Text Classification: Naïve Bayes Classifier with Sentiment Lexicon

Cong-Cuong Le[1], P.W.C. Prasad*[1], Abeer Alsadoon[1], L. Pham[1], A. Elchouemi[2]

***Abstract*— This paper proposes a method of linguistic classification based on the analysis of positive, negative and neutral sentiments expressed within text written in Vietnamese and English. It includes a process for document preparation and is based on the development of training data using Naïve Bayes classification in conjunction with a sentiment lexicon dictionary, thus reducing the size of the training corpus and limitation of using bag-of-words. Naïve Bayes, a machine learning and information mining algorithm, was chosen for its proven viability and its central role in data retrieval in general. The effectiveness of Naïve Bayes is further enhanced through the use of the dictionary as the input source, reducing the magnitude of the training corpus and consequently training time. In addition, the implementation of a document preparation process significantly improves accuracy to 98.2 % when compared with traditional Naïve Bayes (96.1%) and the lexical method (87.3 %).**

***Index Terms*— Vietnamese; Sentiment lexicon; Naïve Bayes; Machine learning; Classification document; Probability; Preparation; Tokenization; Stop-word**

I. Introduction

Maintenance of national and international security continues to be a serious challenge for law enforcement bodies. Scrutiny of world-wide social media content is one way to identify developing security threats [1] through an analysis of sentiments expressed in online documents. Feldman [4] has stressed the urgent need for more research in this area despite the existence of in excess of 7,000 recent studies. Corporations are also responding to this need and statistical analysis packages, such as IBM's SPSS, now include modules devoted to opinion (sentiment) investigation [4]. Current sentiment analysis relies to a large extent on sentiment lexicons, composed of sets of seed words that are used for the extraction of domain related expressions [2]. Sentiment lexicons, therefore, play a highly significant role when determining the positive or negative value of words and phrases, sentiment analysis (SA), and, if scaled, are also vital for determining the strength of a sentiment [3]. Widely used sentiment lexicon databases are

AFINN, and SentiWordNet (as cited in Al-Rowaily, & Abulaish) [1], primarily developed for English and Arabic. For other languages, including Vietnamese, there is little evidence that existing sentiment lexicons are used in document classification processing. This highlights a need to develop such sentiment databases for other languages, among them Vietnamese.

Examining high volumes of information requires computer-based identification of the semantic origins of every term (t) in a record (d) together with consideration of connections (contextual terms) between t in d [5]. This requires a computer based automated program capable of classification at high rates of reliability and significant amounts of training data. The latter are often rare and costly. There is, therefore, a need for a method that does not require high volumes of training data, but has the ability to assign labels to take advantage of existing data.

This research intends to analyses current solutions for creating a sentiment lexicon for a specific domain followed by the identification of a method that can then be applied. In addition, a model for applying sentiment lexicons to the classification of textual data on the Internet and social media will be developed. The primary goal of this research is the construction of a model for seed word identification and for the compilation of sentiment lexicons that are related to a particular sentiment domain -not only for Vietnamese but also for other languages. The aim of this paper is to identify a suitable method that has the potential to produce the best outcomes. In addition, this research aims to develop a process whereby sentiment lexicons may be used as training data for Naïve Bayes classifier to analyze social media content aided by a document preparation process.

II. Related Work

This paper aims to provide solutions for identifying positive and negative sentiments in documents, based on document classification (lexical and machine learning approaches) and a sentiment dictionary. The lack of sentiment lexicons in Vietnamese and other will be overcome through the development of a simple method for sentiment lexicon generation. The work includes building such a Vietnamese sentiment lexicon (VSL).

A. Constructing a sentiment lexicon

Classification strategies are frequently based on lexicons containing words and phrases that have been identified as "positive "or "negative". This permits scoring of sentiments within a text, together with the identification of its strength (extremity). The overall sentiment expressed in a text is then evaluated based on the frequency of negative and positive words appearing in this context [7].

A significant amount of research has so far been devoted

Cong-Cuong Le is with the Charles Sturt University, Australia.

Abeer Alsadoon is a Senior Lecturer with the Charles Sturt University, Australia.

Linh Pham is a Lecturer with the Charles Sturt University, Australia.

P.W.C. Prasad* is an Associate Professor with the Charles Sturt University, 63, Level 1, Oxford Street, Darlinghurst, Australia (corresponding author) Phone: 61-40-2690369; e-mail: cwithana@studygroup.com.

A. Elchouemi is an Associate Professor with the Colorado State University - Global campus, USA.

to the creation of words from which assumptions may be drawn with negative or positive connotations. Strategies for producing assessment vocabulary fall into two primary classes - lexicon and corpus-based methodologies [5]. Paltoglou and Thelwall [8] provided a method for using a Term Frequency Times Inverse Document Frequency (TF-IDF) algorithm for generating scores based on words in sentiment lexicon databases.

