FIFTH INTERNATIONAL WORKSHOP ON VIETNAMESE LANGUAGE AND SPEECH PROCESSING

VLSP 2018

PROCEEDINGS OF THE WORKSHOP

March 23, 2018
Hanoi, Vietnam

VLSP 2018 - Fifth International Workshop on Vietnamese Language and Speech Processing

In conjunction with 

Time 10h00 – 16h00, March 23, 2018
Location University of Science and Technology of Hanoi
18 Hoang Quoc Viet, Cau Giay, Hanoi, Vietnam

Workshop Co-chairs
Luong Chi Mai, Institute of Information Technology, VAST, Hanoi, Vietnam
Nguyen Thi Minh Huyen, VNU University of Science, Hanoi, Vietnam

Workshop Organizing Co-chairs
Ngo Xuan Bach, Posts and Telecommunications Institute of Technology, Hanoi, Vietnam
Nguyen Viet Son, Hanoi University of Science and Technology, Hanoi, Vietnam

Local Co-chairs
Nguyen Van Huy, Thai Nguyen University of Technology, Thai Nguyen, Vietnam
Tran Mai Vu, University of Engineering and Technology, VNU, Hanoi, Vietnam

Program Committee
1. Vu Hai Quan, University of Science, VNU-HCM, Vietnam (co-chair)
2. Le Anh Cuong, Ton Duc Thang University, Ho Chi Minh city, Vietnam (co-chair)
3. Le Hong Phuong, University of Science, VNU, Hanoi, Vietnam
4. Le Thanh Huong, Hanoi University of Science and Technology, Vietnam
5. Luong Chi Mai, Institute of Information Technology, VAST, Vietnam
6. Ngo Xuan Bach, Posts and Telecommunications Institute of Technology, Hanoi, Vietnam
7. Nguyen Le Minh, JAIST, Japan
8. Nguyen Phuong Thai, University of Engineering and Technology, VNU, Hanoi, Vietnam
9. Nguyen Van Huy, Thai Nguyen University of Technology, Thai Nguyen, Vietnam
10. Nguyen Thi Minh Huyen, University of Science, VNU, Hanoi, Vietnam
11. Nguyen Van Vinh, University of Engineering and Technology, VNU, Hanoi, Vietnam
12. Nguyen Viet Cuong, HPC Systems, Inc., Japan
13. Nguyen Viet Son, Hanoi University of Science and Technology, Vietnam
14. Phan Xuan Hieu, University of Engineering and Technology, VNU, Hanoi, Vietnam
15. Tran Do Dat, Ministry of Science and Technology, Vietnam
16. Tran Mai Vu, University of Engineering and Technology, VNU, Hanoi, Vietnam

Workshop Program

10:15 - 10:25	Opening
10:25 - 10:40	Nguyen Viet Son (HUST): *Text To Speech summary report* - Award presentation
10:40 - 10:55	Nguyen Van Huy (TNUT): *Automatic Speech Recognition summary report* - Award presentation
10:55 - 11:10	Ngo Xuan Bach (PTIT): *Sentiment Analysis summary report* - Award presentation
11:10 - 11:25	Tran Mai Vu (VNU-UET): *Named Entity Recognition summary report* - Award presentation
11:25 - 11:45	Coffee Break
11:45 - 12:10	Nguyen Quoc Bao (Viettel-VCC): *VLSP 2018 - Development of a Vietnamese Large Vocabulary Continuous Speech Recognition and a Vietnamese Speech Synthesis System*
12:10 - 12:25	Nguyen Tien Thanh (HUST): *MICATTS: Non-uniform unit selection speech synthesis system for Vietnamese*
12:25 - 12:50	Do Quoc Truong (VAIS): *VAIS-Speech: An Overview of Automatic Speech Recognition and Text-to-speech Development at VAIS*
12:50 - 14:00	Lunch
14:00 - 14:15	Vu Anh (1link): *A System for Aspect Based Sentiment Analysis at VLSP 2018 Evaluation Campaign*
14:15 - 14:30	Dang Van Thin (VNU-HCM UIT): *NLP@UIT at VLSP 2018: A Supervised Method for Aspect Based Sentiment Analysis*
14:30 - 14:45	Pham Quang Nhat Minh (Alt VN): *A Feature-Based Model for Nested Named-Entity Recognition at VLSP-2018 NER Evaluation Campaign*
14:45 - 15:00	Luong Viet Thang (VNG): *ZA-NER: Vietnamese Named Entity Recognition at VLSP 2018 Evaluation Campaign*
15:00 - 15:20	Coffee Break
15:20 - 16:00	Panel Discussion and Closing

Table of Contents

A Feature-Based Model for Nested Named-Entity Recognition at VLSP-2018 NER Evaluation Campaign

Pham Quang Nhat Minh
Alt Vietnam Co., Ltd
92 Trieu Viet Vuong, Hai Ba Trung, Hanoi
pham.minh@alt.ai

Abstract

In this report, we describe our participant named-entity recognition system at VLSP 2018 evaluation campaign. We formalized the task as a sequence labeling problem using BIO encoding scheme. We applied a feature-based model which combines word, word-shape features, Brown-cluster-based features, and word-embedding-based features. We compare several methods to deal with nested entities in the dataset. We showed that combining tags of entities at all levels for training a sequence labeling model (joint-tag model) improved the accuracy of nested named-entity recognition.

