A Multi-layer Neural Network-based System for Vietnamese Sentiment Analysis at the VLSP 2016 Evaluation Campaign

Thy Thy Tran*, Xanh Ho† and Nhung T.H. Nguyen‡
Faculty of Information and Technology
VNUHCM University of Science
*Email: *thy2512@gmail.com, †xanhhocntt@gmail.com, ‡nthnhung@fit.hcmus.edu.vn*

***Abstract*—We present a description of our system submitted to the VLSP 2016 Evaluation Campaign of Sentiment Analysis for the Vietnamese Language. This year the campaign focussed on polarity classification, i.e., to classify Vietnamese reviews or documents into positive, negative or neutral. In order to address the task, we implemented a multi-layer neural network-based method that uses three types of features as input. Our internal evaluations indicate that by using TF-IDF feature to represent sentences, we can obtain the best performance with 66% of precision and 65% of recall. The official accuracy of the proposed method on the testing set (evaluated by the organiser) is 65.9%.**

1. Introduction

Recently, with the explosive of social media, there is a high demand from brands and industries to automatically analyse the customers' comments on their products so that they can know how consumers perceive their products as well as those of their competitors. This sentiment information is not only useful for marketing and product benchmarking but also useful for product design and product development [1]. Extracting opinion or sentiment from text can be defined as sentiment analysis or opinion mining. The task receives raw texts talking about brands or products as input, and outputs sentiment information, which indicates the author's opinions about a specific brand, product, or about products' features. In general, the task of sentiment analysis can be divided into three levels: document-based, sentence-based, and aspect-based. The VLSP 2016 evaluation campaign of sentiment analysis for the Vietnamese language focussed on the first level[1], i.e., classify input documents or reviews into one over three sentiment classes: positive, negative and neutral.

Most approaches for sentence-based sentiment analysis are supervised learning-based methods such as Naive Bayes, support vector machine (SVM) and neural networks (NN). In this paper, we have implemented multilayer NN (MLNN) for Vietnamese sentiment analysis. Specifically, our model is a fully connected neural network with only one hidden layer and gets feature vector as input. We have conducted experiments to compare the proposed method with SVM implementation from scikit-learn [2] and fastText from facebook [3]. For SVM, we also extract the same features that we use for MLNN. Meanwhile, regarding fastText, the model first randomizes a weight matrix that is a lookup table over words, then the word representations are averaged into a text representation and experience classification on it. Experimental results show that MLNN with TF-IDF feature as input produced the best scores among the models. We have applied the best performing method to the testing set and obtained an official accuracy of 65.9%.

2. Related Work

While research on sentiment analysis for English has been being grown and obtained the state of the art [4], [5], [6], [7], [8], there is a few work for Vietnamese. Kieu and Pham [9] introduced a rule-based system using the GATE framework for sentence-level sentiment analysis. They conducted their experiments on a corpus of computer product reviews and obtained 61.16% of precision, 64.62% of recall. Ha et al. [10] described an extension of a feature-based opinion mining and summarizing model to extract sentiments from reviews on mobile phone products. Feature words and opinion words were extracted based on some Vietnamese syntactic rules. Opinion orientation and summarization on features were determined by using VietSenti WordNet. Their model produced 69.16% of precision and 68.86% of recall. Nguyen et al. [11] proposed an approach extract opinions from Vietnamese documents using a domain specific sentiment dictionary. The sentiment dictionary is built incrementally by applying statistical methods to 20,083 comments about mobile products crawled from the Internet. Nguyen et al. [12] introduced an annotated corpus for document-level sentiment analysis that consists of hotel reviews. They implemented several machine learning-based methods that use different types of features. Their experimental results indicated that using word-based features produced better performance than using syllable-based ones. Among different types of n-grams, only unigrams were effective for the task.

[1] http://vlsp.org.vn/evaluation_campaign_OM

3. System Overview

For each input document, we extract a feature vector that can represent as exactly as possible the characteristic of the input. Specifically, we extract three types of features: Bag-Of-Word (BOW), TF-IDF and SentiWordNet-based features. Then, each document vector based on the above features along with its label will be fed to a classifier to learn a model that can determine the sentiment class.

3.1. Feature Extraction

BOW and TF-IDF are common features that have been used in text mining as well as other NLP problems. While SentiWordNet-based features are extracted based on Viet-Senti Wordnet [13]–a lexicon resource that contains sentiment expressions and its three types of scores. Details about these features will be presented below.

3.1.1. BOW. Each review is represented by a sparse vector as the bag of its words over a fixed vocabulary, ignoring grammar and word order.