Further studies proposed techniques that can be used to generate sentiment lexicons relating to specific domains. Yang and Lin [9] offered a solution by building a Chinese sentiment lexicon based on an improved semantic orientation pointwise mutual information (SO-PMI) algorithm, to classify hotel reviews. The work included a basic sentiment lexicon (BSL) as input from which w-fronted words were chosen to determine PointWise Mutual Information (PDPMI) to identify nondomain-specific words and their weight score, and then use the weight score to adjust future probability during training. Insights from this work are important for comparison purposes, although this lexicon is not publicly available.

A similar technique using a published sentiment lexicon with a domain corpus has also been applied in the research of Al-Rowaily, Abulaish [1]. The researchers selected words from a domain corpus, comparing these with words from four other basic sentiment lexicons and creating a new seed word database, based on a min-max normalisation formula. Al-Rowaily et al. [1] also developed a solution by creating an Arabic sentiment lexicon that rated words with high frequency in the domain corpus, identified based on expert opinion, with subsequent comparison of the results. The limitation of this approach lies in its dependence on human selection criteria which makes it vulnerable to subjectivity.

In terms of the Vietnamese language, Vu and Park [10] have contributed a solution that is based on a Vietnamese version of SentiWordNet by using the English SentiWordNet (ESWN) in conjunction with a Vietnamese dictionary (Vdict). This method requires translation of Vietnamese to English to calculate polarity scores based on the ESWN. The components of this solution include the ESWN, a Vdict and the Google translate Application Programming Interface (API). The biggest limitation of this work lies in the fact that Google functions erect a significant barrier as it is based on the structure of English, when there are different word meanings meanings possible in these two languages. In detail, one Vietnamese word may be translated into two or more English words. As a result, scores may be not exact.

B.Document classification

There have been several recent studies of document classification using lexicon-based [4, 7, 11, 12] and machine learning approaches [6,12]. Using the lexical methodology, a sentiment lexicon is first set up to store polarity-driven conceptual inferences for words. To determine the polarity of content, polarity scores for every expression present in the sentiment lexicon database are added to get an 'overall polarity score' [12]. For instance, if the vocabulary matches a word identified as positive in the lexicon, then the aggregate extremity score of the content is high. If the general extremity score of content is certain, then that content is designated as positive, else it is flagged as negative. However, the weakness of this approach, when compared with a machine learning approach, lies in its potential lack of accuracy as it relies on human judgement. For example, the task of identifying negativity or positivity is carried out by individuals whose bias may affect the accuracy of scores. Nevertheless, this method appears to be the best available at present and, variations of this lexical methodology have been found to have impressively high precision. In terms of machine learning approaches, the most popular technique for document classification seems to be Naïve Bayes [12, 13, 14, 15] (see figure 1).

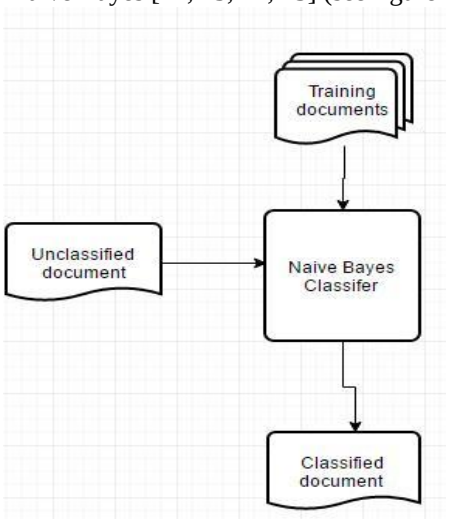

Figure 1. Model of document classification with the original Naïve Bayes classifier

After sentiment determination, the training data need to be linked to a system capable of machine learning. From the range of available methodologies, robotized content grouping has been found to reliable, offering options such as Support Vector Machines, Neural Networks, Naive Bayes Classifier, Decision Trees, Rocchio's Algorithms, and k-nearest neighbor [16].

C. Review and comparison of Vietnamese sentiment analysis

In terms of the English language, solutions to the problem of computerized content classification are available. However, for less popular languages, such as Vietnamese the process is far more complex and time-consuming.

Furthermore, research about Vietnamese sentiment analysis is exceedingly rare. Kieu and Pham [17], Nguyen, Van Le, Le and Pham [18] all focused on a lexicon-based method which is rule-based. For instance, Positive Negative > Negative, Negative; Negative >Positive This is relatively easy to implement, but accuracy is not high. This can be demonstrated by an example from the work of Nguyen, Van Le, Le and Pham [18] where "máy hơi nóng, pin nhanh hết (the phone is hot, spends battery very fast)" represents a negative opinion due to the word "hot" and "very fast", whereas the same words in another case, "Weather is hot, rice dries very fast", it may be positive.

In these cases, the lexicon method was used to analyse words and phrases which may lead to an incorrect result at

sentence or document level. Furthermore, to build the associated domain specific sentiment dictionary, English SentiWordnet 3.0 needs to be translated to Vietnamese. The difference between the two languages and the translation framework can lead to further inaccuracies. To increase accuracy, Duyen, Bach and Phuong [19] experimented with a machine learning approach for Vietnamese sentiment classification However, this produced limited results as they targeted only short structures at sentence level. To overcome such limitations, this research proposes a method at document level using a Sentiment dictionary as the main word vector for training to reduce time. A comparative analysis is shown in Table 1.

