo.com.vn

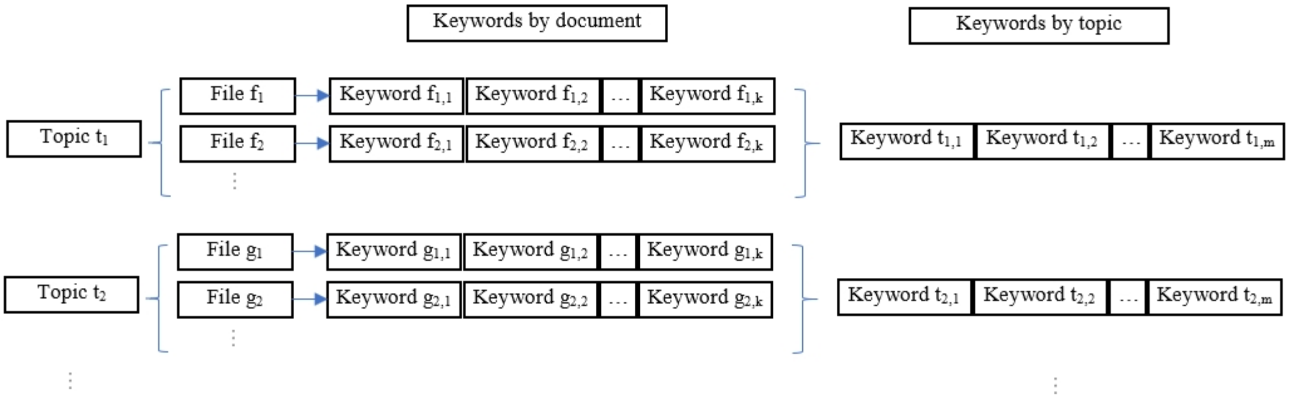

Figure 3: The keyword extraction for topics.

collected data was distributed into 80% training data group and 20% test data group.

Table 1: Summary of 15 topics

#	Topic	Description
1	Music	Singers, songs, musicians, performers, audiences.
2	Cuisine	Food, restaurants, cooks, menus.
3	Movies	Cinemas, genres, actors/actresses.
4	Tourism	tourist destinations, resorts, attractions, hotels.
5	Family	Marriage, family, grandpa, father, mother.
6	Education	Education and training, enrollment, students.
7	Science	Research, inventions, scientific discoveries.
8	Business	market, buying and selling, gold prices, currencies.
9	Beauty	beauty salon, care, therapies.
10	Motorcycle	motorbikes, cars, prices, comparison, repair.
11	Law	Legal, criminal, police events.
12	Health	health information, hospitals.
13	Sports	soccer matches, games, scores, comments.
14	Fashion	Field of fashion, costumes, designers, stylists.
15	Computers	Digital technology, computers, operating systems.

4.2 Evaluation Metric

We define the number of true positive (TP), true negative (TN), false positive (FP), and false negative (FN). We also define m_+ the total of condition positives, m_- the total of condition negatives, $\hat{m}_+$ the total predicted condition positives, $\hat{m}_-$ the total predicted condition negatives, and m the total population. Then, we compute the sensitivity or recall by using:

$$\text{recall} = \frac{\text{TP}}{m_+} . \tag{3}$$

We compute precision as follows:

$$\text{precision} = \frac{\text{TP}}{\hat{m}_+} . \tag{4}$$

Then, we compute F1-score as follows:

$$\text{F1-score} = 2\frac{\text{recall} \cdot \text{precision}}{\text{recall} + \text{precision}} . \tag{5}$$

Table 2: Statistics on all experimental scenarios.

# Keywords	Accuracy (%)	Training time (s)	Test time (s)
10	75.27	2155	1111
15	82.47	2156	1116
20	85.87	2162	1114
25	87.40	2168	1115
30	88.13	2178	1116
35	88.00	2177	1118
40	89.40	2184	1119
45	89.90	2191	1119
50	90.07	2199	1120
55	90.07	2205	1112
60	90.03	2213	1123
65	90.06	2219	1127
70	90.07	2307	1125
all	**95.40**	9279	1362