3.1.2. TF-IDF. TF-IDF reflects how importance is a word to a document in the corpus. It is composed by two terms: (1)Term Frequency (TF) computes the number of times that word occurs in the current document, and (2) Inverse Document Frequency (IDF) shows how much information a term provides. In this work, we use TF-IDF scores as review representation vectors.

3.1.3. SentiWordnet-based. VietSentiWordNet [13] is a synset of words that express sentiment. However, it contains not only sentiment words but also phrases with some included sentiment shifter such as *not*. Therefore, we have to extract n-gram, range from unigram to pentagram, to map vocabulary of VietSentiWordnet. In details, we use the lexicon in three ways:

- **BOW-senti** We extract the BOW feature based on a fixed vocabulary extracted from VietSenti Wordnet. The whole collocation contains 1198 terms with their corresponding frequency of n-grams showed in Table 1.
- **TF-IDF-senti** We also used the VietSentiWordnet to compute TF-IDF scores.
- **Objectivity-score** The Wordnet also provides information of the sentiment scores involving positive, negative, and objectivity scores extracted from positive and negative as

  $$ObjScore = 1 - (PosScore + NegScore)$$

  Based on the objectivity score, we build a feature vector represents VietSenti Wordnet, for each sentiment word/phrase appears in a sentence, we add its corresponding objectivity score.

TABLE 1. VIETSENTIWORDNET N-GRAM OCCURENCE

N-gram	Count	Example
1-gram	885	’khủng_khiếp’
2-gram	236	’tốt nhất’
3-gram	61	’giá quá cao’
4-gram	13	’không có ý định’
5-gram	3	’không tạo được cảm hứng’

TABLE 2. CHARACTERISTICS OF THE PROVIDED DATA

	POS	NEG	NEU	Total
TRAIN	1700	1700	1700	5100
TEST	350	350	350	1050

3.2. Classification

In this work, we use three different algorithms to classify reviews' polarity. The first one is a linear **support vector machine** (**SVM**) classifier provided by the scikit-learn toolkit [2]. The second is an implementation of **multilayer neural network** (**MLNN**) using NumPy which provides multidimensional arrays and functions [14]. The last is an extra experiment using a recent released library named **fastText** [3].

Regarding the SVM classifier, we use the linear kernel which handles multiclass classification by a one-vs-rest scheme. In details, the strategy involves training a single classifier per class in order to produce a confidence score for its decision, then the label that has the highest confidence would be the predicted class.

For the MLNN model, we conducted experiments with several architectures and hyperparameters include learning rate, and $l2$ regularization scale. Specifically, we use Stochastic Gradient Descent (SGD) to optimize the objective function and use a flag for early stopping if the validation accuracy is not increased after 20 epochs. The extracted features will be used as input to the network.

For comparison, we also train a supervised classifier using fastText, which is a library for learning word representations and sentence classification introduced by Facebook.

We evaluate all the above-mentioned models using cross-validation to assess how the models generalize for an independent data.

4. Experiments

4.1. Dataset

The provided data this year (2016) is in the domain of technical devices, of which the training data consists of 5100 reviews with an equal proportion for each sentiment class. The data distribution is presented in Table 2. A noticeable point is that a neutral label will be assigned to a review if the review contains both positive and negative opinions. Table 3 illustrates this phenomenon, which makes the classification task more difficult.

TABLE 3. A REVIEW THAT IS LABELED AS NEUTRAL SINCE IT CONTAINS BOTH POSITIVE AND NEGATIVE SENTIMENT

Rất tốt nha bạn , hơi cấn ngón_cái chút vì có hai nút bấm gồ lên , xài lâu cũng quen , hơi nặng chút , nếu bạn xài Mac nên đầu_tư một con mighty để bàn , con này đem đi_lại , về kết_nối bluetooth thì cảm_giác lag hơn Magic_mouse , còn nếu dùng usb receiver thì rất ngon , battery thì mình dùng một tháng nay chưa hết , không_bao_giờ Off nguồn .

TABLE 4. PERFORMANCES OF SVM WITH DIFFERENT TYPES OF FEATURES

SVM	Precision	Recall	F1-score
BOW	0.61	0.62	0.61
TF-IDF	**0.65**	**0.65**	**0.65**
BOW-senti	0.44	0.38	0.34
TF-IDF-senti	0.44	0.39	0.34
Objectivity-score	0.49	0.48	0.47

4.2. Evaluation Metrics

Evaluation metrics in this work include accuracy, precision, recall, and F1 score. However, we only compute the last three ones for each model while running cross-validation, including the average scores for three categories (positive, negative, and neutral) and the total scores overall. The purpose is to select the best performing model that will be applied to the official testing set of the sentiment analysis campaign.