Table 1 Comparative Analysis

Using VSL	[17, 18]	[19]
-Machine learning method -Document level -Using Sentiment dictionary -Time required: short -Accuracy: high -Training pool needs to be increased	-Rule-based method -Sentence level -Using Sentiment dictionary -Time required: long -Accuracy: medium -Complicated to improve	-Machine learning method -Sentence level -A further step is needed to build the main vector -Time required: medium -Accuracy: Medium -Training pool needs to be increased

D. Naïve Bayes Classifier

According to Ahmed and Guan et al. [14], Naïve Bayes is one of the most straightforward probabilistic classifiers as long as it is Naïve Bayes with solid credulous freedom presumption. This presumption regards every single word as solitary, free and fundamentally unrelated. This methodology presupposes access to an accumulation of articles with pre-appointed supposition and certainty names at the archive level [12]. In this algorithm, C is the set of labels (Positive and negative) and ti is a new document that needs to be classified; the probability that ti belongs to class ci is

$$\Pr(c \mid t_i) = \frac{\Pr(c).\Pr(t_i \mid c)}{\Pr(t_i)} \quad (1)$$

With ci ∈ C;

Pr(c) can be calculated for the total document of class c divided by the total documents of all classes.

When estimating the highest probability of Pr(c|ti), it can skip calculating Pr(ti) due to the fact that it will not be necessary when comparing

Probability Pr(ti|c) is estimated by:

$$\Pr(t_i \mid c) = (\sum_n f_{ni})! \prod_n \frac{\Pr(w_n \mid c)^{f_{ni}}}{f_{ni}!} \quad (2)$$

Where fni is the frequency of word n in the test document ti and Pr(wn|c) is the probability of word n given class c

Because the term ∑nfni! And Πnfni! can be detected without any change in the final result due to independence from class c, the probability Pr(ti|c) is

$$\Pr(t_i|c) = \alpha \prod_n \Pr(w_n|c)^{f_{ni}} \quad (3)$$

α is the constant (∑nfni! And Πnfni!)

However, the advantage of this method simultaneously represents a disadvantage. Naive Bayes utilizes single words, without stem or stop word expulsion [12]. By skipping this process, identification will be faster and simpler. However, since the extremity of the content relies upon the score given to every vocabulary element, it involves an extensive volume of work, required to determine which lexical data is generally sufficient [12].

III. Proposed Method

The purpose of this research is to create a sentiment lexicon for Vietnamese and to use that lexicon as a basis for a machine learning approach to classifying documents. A sentiment lexicon will be generated by using a Term Frequency Times Inverse Document Frequency (TF-IDF) algorithm (Statistical). The characteristics of Naïve Bayes make it a suitable algorithm as it belongs to the group of probabilistic classifiers [6].

The new method consists of:

- Using VSL as Training Vector
- All documents will be present as Vectors
- Apply Naïve Bayer algorithms to classify positive/negative sentiments.

A Sentiment lexicon database is the input source for the machine learning method. This research also provides a technique that utlises sentiment lexicons for the analysis of information. This overcomes one of the weaknesses of Naïve Bayes which is that all training variables must be independent. There are, nevertheless, limitations as the VSL depends on its keywords which may reduce the accuracy of the classifiers. Furthermore, the lexicon still includes redundant words that were not removed by the algorithm. A possible solution is an expert review to perfect the lexicon database.

A. Approach to VSL

The publicly available existing sentiment lexicon supports only a small number of languages such as English and Arabic [1]. For other languages, such as Vietnamese, there is no sentiment lexicon available to be used as the dataset for document analysis. However, the inclusion of a sentiment lexicon is vital as text and spoken communication may include slang words. [20].

VSL is the Vietnamese sentiment lexicon that has been created by using a TF-IDF algorithm that calculates the weight of words based on a 'Bag-of-Words', an accumulation of unordered words, ignoring language structure, and which are the result of being tagged as negative or positive within a selected document. Each word or term is weighted for quality in terms of their significance or centrality for the arrangement within a procedure called "term-weighting". Term weights can be computed from a variety of points of view but constitute term weighting

methodologies in view of Term Frequency Times Inverse Document Frequency plans and other weighting plans [21].
Tf- term frequency: used to estimate the frequency of occurrences of a word in the text. However, each text is of different length so the number of occurrences of words may be higher if the text is longer. Thus, the number of occurrences of the word must be divided by the length of the text (the total number of words in that text) [21].
IDF- Inverse Document Frequency: estimates the importance of a word. When calculating tf, all words are considered equally important. However, some words are often used but not important for the main meaning of the paragraph.