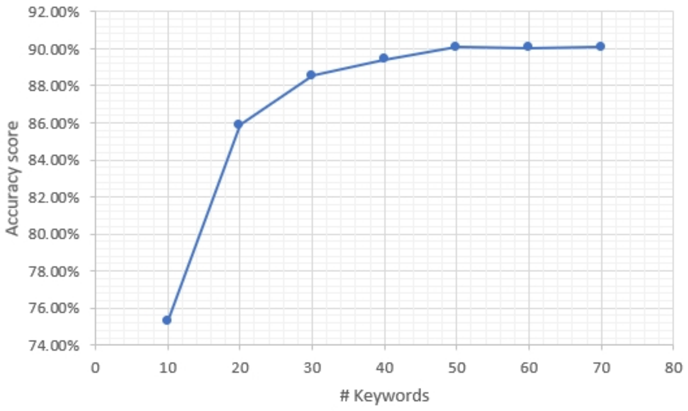

Figure 4: The correlation between the number of keywords and accuracy score.

4.3 Experimental Results

In the experiments, the authors have designed several scenarios where the number of keywords is varied to test the model's performance. The number of keywords $T = \{10, 20, 30, 40, 50, 60, 70\}$ and in an extreme scenario where all the keywords selected by the TextRank algorithm are used. The experimental results are presented in Table (3), Table (4), Table (5), Table (6), Table (7), Table (8), Table (9), and Table (10) respectively.

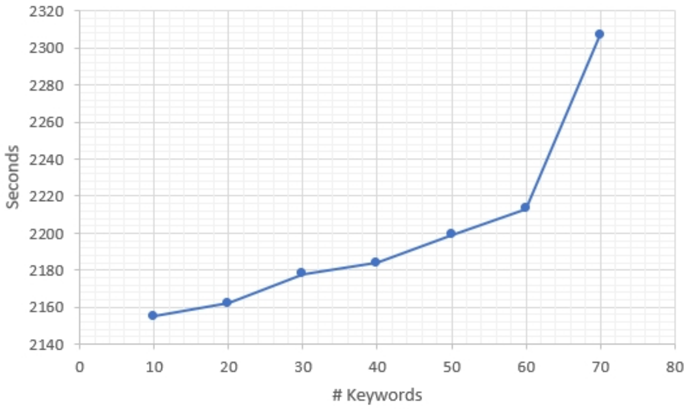

Figure 5: The correlation between the number of keywords and training time.

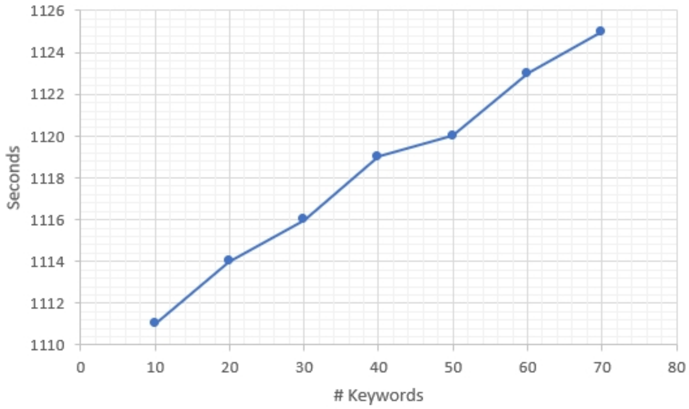

Figure 6: The correlation between the number of keywords and test time.

Table 3: Test set's confusion matrix with 10 keywords selected by the TextRank algorithm.

Topic ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Accuracy (%)
1	**87**	0	4	0	7	0	0	0	0	0	1	0	0	1	0	87
2	0	**73**	0	2	6	0	11	2	4	0	0	0	0	2	0	73
3	6	0	**93**	0	1	0	0	0	0	0	0	0	0	0	0	93
4	5	2	3	**60**	14	0	0	6	2	0	2	1	2	3	0	60
5	4	2	2	0	**78**	2	0	2	0	0	6	2	0	0	2	78
6	4	2	0	0	2	**80**	0	0	0	0	10	0	2	0	0	80
7	4	2	3	5	9	5	**49**	0	7	2	0	1	3	1	9	49
8	4	1	0	2	7	0	2	**77**	0	2	0	1	2	0	2	77
9	0	1	1	0	2	0	4	1	**83**	0	0	6	2	0	0	83
10	0	0	0	1	0	0	0	4	1	**91**	1	0	1	1	0	91
11	0	1	1	1	7	0	1	6	0	0	**82**	1	0	0	0	82
12	4	0	0	3	9	1	5	0	6	1	5	**66**	0	0	0	66
13	2	1	0	1	18	1	0	0	1	0	0	0	**74**	1	1	74
14	6	0	10	2	2	0	2	2	3	4	1	0	1	**67**	0	67
15	2	0	2	0	1	0	1	10	3	10	0	0	0	2	**69**	69
Average accuracy																**75**

The authors also make statistics on all experimental scenarios in Table (2). Note that *all* in the Table means all the tokens in a particular text. The texts' length is varied, and the average number of tokens is 500. We plotted the correlation between the number of keywords and accuracy scores in Figure (4). The accuracy of 90% is stable at 50 keywords. In Figure (5), we observe a considerable increase from 60 to 70 keywords. While in Figure (6), the test time grows eventually.