4.3. Experimental Results

Table 4 presents the performances of SVM when we use five separate features as input through cross-validation. Overall, TF-IDF obtains the highest scores for all precision, recall, and F1 with 0.65, 0.65 and 0.65, respectively. Meanwhile, the performance when using features extracted from the VietSenti Wordnet, is even worse than that of BOW.

Regarding MLNN, we designed its architecture as follows. The input layer has a dimension that depends on the dimension of the feature vectors. The hidden layer has 100 computational units. The last layer with 3 neurons stands for the number of labels which is positive, negative, and neutral. The neural structure was chosen as the model that produces the best results after several running and evaluating using cross-validation Table 5 shows the performance of MLNN using the same features with SVM. As can be seen, using TF-IDF as input for MLNN also performs the best among our features, similarly to SVM. Specifically, the F1-score over the three sentiment classes is also 0.65, which is equal to the best score produced by SVM. However, MLNN produces a little higher precision score (0.66 vs. 0.65 by SVM).

Training a classifier using fastText is a bit different with that for SVM and MLNN. We do not need to feed into the input layer the extracted features, the model already contains a random initialization weight matrix as a word representation look up table. The matrix would be adjusted through the training step. Then, the text representation is composed by averaging its word representations before transform to the classification layer. By using fastText, we obtained the same scores of precision and recall with 0.498.

TABLE 5. PERFORMANCES OF MLNN WITH DIFFERENT TYPES OF FEATURES

MLNN	Precision	Recall	F1-score
BOW	0.64	0.63	0.63
TF-IDF	**0.66**	**0.65**	**0.65**
BOW-senti	0.43	0.37	0.29
TF-IDF-senti	0.39	0.37	0.30
Objectivity-score	0.47	0.45	0.43

Among the three classifiers, when using SVM and MLNN with TF-IDF features, we obtained the best score of 0.65 F1-score. However, precision produced by MLNN is a little bit higher than that by SVM. Therefore, we select MLNN with TF-IDF features to apply to the official testing set of the campaign. The results on the testing set are presented in Table 6.

TABLE 6. OFFICIAL TESTING RESULTS ON INDIVIDUAL CLASS. THE OVERAL ACCURACY IS 65.9%

Class	Precision	Recall	F1-score
POS	69.06	71.43	70.23
NEG	65.67	62.86	64.23

4.4. Discussion

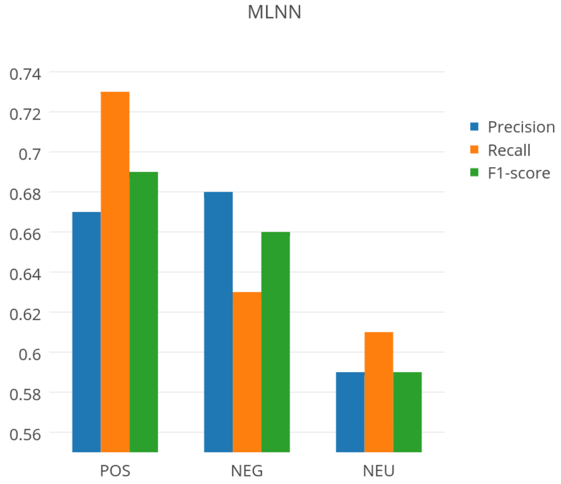
The above-mentioned experimental results indicate that with this specific data of Vietnamese sentiment analysis, SVM and MLNN produce comparable scores for all metrics. This situation is a bit conflict with the trend in English natural processing, in which neural networks beat state-of-the-art in many areas such as speech recognition [15], language modeling [16], and especially sentiment classification [8]. However, such situation can be explained by the fact that the provided training data is smaller than that for the English. Our dataset involves only 146,440 tokens and the vocabulary size is only 2,248 words. Meanwhile, to obtain the state-of-the-art, the English sentiment tree bank consists of 184,837 tokens with informative treebanks. This means that in order to effectively train a neural network-based model, we need a bigger training data with more linguistic information.

Another interesting phenomenon is that, although the training data has a balanced ratio of sentences in each class, the average length among them is quite significant different as showed in Table 7. The average length of a neutral review is nearly 6 to 7 words longer than a positive or negative review. Likewise, the number of tokens in neutral reviews is also close to twice times of that in positive or negative ones. These points make the problem hard to solve the imbalance word occurrence over three classes. Even in our final model, the average performance through cross-validation also bias to learn well on positive class than the other two, and often classify the neutral label incorrectly (see Figure 1).