Tokenization

Vietnamese is an alphabetic script which belongs to the group of Occidental languages. Alphabetic scripts generally isolate words with spaces and a tokenizer which essentially replaces spaces with word limitation [22]. However, not all dialects in Vietnamese use spaces to isolate words, nor are independent syllables always used in a similar fashion to make up words. Moreover, a number of Vietnamese syllables are words without any other input, yet can also be an element of multi-syllable words with syllables isolated by spaces between them. Furthermore, Vietnamese dialects create expressions of high complexity by joining syllables that more often than not also have significance when considered independently. This semantic pattern makes Vietnamese akin to that of syllabic scripts, such as Chinese [22]. That creates issues for all regular dialect handling tasks, dreating ambiguity in terms of what constitutes a word in context.

Vector creation

Vector creation is an important step common to all machine learning approaches [16]. Generally, each document is present as a numeric vector, signifying that each document Di will be present as Di=(di,i), where di is the vector of the document Di and di=(Pi1,Pi2,…,Pin). A dataset is imbalanced if the group classes are not roughly similarly represented. As this strategy delineates the execution issue of the entire framework, this is considered as the center part as it impacts the general operation [14]. Moreover, all phrase attributes in the vector are contained in the VSL. In this step, they will be divided into training vector and document vector.

(i) Training vector

The training vector is created from the VSL - element values are scores (see Table 2).

In the original Naïve Bayes classifier, the training table may contain several documents, which may make data training time consuming. The VSL has created two classes, and there are no requirements to collect excessive amounts of training documents, shortening the process of estimating the value of each element in the vector.

TABLE 2 TRAINING VECTOR USING VSL FOR NAÏVE BAYES CLASSIFIER

Phrase	Positive	Negative
Đảng	20.038	0
người	1.89	5.91
sáng_lập	0.13	0
trở_về	0	0.33
nguồn_cội	0	0.33
……..		

(ii) Document vector

The document Vector is the vector that was created from an unclassified document (see Table 3). The phrase attributes perform similarly to the training vector, with the value of each element of the vector being equal to the count of that element in the document.

Table 3. Document in the Vector format

Doc	Phrase					
	Đảng	người	sáng_lập	….	trở_về	nguồn_cội
1	2	1	0	…..	6	3
2	1	1	7	…	2	0
3	16	2	1	….	0	0
4	0	0	0	0	0	0

B.Naïve Bayes classifier

Following the document preparation stages, the quantity of elements that need to be considered will have been reduced considerably and are more exact for utilization in building the grouping model. For the classification stage, Naïve Bayes is utilized as the classifier due to its simplicity and its proven high track record in recording and ordering content, making it the least complicated example of a probabilistic classifier. The yield Pr(C|d) of a probabilistic classifier is the likelihood that an archive d has a place within a class c. Every record contains terms which are given probabilities taking into account the number of an event inside specifically designated archives.

For example, using training data from a phrase vector p={phrase1, phrase2, phrase3, phrase4, phrase5}, positive training vector Pos={3,2,1,0,1}, negative training vector Neg={0,2,0,9,4} and the needed classification vector D={1,2,1,0,1}. As an example, this produces the following data:

- Prior probability of Positive P(Positive)=1/2
- Prior probability of Negative P(Negative)=1/2
- Total number of phrases =5
- Total value of phrase that in positive vector =7
- Total value of phrase that in negative vector =15

Therefore, by applying Naïve Bayes classifier,

- The prior probability that D belong to Positive group is estimated by :

P(D|Pos)=P(Positive)*P(phrase1|Pos)1*P(phrase2|Pos)2*P(phrase3|Pos)1*P(phrase4|Pos)0 *P(phrase5|Pos)1

- The prior probability that D belong to Negative group is estimated by :

P(D|Neg)=P(Negative)*P(phrase1|Neg)1*P(phrase2|Neg)2*P(phrase3|Neg)1*P(phrase4|Neg)0

*P(phrase5|Neg)1

Applying Naïve Bayes formula, it generate

- P(phrase1|Pos)=(3+1)/(5+7)=0.3333
- P(phrase2|Pos)= (2+1)/(5+7)=0.25
- P(phrase3|Pos)= (1+1)/(5+7)=0.1667
- P(phrase4|Pos)= (0+1)/(5+7)=0.0833
- P(phrase5|Pos)= (1+1)/(5+7)=0.1667
- P(phrase1|Neg)=(0+1)/(5+15)=0.05
- P(phrase2|Neg)= (2+1)/(5+15)=0.15
- P(phrase3|Neg)= (0+1)/(5+15)=0.05
- P(phrase4|Neg)= (9+1)/(5+15)=0.5
- P(phrase5|Neg)= (4+1)/(5+15)=0.25

Finally, from the above equations, P(D|Pos)=0.000289 and P(D|Neg)=0.000007. Because P(D|Pos) > P(D|Neg), document D will be tagged by appropriate positive Naïve Bayes rules. In addition, in the case of that probability of positive being equal with the probability of being negative, that document is marked as neutral.

IV. Evaluation

Classification methods based on probability theory determine the probability of an event taking place. The higher the probability of an outcome, the more likely it is that it will happen. This is particularly significant for predictive and categorical problems for Machine Learning. In statistical terms, each determination based on probability is generally accompanied by a probability distribution that is consistent with the problem (Vapnik et al., in Bengio et al. [31].