Table 4: Test set's confusion matrix with 20 keywords selected by the TextRank algorithm.

Topic ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Accuracy (%)
1	**96**	0	1	0	3	0	0	0	0	0	0	0	0	0	0	96
2	1	**85**	0	1	5	0	3	2	0	1	1	0	0	1	0	85
3	0	0	**95**	1	3	0	0	0	1	0	0	0	0	0	0	95
4	0	0	1	**83**	5	0	0	10	0	0	0	0	0	1	0	83
5	0	0	0	2	**92**	0	0	0	0	0	6	0	0	0	0	92
6	0	0	2	0	7	**77**	0	2	0	0	10	0	0	0	2	77
7	0	0	0	3	5	2	**78**	3	3	2	0	2	1	0	1	78
8	0	0	2	1	2	0	2	**89**	0	1	2	0	0	0	1	89
9	0	0	0	0	4	0	2	1	**81**	3	0	7	0	2	0	81
10	1	0	0	0	0	0	1	1	0	**95**	0	0	1	1	0	95
11	1	0	0	0	8	0	0	5	0	0	**85**	1	0	0	0	85
12	0	1	0	2	14	2	5	2	6	1	3	**63**	0	0	1	63
13	0	0	0	0	4	0	0	0	0	0	0	0	**96**	0	0	96
14	5	0	3	1	3	0	0	0	1	1	0	0	0	**86**	0	86
15	1	0	2	0	0	0	2	3	0	5	0	0	0	0	**87**	87
Average accuracy																**86**

Table 5: Test set's confusion matrix with 30 keywords selected by the TextRank algorithm.

Topic ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Accuracy (%)
1	**95**	0	2	0	3	0	0	0	0	0	0	0	0	0	0	95
2	0	**91**	0	2	7	0	0	0	0	0	0	0	0	0	0	91
3	0	0	**95**	0	3	0	0	0	0	0	0	0	0	2	0	95
4	0	0	2	**84**	5	0	0	8	0	0	1	0	0	0	0	84
5	0	0	0	0	**94**	0	0	0	0	0	4	2	0	0	0	94
6	0	0	0	0	12	**74**	2	6	0	2	4	0	0	0	0	74
7	0	0	0	0	2	0	**88**	5	3	1	0	1	0	0	0	88
8	0	0	2	1	1	1	3	**91**	0	0	1	0	0	0	0	91
9	0	0	0	0	4	0	0	0	**88**	2	0	4	0	2	0	88
10	0	0	0	0	0	0	0	3	0	**96**	0	0	1	0	0	96
11	0	0	0	0	5	0	0	7	0	0	**88**	0	0	0	0	88
12	0	3	0	2	15	1	5	5	5	0	2	**62**	0	0	0	62
13	0	0	0	0	4	0	0	0	0	0	0	1	**95**	0	0	95
14	4	0	2	1	0	0	0	2	1	0	0	0	0	**90**	0	90
15	0	0	0	0	1	0	1	5	0	2	0	0	0	0	**91**	91
Average accuracy																**88**

Table 6: Test set's confusion matrix with 40 keywords selected by the TextRank algorithm.

Topic ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Accuracy (%)
1	**91**	0	1	0	7	0	0	0	0	0	0	0	0	1	0	91
2	0	**95**	0	2	3	0	0	0	0	0	0	0	0	0	0	95
3	0	0	**94**	1	3	0	0	0	0	2	0	0	0	0	0	94
4	0	0	1	**87**	1	0	0	11	0	0	0	0	0	0	0	87
5	0	0	0	0	**92**	2	0	0	0	0	4	2	0	0	0	92
6	0	0	0	0	12	**78**	2	2	0	0	6	0	0	0	0	78
7	0	0	1	1	0	0	**90**	4	2	1	0	1	0	0	0	90
8	0	0	1	0	2	0	0	**94**	0	1	2	0	0	0	0	94
9	0	0	0	1	3	0	0	0	**90**	1	0	2	0	3	0	90
10	0	0	0	0	0	0	0	4	1	**95**	0	0	0	0	0	95
11	0	0	1	1	5	0	0	3	0	0	**89**	0	0	0	1	89
12	0	3	0	3	14	0	3	11	5	0	0	**61**	0	0	0	61
13	0	0	0	0	4	0	0	0	0	0	0	0	**96**	0	0	96
14	2	0	1	0	0	0	0	2	0	0	0	0	1	**94**	0	94
15	0	0	0	0	1	1	0	2	0	1	0	0	0	0	**95**	95
Average accuracy																**89**