TABLE 7. AVERAGE LENGTH (IN WORDS) OF DOCUMENTS IN THE TRAINING DATA

	POS	**NEG**	**NEU**	**Total**
Avg. length	11.67	12.60	**18.79**	14.36
Num of tokens	39689	42845	**63906**	146440

Figure 1. Precision, Recall and F1-score of MLNN using TF-IDF over three categories: positive (POS), negative (NEG) and neutral (NEU)



5. Conclusion

This paper has described our system submitted to the sentiment analysis evaluation campaign of the VLSP 2016. To conclude, for small data, the traditional feature is still better than neural networks with word representation. Due to the limited collocation of the SentiWordnet, it is not a good choice to use separately as the only feature. Moreover, our experiments only use distinct features as input without combining them, it should be considered as an avenue for the further work.

References

[1] B. Liu, "Sentiment analysis and opinion mining," *Synthesis Lectures on Human Language Technologies*, vol. 5, no. 1, pp. 1–167, May 2012. [Online]. Available: http://dx.doi.org/10.2200/S00416ED1V01Y201204HLT016

[2] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[3] A. Joulin, E. Grave, P. Bojanowski, and T. Mikolov, "Bag of tricks for efficient text classification," *arXiv preprint arXiv:1607.01759*, 2016.

[4] R. Tong, "An operational system for detecting and tracking opinions in on-line discussions," in *Working Notes of the SIGIR Workshop on Operational Text Classification*, New Orleans, Louisianna, 2001, pp. 1–6.

[5] B. Pang, L. Lee, and S. Vaithyanathan, "Thumbs up? sentiment classification using machine learning techniques," in *IN PROCEEDINGS OF EMNLP*, 2002, pp. 79–86.

[6] S. Kiritchenko, X. Zhu, and S. M. Mohammad, "Sentiment analysis of short informal texts," *Volume 50*, pp. 723–762, 2014.

[7] G. Ganu, N. Elhadad, and A. Marian, "Beyond the stars: Improving rating predictions using review text content," 2009.

[8] R. Socher, J. Pennington, E. H. Huang, A. Y. Ng, and C. D. Manning, "Semi-Supervised Recursive Autoencoders for Predicting Sentiment Distributions," in *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2011.

[9] B. T. Kieu and S. B. Pham, "Sentiment Analysis for Vietnamese," IEEE CS. IEEE CS, 2010.

[10] Q. Ha, T. Vu, H. Pham, and C. Luu, "An Upgrading Feature-Based Opinion Mining Model on Vietnamese Product Reviews," in *Active Media Technology - 7th International Conference, AMT 2011, Lanzhou, China, September 7-9, 2011. Proceedings*, 2011, pp. 173–185. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-23620-4$_2$1

[11] H. N. Nguyen, T. V. Le, H. S. Le, and T. V. Pham, "Domain Specific Sentiment Dictionary for Opinion Mining of Vietnamese Text," in *Multi-disciplinary Trends in Artificial Intelligence - 8th International Workshop, MIWAI 2014, Bangalore, India, December 8-10, 2014. Proceedings*, 2014, pp. 136–148. [Online]. Available: http://dx.doi.org/10.1007/978-3-319-13365-2$_1$3

[12] N. T. Duyen, N. X. Bach, and T. M. Phuong, "An empirical study on sentiment analysis for Vietnamese," in *2014 International Conference on Advanced Technologies for Communications (ATC 2014)*, Oct 2014, pp. 309–314.

[13] X.-S. Vu and S.-B. Park, "Construction of vietnamese sentiwordnet by using vietnamese dictionary," *The 40th Conference of the Korea Information Processing Society*, vol. 21, pp. 745–748, 2014.

[14] S. v. d. Walt, S. C. Colbert, and G. Varoquaux, "The numpy array: A structure for efficient numerical computation," *Computing in Science and Engg.*, vol. 13, no. 2, pp. 22–30, Mar. 2011. [Online]. Available: http://dx.doi.org/10.1109/MCSE.2011.37

[15] G. E. Dahl, M. Ranzato, A. Mohamed, and G. E. Hinton, "Phone recognition with the mean-covariance restricted Boltzmann machine," in *Advances in Neural Information Processing Systems 23*, J. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. Zemel, and A. Culotta, Eds., 2010, pp. 469–477.

[16] T. Mikolov, "Statistical language models based on neural networks," Ph.D. dissertation, Ph. D. thesis, Brno University of Technology, 2012.