The purpose of this research is to classify documents from the Internet that contain information within news, comments, or blogs. To test the proposed method of using Naïve Bayes with document preparation for Vietnamese document classification, a total of 1000 documents containing textual data from 5 Vietnamese websites were utilised. These websites were grouped in terms of potential positive and negative content:

- Positive: nguyenphutrong.org, dancongsan.vn, vnexpress.net, thanhnien.vn
- Negative: viettan.org

All documents underwent a document preparation phase to reduce the number of phrases. This process proved to be imperfect, however, as the resulting VSL still contains about 2166 phrases. The figure shows the word reduction that took place during document preparation for use in the VSL.

The experiment consists of two parts, creating first a VSL based on data from dangcongsan.vn as the positive training document and viettan.org as the negative training document. The second element is the application of the proposed model so that classification can take place.

A.Building a Vietnamese sentiment lexicon (VSL)

The VSL is generated by calculating the weight of every phrase in the training document after the process of document preparation in order to reduce the length (remove any unnecessary words that may create noise and so interfere with classification). The format of the VSL is similar to that of SentiWordNet 3.0. Each phrase is assigned a phrase number followed by a positive or negative score.

In the VSL, each element, therefore, has a positive or negative score; if the positive score is higher than the negative score, that element is presented as positive. If positive scores equaled negative scores, the word or phrase was identified as neutral. The key to the VSL is the available number of training documents, which were just two instead of a large document corpus. The challenge when building the VSL is collecting training documents. All elements in the VSL come from training documents so the effectiveness of the VSL depends on how many documents are collected and the value of those documents.

B.Naïve Bayes classifier

For practical applications, the text has more than 10 words, but can be up to millions, resulting in a long vector. A text with only one sentence, and one of a thousand pages is represented by vectors with dimensions of 100 thousand or 1 million.

There are many words in the dictionary not appearing in a text. Thus, the derived vectors usually have a significant number of zero elements. Vectors with many zero elements are called sparse. For more efficient storage, we do not store the vector but only the position of the non-zero elements and their corresponding value. In addition, sometimes there are rare words not in the dictionary. Often carrying the most important information. This is a downside of bag-of-words. An alternative method to overcome this disadvantage is using VSL which is based on Term Frequency-Inverse Document Frequency (TF-IDF) to determine the importance of a word in a text based in the entire text in a corpus

The input of Naïve Bayes classifiers is the VSL and documents that need to be classified, and these data is transformed to the type of Vector. The testing document is divided into 2 groups: labelled (negative and positive website) and unlabeled (BBC, RFA, VOA, x-café.net). Three methods (the proposed method, the Naïve Bayes and the lexical method) were applied to the labelled group to calculate the accuracy, recall, precision and F-score (See figure 2).

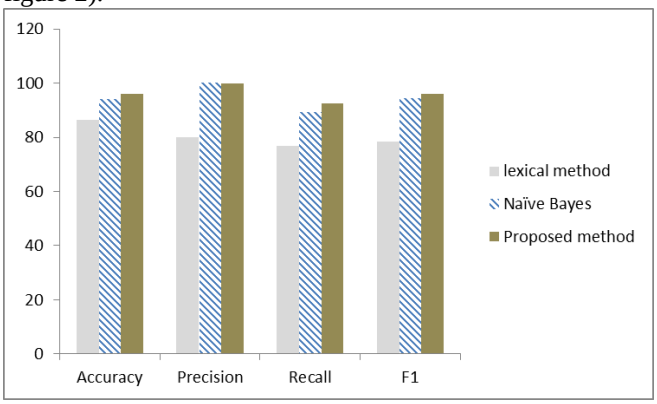

Figure 2 Evaluation

The most common ways of evaluating the result of the machine learning experiment are accuracy, recall, precision and F-measure [23]. In detail, recall is the extent to which Real Positive cases are accurately predicted and positive while precision means the extent to which the predicted positive cases are in effect 'Real Positives'.

Table 4 The accuracy of three methods that applying to same text bank.

Sources	Proposed method	Naïve Bayes method	Lexical method
Nguyenphutrong.org	98.5 %	96%	88%
dangcongsan.vn	99%	96.5%	82%
vnexpress.vn	97.5%	95.5 %	90.5%
thanhnien.vn	96%	93.5%	85%
viettan.org	100%	99%	91%
Average	98.2 %	96.1 %	87.3 %

Table 4 illustrates a score that has been tested through three methods. The highest score belongs to the proposed method with the use of Naïve Bayes and a document preparation process. When applying Naïve Bayes with VSL, there are 2166 elements in total, which need to be calculated and multiplied 2100 times. The output result is extremely small and cannot be achieved manually. To handle these issues, Matlab is considered appropriate software that supports the Naïve Bayes classifier.