Table 7: Test set's confusion matrix with 50 keywords selected by the TextRank algorithm.

Topic ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Accuracy (%)
1	**94**	0	1	0	4	0	0	0	0	0	0	0	0	1	0	94
2	1	**96**	0	1	1	0	0	1	0	0	0	0	0	0	0	96
3	6	0	**87**	0	2	0	0	1	0	2	0	0	0	2	0	87
4	0	0	0	**89**	3	0	0	8	0	0	0	0	0	0	0	89
5	0	0	0	0	**96**	0	0	0	0	0	2	2	0	0	0	96
6	0	0	0	0	14	**82**	2	0	0	0	2	0	0	0	0	82
7	0	0	1	3	0	0	**87**	3	1	4	0	1	0	0	0	87
8	0	0	0	2	1	0	0	**95**	0	0	2	0	0	0	0	95
9	0	0	0	0	1	0	0	0	**90**	1	0	4	1	3	0	90
10	0	0	0	1	0	0	1	4	1	**93**	0	0	0	0	0	93
11	0	0	1	0	6	0	0	2	0	0	**91**	0	0	0	0	91
12	0	1	0	1	13	0	3	6	3	0	1	**72**	0	0	0	72
13	0	0	0	0	4	0	0	0	0	0	0	0	**96**	0	0	96
14	3	0	2	0	1	0	0	0	0	0	0	0	0	**94**	0	94
15	2	0	0	0	1	1	0	5	1	1	0	0	0	0	**89**	89
Average accuracy																**90**

Table 8: Test set's confusion matrix with 60 keywords selected by the TextRank algorithm.

Topic ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Accuracy (%)
1	**93**	0	1	0	4	0	0	1	0	0	0	0	0	1	0	93
2	0	**94**	0	2	4	0	0	0	0	0	0	0	0	0	0	94
3	2	0	**89**	2	4	0	0	1	0	0	0	0	0	2	0	89
4	0	0	0	**87**	3	0	0	10	0	0	0	0	0	0	0	87
5	0	0	0	0	**94**	2	0	0	0	0	4	0	0	0	0	94
6	0	0	1	1	12	**74**	2	2	0	0	8	0	0	0	0	74
7	0	0	1	0	0	0	**87**	5	0	6	0	1	0	0	0	87
8	0	0	2	2	1	0	0	**93**	0	0	2	0	0	0	0	93
9	0	1	0	1	0	0	0	0	**93**	1	0	2	0	2	0	93
10	0	0	0	1	0	0	1	3	0	**95**	0	0	0	0	0	95
11	0	0	0	1	1	0	0	3	0	0	**95**	0	0	0	0	95
12	0	1	0	2	7	0	5	6	4	0	3	**72**	0	0	0	72
13	0	0	0	0	4	0	0	0	0	0	0	1	**95**	0	0	95
14	3	0	1	0	0	0	0	2	0	0	0	0	0	**94**	0	94
15	0	0	0	0	2	0	0	2	0	1	0	0	0	0	**95**	95
Average accuracy																**90**

Table 9: Test set's confusion matrix with 70 keywords selected by the TextRank algorithm.