All processes are carried out by using the command statements, making it easier and faster. A further issue is the similarity of the scores. For example, if document d has a positive score of 50.1 % and a negative score is 49.9 %, then according to the rule of Naïve Bayes, d is a positive document. However, these probability scores are too close to be meaningful, although there were other elements where the proposed method was able to produce more precise outcomes (see Figure 3)

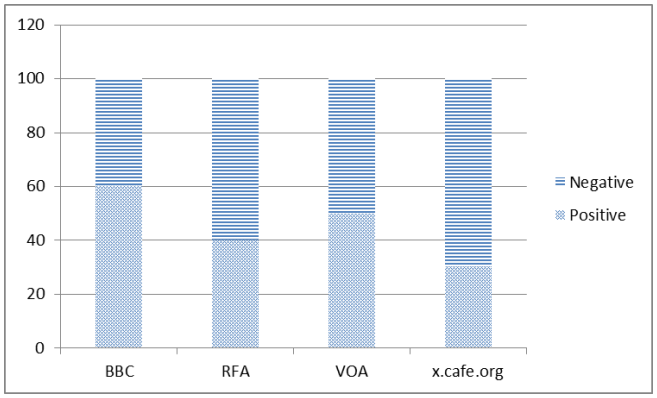

Figure 3. Evaluation of experiments

On the other hand, in the case of a group of unlabelled websites (BBC Vietnamese, RFA Vietnamese, VOA Vietnamese, x-cafe.org), those contained both positive and negative elements. The expected outcome when applying the proposed method is that they have both positive and negative documents. Figure 7 shows the results of using the Naïve Bayes classifier in conjunction with the document preparation process.

As expected, these websites contain both positive and negative content. While only for BBC Vietnamese the amount of positive information was greater than that being negative, other websites had smaller numbers of positive documents.

In addition, this paper made a experiment with English data.The method is same as proposed method with Vietnamsese. The training for Naïve Bayes classification is English SentiWordNet 3.0. SentiWordNet 3.0 is an enhanced rendition of SentiWordNet 1.0, a freely accessible research dictionary, as of now authorized to in excess of 300 research gatherings and utilized as a part of an assortment of research ventures. Around the globe. Both SentiWordNet 1.0 and 3.0 are the aftereffect of naturally clarifying all words adjusts as per their positive, negative and unbiased levels. SentiWordNet 1.0 and 3.0 vary in the renditions of words they comment on [28]. 3000 classified sentences were collected and were used to testing [29]. Figure 4 show the result of the experiment when apply 1500 positive sentences and 1500 negative sentences. By using SentiWordNet 3.0, 84.99 % sentences in positive group is true , while it is 92.96 % in negative group.

Both Precision and Recall are less than or equal one (See figure 5). High precision means that the accuracy of the find is high. High Recall means high True Positive Rate, ie the percentage of missing positive real points is low.

When Precision = 1, all points found are really positive, it means no negative points are added to the result. However, Precision = 1 does not guarantee the model is good, because the question is whether the model has found all the positive points or not. If a model only finds the right positive point, it is most likely not to be called a good model.

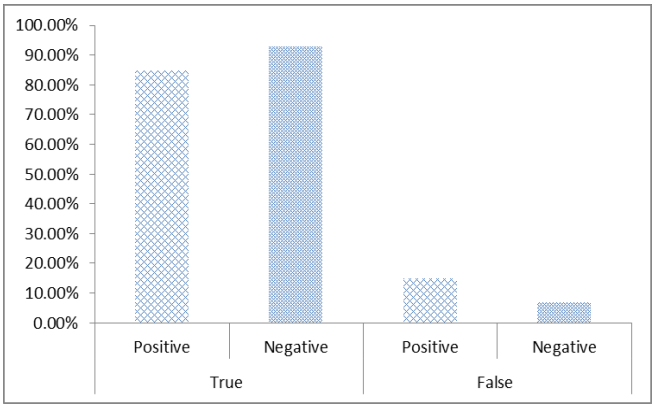

Figure 4 Accuracy of two labelled groups

When Recall = 1, all positive points are found. However, this quantity does not measure how many negative points get mixed up. If the model categorizes every point as positive then surely Recall = 1. A good layering model is the model with both Precision and Recall are as high, as close as possible.

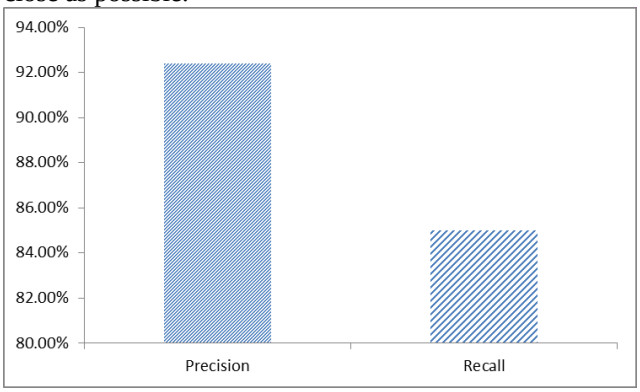

Figure 5 Recall and precision

The experience's result in table 5 clearly indicated that the proposed, with only two training class, is got a high accuracy as using 100 training class in normal Naïve Bayes classification. Normal Naïve Bayes classification uses classified document or sentences as training class

The table 5 is a result of using separate positive dataset and negative dataset. Each dataset contains 1500 sentences.

In comparison with other author (table 6), the proposed method which using SentiWordNet, gets a good result in term of precision, recall and F-score. The results of comparison shows that the proposed method yields less results than the Ting et al (2011) method, but the difference is negligible (+/- 3%), the training dataset when classified by proposed method is 2 class , while the other used high number of training to get high accuracy.

Table 5 Comparison of classification

Prediction	Class	Proposed method	Naïve Bayes		
			2 training	50 training	100 training
True	Positive	84.99%	95.43%	45.25%	85.34%
	Negative	92.96%	16.40%	64.35%	90.15%
False	Positive	15.01%	4.57%	54.75%	14.66%
	Negative	7.04%	83.60%	35.65%	9.85%

In addition, the Bayesian method has a distinct advantage in classification rates with Gitari et al (2015) method. They used dictionary and lexical based method. The method get advantage of training data , however the accuracy is less than 20% with proposed method.

Table6 Comparison

Author	Precision	Recall	F-measure	Training data
Proposed	92.39%	85.00%	88.54%	2
Mulajati et al, 2017	87.38%	96.26%	91.60%	915
Ting et al, 2011	95.60%	95.50%	95.50%	1200
Gitari et al, 2015	71.55%	68.24%	69.85%	Dictionary
Geng et al, 2015	16.10%	48.34%	24.02%	72
Xu, 2018	77.08%	72.57%	72.28%	N/A

V. Application

Real time Prediction

Naïve Bayes classifier runs relatively fast which makes it suitable for real-time applications, such as warning and trading systems [26].

Multi class Prediction

Thanks to Bayes' extended theorem, we can apply it to any kinds of predictive applications, such as predicting target hypotheses [24].

Text classification / Spam Filtering / Sentiment Analysis

The Naïve Bayes classifier is also well suited for text or natural language classification systems because its accuracy is greater than that of other algorithms. In addition, anti-spam systems also favor this algorithm. And psychoanalytic systems also apply the Naïve Bayes to conduct psychological analysis of preferred and unpopular products from the analysis of customer behavior and habits [25].

Recommendation System

Naive Bayes Classifier and Collaborative Filtering are often used in combination to build a system of suggestions, such as appearing ads that users are most interested in learning the habit of using the internet of the user [27], or as an example of the beginning of the article which gives hints for the next song that the user would like in a music application.

VI. Conclusion

This research has concentrated on communication and has set out to solve the question "how to build a sentiment lexicon that is related to specific domains". It involves processes of data collection and algorithms to calculate their polarity scores. This output is essential to create a sentiment lexicon which is the key element for the classification of document data. Moreover, the research presents a model/process and algorithms about "How to apply sentiment lexicons" as well as creating an experiment to test the accuracy of this method. This method uses the Naïve Bayesian method, where the object is made based on social media textual data, provided by statistics about the classification of information. It also allows the use of training topics at the discretion of the user when having a standard data set.

The major contribution of the proposed method of using a sentiment lexicon as a basis for Naïve Bayes classifier is that it is providing existing training data for the learning machine method instead of forcing users to collect a massive training corpus and it can be applied other languages rather than Vietnamese .

From the results of the experiment, it is evident that the proposed method gains the highest score. However, most of the results are either positive or negative, whereas there should also be neutral documents. Furthermore, the VSL is too time-consuming as it still includes a number of stop words. In future work, the VSL should be reviewed by experts in the Vietnamese language. Moreover, it should include the capacity to identify neutral documents.

Acknowledgement

We are grateful to Mrs. Angelika Maag for proof reading and making corrections to this article. Without her support, it would have not been possible to submit this in the current form.

References

[1] K. Al-Rowaily, A. Muhammad, A.H. Nur, and A. R. Majed, "BiSAL – A bilingual sentiment analysis lexicon to analyze Dark Web forums for cyber security," *Digital Investigation*, vol. 14,(2015), pp. 53-62.

[2] S. Park, W. Lee, and L.C. Moon, "Efficient extraction of domain specific sentiment lexicon with active learning," *Pattern Recognition Letters*, vol. 56, (2015), pp. 38-44.

[3] H. H. Lek, and D. C. C. Poo, "Automatic Generation of an Aspect and Domain Sensitive Sentiment Lexicon," *International Journal on Artificial Intelligence Tools*, vol. 23,(2014), pp. 1-21.

[4] R. Feldman, "Techniques and Applications for Sentiment Analysis," *Communications of the ACM*, vol. 56(4), (2013), pp. 82-89.

[5] N. D. Gitari, Z. Zuping, H. Damien, and J. Long, "A Lexicon-based Approach for Hate Speech Detection," *International Journal of Multimedia and Ubiquitous Engineering*, vol.10(4), (2015), pp. 215-230.

[6] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Engineering Journal*, vol. 5(4), (2014), pp. 1093-1113.