Topic ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Accuracy (%)
1	**94**	0	1	0	2	0	1	1	0	0	0	0	0	1	0	94
2	0	**93**	0	1	3	0		0	0	0	0	2	0	1	0	93
3	0	0	**87**	2	6	0	0	1	0	0	0	0	0	4	0	87
4	0	0	0	**94**	0	0	0	6	0	0	0	0	0	0	0	94
5	0	0	0	0	**94**	0	0	0	0	0	4	2	0	0	0	94
6	0	0	0	3	10	**75**	2	4	0	0	6	0	0	0	0	75
7	0	0	0	0	0	0	**91**	5	0	4	0	0	0	0	0	91
8	0	0	1	1	1	0	0	**95**	0	0	2	0	0	0	0	95
9	0	0	0	1	3	0	0	0	**90**	1	0	2	0	3	0	90
10	0	0	0	1	0	0	1	4	0	**94**	0	0	0	0	0	94
11	0	1	0	0	1	0	0	2	0	0	**95**	1	0	0	0	95
12	0	1	0	0	10	1	3	6	8	0	4	**67**	0	0	0	67
13	0	0	0	0	4	0	0	0	0	0	0	1	**95**	0	0	95
14	2	0	2	0	0	0	1	1	0	0	0	0	0	**94**	0	94
15	0	0	0	0	1	0	0	4	1	0	0	0	0	1	**93**	92
Average accuracy																**90**

5 Conclusion

In this article, the authors have described a proposed approach that allows text classification based on the solution of extracting specific representative keywords of the text. We discussed the proposed system in detail from the abstract design, text pre-processing, and Vietnamese characteristics. Then we described the TextRank algorithm based on graphs to score important information about the text's structure. Intensive experiments have been conducted to prove the stability and robustness of the proposed system. High accuracy of 90.07% has been achieved. Although many Vietnamese documents are electronically available, this is the first to conduct text classification based on keywords. This research portrays our endeavor to construct a practical framework for addressing Vietnamese texts' classification tasks on news websites.

Table 10: Test set's confusion matrix with all keywords selected by the TextRank algorithm.

Topic ID	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Accuracy (%)
1	**92**	0	0	0	8	0	0	0	0	0	0	0	0	0	0	92
2	0	**92**	0	2	6	0	0	0	0	0	0	0	0	0	0	92
3	0	0	**86**	2	12	0	0	0	0	0	0	0	0	0	0	86
4	0	0	0	**96**	2	0	0	2	0	0	0	0	0	0	0	96
5	0	0	0	0	**99**	0	0	0	0	0	1	0	0	0	0	99
6	0	0	0	0	16	**84**	0	0	0	0	0	0	0	0	0	84
7	0	0	0	0	0	0	**99**	1	0	0	0	0	0	0	0	99
8	0	0	1	0	0	0	0	**99**	0	0	0	0	0	0	0	99
9	0	0	0	0	1	0	0	0	**99**	0	0	0	0	0	0	99
10	0	0	0	1	1	0	1	0	0	**97**	0	0	0	0	0	97
11	0	0	0	0	1	0	0	0	0	0	**99**	0	0	0	0	99
12	0	0	0	0	2	0	0	0	1	0	0	**97**	0	0	0	97
13	0	0	0	0	4	0	0	0	0	0	0	0	**96**	0	0	96
14	0	0	1	0	0	0	0	0	0	0	0	0	0	**99**	0	99
15	0	0	0	0	1	0	1	0	0	0	0	0	0	0	**98**	98
Average accuracy																**95**

References

[1] H. T. Huynh, N. Duong-Trung, X. S. Ha, N. Q. T. Tang, H. X. Huynh, D. Q. Truong, "Automatic Keywords-based Classification of Vietnamese Texts," in 2020 RIVF International Conference on Computing and Communication Technologies (RIVF), 1–3, IEEE, 2020, doi:10.1109/RIVF48685.2020.9140761.

[2] L. Zhang, L. Jiang, C. Li, G. Kong, "Two feature weighting approaches for naive Bayes text classifiers," Knowledge-Based Systems, **100**, 137–144, 2016, doi:10.1016/j.knosys.2016.02.017.

[3] C. C. Aggarwal, Data classification: algorithms and applications, CRC press, 2014.

[4] S. Jiang, G. Pang, M. Wu, L. Kuang, "An improved K-nearest-neighbor algorithm for text categorization," Expert Systems with Applications, **39**(1), 1503–1509, 2012, doi:10.1016/j.eswa.2011.08.040.

[5] J. Nam, J. Kim, E. L. Mencía, I. Gurevych, J. Fürnkranz, "Large-scale multi-label text classification—revisiting neural networks," in Joint european conference on machine learning and knowledge discovery in databases, 437–452, Springer, 2014, doi:10.1007/978-3-662-44851-9_28.

[6] V. B. Kobayashi, S. T. Mol, H. A. Berkers, G. Kismihók, D. N. Den Hartog, "Text classification for organizational researchers: A tutorial," Organizational research methods, **21**(3), 766–799, 2018, doi:10.1177/1094428117719322.