[7] G. Katz, N., Ofek, and B. Shapira, "ConSent: Context-based sentiment analysis," *Knowledge-Based Systems*, vol. 84, (2015), pp. 162-178.

[8] G. Paltoglou, and M. Thelwall, "A study of information retrieval weighting schemes for sentiment analysis," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, (2012).

[9] A.M. Yang, J. H. Lin, Y. M. Zhou, and J. Chen, "Research on building a Chinese sentiment lexicon based on SO-PMI," in *Applied Mechanics and Materials*, (2012), pp. 1688-1693.

[10] X. S. Vu, and S. B. Park, "Construction of vietnamese SentiWordNet by using vietnamese dictionary," arXiv preprint arXiv:1412.8010, (2014).

[11] M. Thelwall, and K. Buckley, "Topic-based sentiment analysis for the social web: The role of mood and issue-related words," *Journal of the American Society for Information Science and Technology*, vol. 64(8), (2013), pp. 1608-1617.

[12] C. H. Bhadane, H. Dalal, and H. Doshi, "Sentiment Analysis: Measuring Opinions," *Procedia Computer Science*, vol. 45, (2015), pp. 808-814.

[13] N. Korada, N. Kumar, and Y. Deekshitulu, "Implementation of Naive Bayesian Classifier and Ada-Boost Algorithm Using Maize Expert System," International *Journal of Information Sciences and Techniques (IJIST)*, vol. 2(3), (2012), pp. 63-75.

[14] I. Ahmed, D. Guan, and T. Chung, "SMS classification based on naive bayes classifier and apriori algorithm frequent itemset," *International Journal of machine Learning and computing*, vol. 4(2), (2014), pp. 183-187.

[15] M. F. Kabir, "Enhanced classification accuracy on naive bayes data mining models," *International Journal of Computer Applications*, vol. 28(3), (2011), pp. 9-16.

[16] K. Thaoroijam, "A Study on Document Classification using Machine Learning Techniques," *International Journal of Computer Science Issues*, vol. 11(2), (2014), pp. 1694-0784.

[17] B.T. Kieu, and S. B. Pham, "Sentiment analysis for Vietnamese. in Editor (Ed.)^(Eds.):", 'Book *Sentiment analysis for vietnamese*' (IEEE, 2010, edn.), (2015), pp. 152-157.

[18] H. N. Nguyen, T. Le, H.S. Le, and T. V. Pham, "Domain specific sentiment dictionary for opinion mining of vietnamese text," in Editor (Ed.)^(Eds.): (2014), pp. 136-148.

[19] N. T. Duyen, N. X. Bach, and T. M. Phuong, "An empirical study on sentiment analysis for Vietnamese," in Editor (Ed.)^(Eds.): Book An empirical study on sentiment analysis for Vietnamese, (2014), pp. 309-314.

[20] S. Huang, Z. Niu, and C. Shi, "Automatic construction of domain-specific sentiment lexicon based on constrained label propagation," *Knowledge-Based Systems*, vol. 56, (2014), pp. 191-200.

[21] T. C. Ying, S. Doraisamy, and L. N. Abdullah, "Lyrics-Based Genre Classification Using Variant tf-idf Weighting Schemes," *Journal of Applied Sciences*, vol. 15(2), (2015), pp. 289- 294.

[22] N. T. M. Huyên, A. Roussanaly, and H. T. Vinh, "A hybrid approach to word segmentation of Vietnamese texts," in *Language and Automata Theory and Applications*, (2012), pp. 240-249.

[23] S. Ting, W. Ip, and A. H. Tsang, "Is Naive Bayes a good classifier for document classification?," *International Journal of Software Engineering and Its Applications*, vol. 5(3), (2011), pp. 37-46.

[24] Vural, M.S. and M. Gök, Criminal prediction using Naive Bayes theory. *Neural Computing and Applications*, (2017), pp. 2581-2592.

[25] Xu, S., Bayesian Naïve Bayes classifiers to text classification. *Journal of Information Science*, (2018), pp. 48-59.

[26] Gujar, S. and B. Patil, Intrusion Detection using Naïve Bayes for real time data. *International Journal of Advances in Engineering & Technology*, (2014), pp. 568-568.

[27] Mulajati, M. and R.B.F. Hakim, Sentiment Analysis on online reviews using Naïve Bayes Classifier Method and Text Association (Case Study: Garuda Indonesia Airlines Passengers Reviews on TripAdvisor Site). *Indian Journal of Scientific Research*, (2017), pp. 274.

[28] Baccianella, S., A. Esuli, and F. Sebastiani, *SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining*. Vol. 10,(2010).

[29] Dimitrios Kotzias, et al., *From Group to Individual Labels using Deep Features*, (2015).

[30] Geng, H., et al., Prediction of protein-protein interaction sites based on naïve bayes classifer. *Biochemistry research international*.(2015)

[31] Y. Bengio, I. J. Goodfellow, & A. Courville, "Deep learning". *Nature*, *521*(7553), pp. 436-444.