[7] N. Duong-Trung, Social Media Learning: Novel Text Analytics for Geolocation and Topic Modeling, Cuvillier Verlag, 2017.

[8] G. Xu, J. Qi, D. Huang, M. Daneshmand, "Detecting spammers on social networks based on a hybrid model," in 2016 IEEE International Conference on Big Data (Big Data), 3062–3068, IEEE, 2016, doi:10.1016/j.neucom.2015.02.047.

[9] N. Duong-Trung, L. Schmidt-Thieme, "On Discovering the Number of Document Topics via Conceptual Latent Space," in Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM '17, 2051–2054, Association for Computing Machinery, New York, NY, USA, 2017, doi:10.1145/3132847.3133086.

[10] N. Duong-Trung, N. Schilling, L. Schmidt-Thieme, "Near real-time geolocation prediction in twitter streams via matrix factorization based regression," in Proceedings of the 25th ACM international on conference on information and knowledge management, 1973–1976, ACM, 2016, doi: 10.1145/2983323.2983887.

[11] D. Kim, D. Seo, S. Cho, P. Kang, "Multi-co-training for document classification using various document representations: TF–IDF, LDA, and Doc2Vec," Information Sciences, **477**, 15–29, 2019, doi:10.1016/j.ins.2018.10.006.

[12] R. Niu, B. Shen, "Microblog User Interest Mining Based on Improved TextRank Model," Journal of Computers, **30**(1), 42–51, 2019, doi:10.3966/199115992019023001005.

[13] N. Akhtar, M. S. Beg, H. Javed, "TextRank enhanced Topic Model for Query focussed Text Summarization," in 2019 Twelfth International Conference on Contemporary Computing (IC3), 1–6, IEEE, 2019, doi:10.1109/IC3.2019.8844939.

[14] V. C. D. Hoang, D. Dinh, N. Le Nguyen, H. Q. Ngo, "A comparative study on vietnamese text classification methods," in 2007 IEEE International Conference on Research, Innovation and Vision for the Future, 267–273, IEEE, 2007, doi:10.1109/RIVF.2007.369167.

[15] T.-H. Pham, P. Le-Hong, "Content-based approach for Vietnamese spam SMS filtering," in 2016 International Conference on Asian Language Processing (IALP), 41–44, IEEE, 2016, doi:10.1109/IALP.2016.7875930.

[16] T.-L. Ngo, K. L. Pham, H. Takeda, S. B. Pham, X. H. Phan, "On the identification of suggestion intents from vietnamese conversational texts," in Proceedings of the Eighth International Symposium on Information and Communication Technology, 417–424, 2017, doi:10.1145/3155133.3155201.

[17] V. T. Nguyen, N. T. Hai, N. H. Nghia, T. D. Le, "A Term Weighting Scheme Approach for Vietnamese Text Classification," in International Conference on Future Data and Security Engineering, 46–53, Springer, 2015, doi:10.1007/978-3-319-26135-5_4.

[18] N. H. D. Tri, N. T. Quan, N. V. Tien, N. T. Hung, "Xay dung mo hinh phan tan cho phan lop khoi luong lon van ban theo chu de (in English: building distributed model for classification massive text data by topic)," PROCEEDING of Publishing House for Science and Technology, 2017, doi:10.15625/vap.2016.000104.

[19] B. K. Linh, N. T. T. Ha, N. T. N. Tu, D. T. Tinh, "Phan loai van ban tieng Viet dua tren mo hinh chu de (in English: vietnamese text classification based on topic modeling)," PROCEEDING of Publishing House for Science and Technology, 2017, doi:10.15625/vap.2016.00065.

[20] T. Ngoc Phuc, P. Tran Vu, P. Cong Xuyen, N. Vu Duy Quang, "Phan loai noi dung tai lieu Web tieng viet (in English: classification of vietnamese texts on the web)," Vietnam Journal of Science and Technology, **51**(6), 669–680, doi:10.15625/2525-2518/51/6/11629.

[21] N. T. M. Le Hong Phuong, A. R. Huyen, A. Ho Tuong Vinh, "Hybrid Approach to Word Segmentation of Vietnamese Texts," 2008, doi:10.1007/978-3-540-88282-4_23.

[22] R. Mihalcea, P. Tarau, "Textrank: Bringing order into text," in Proceedings of the 2004 conference on empirical methods in natural language processing, 404–411, 2004.

[23] G. Rebala, A. Ravi, S. Churiwala, An introduction to machine learning, Springer, 2019.