1 Introduction

Named-entity recognition (NER) is an important task in information extraction. The task is to identify in a text, spans that are entities and classify them into pre-defined categories. There have been some conferences and shared tasks for evaluating NER systems in English and other languages, such as MUC-6 (Sundheim, 1995), CoNLL 2002 (Sang, 2002) and CoNLL 2003 (Sang and Meulder, 2003).

In Vietnamese language, VLSP 2016 NER evaluation (Huyen and Luong, 2016) is the first evaluation campaign that aims to systematically compare NER systems for Vietnamese language. Similar to CoNLL 2003 shared-task, in VLSP 2016, four named-entity types were considered: person (PER), organization (ORG), location (LOC), and miscellaneous entities (MISC). In VLSP 2016, organizers provided the training/test with gold word segmentation, PoS and chunking tags. While that setting can help participant teams to reduce effort of data processing and solely focus on developing NER algorithms, it is not so realistic setting. In VLSP 2018 NER evaluation, only raw texts with XML tags were provided. Therefore, we need to choose appropriate Vietnamese NLP tools for preprocessing steps such as word segmentation, PoS tagging, and chunking.

In the report, we describe our NER system at VLSP 2018 NER evaluation campaign. We applied a feature-based model which combines word, word-shape features, Brown-cluster-based features, and word-embedding-based features and adopted Conditional Random Fields (CRF) (Lafferty et al., 2001) for training and testing.

In the VLSP 2018 NER task, similar as VLSP 2016, there are nested entities the NER dataset. An entity may contain other entities inside them. We categorize entities in VLSP 2018 NER dataset into three levels.

- Level-1 entities are entities that do not contain other entities inside them. For example: <ENAMEX TYPE="LOC">Hà Nội</ENAMEX>.
- Level-2 entities are entities contain only level-1 entities inside them. For example: <ENAMEX TYPE="ORG">UBND thành phố <ENAMEX TYPE="LOC">Hà Nội</ENAMEX></ENAMEX>.
- Level-3 entities are entities that contain at least one level-2 entity and may contain some level-1 entities. For example <ENAMEX TYPE="ORG">Khoa Toán, <ENAMEX TYPE="ORG">ĐHQG <ENAMEX TYPE="LOC">Hà Nội</ENAMEX></ENAMEX></ENAMEX>

In our data statistics, we see that the number of level-3 entities is too small compared with the

Word	Level-1 Tag	Level-2 Tag	Joint Tag
ông	O	O	O+O
Ngô_Văn_Quý	B-PER	O	B-PER+O
-	O	O	O+O
Phó	O	O	O+O
Chủ_tịch	O	O	O+O
UBND	O	B-ORG	O+B-ORG
TP	B-LOC	I-ORG	B-LOC+I-ORG
Hà_Nội	I-LOC	I-ORG	I-LOC+I-ORG

Table 1: Generating joint-tags by combing entity tags at all levels of a token

number of level-1 and level-2 entities, so we decided to ignore them in building the model. We just consider level-1 and level-2 entities.

In order to deal with nested named-entities, we investigated two methods. The first method trains separated models for each level of entities. The second method trains a single model on the training data in which tags are generated by combing entity tags of entities of all levels. Table 1 shows an example of how we combined entity tags at all levels of a token to create join tags.

We showed that combining tags of entities at all levels for training a sequence labeling model (joint-tag model) improved the accuracy of nested named-entity recognition.

The rest of the paper is organized as follows. In section 2, we described our participant NER system. In section 3, we present our evaluation results. Finally, section 4 gives conclusions about the work.

2 System description

We formalize NER task as a sequence labeling problem by using the B-I-O tagging scheme and we apply a popular sequence labeling model, Conditional Random Fields to the problem. In this section, first we present how we preprocess the data and then present features that we used in our model.

2.1 Preprocessing

In our NER system, we performed sentence and word segmentation on the data. For sentence segmentation, we just used a simple regular expression to detect sentence boundaries that match the pattern: period followed by a space and upper-case character. Actually, to produce result submissions, we also try not to perform sentence segmentation.

For word segmentation, we adopted RDRsegmenter (Nguyen et al., 2018) which is the state-of-the-art Vietnamese word segmentation tool. Both training and development data are the converted into data files in CoNLL 2003 format with two columns: words and their BIO tags. Due to errors of word segmentation tool, there may be boundary-conflict problem between entity boundary and word boundary. In such cases, we decided to tag words as "O" (outside entity).

2.2 Features

Basically, features in the proposed NER model are categorized into word, word-shape features, features based on word representations including word clusters and word embedding. Note that, we extract unigram and bigram features within the context surrounding the current token with the window size of 5. More specifically, for a feature F of the current word, unigram and bigram features are as follows.

- **unigrams**: F[-2], F[-1], F[0], F[1], F[2]
- **bigrams**: F[-2]F[-1], F[-1]F[0], F[0]F[1], F[1]F[2]

2.2.1 Word Features

We extract word-identity unigrams and bigrams within the window of size 5. We use both word surfaces and their lower-case forms. Beside words, we also extract prefixes and suffixes of surfaces of words within the context of the current word. In our model, we use prefixes and suffixes of lengths from 1 to 4 characters.

2.2.2 Word Shapes

In addition to word identities, we use word shapes to improve prediction ability, especially for unknown or rare words and reduce data spareness problem. We used the same word shapes as presented in (Minh, 2018).

2.2.3 Brown cluster-based features

Brown clustering algorithm is a hierarchical clustering algorithm for assigning words to clusters (Brown et al., 1992). Each cluster contains words which are semantically similar. Output clusters are represented as bit-strings. Brown-cluster-based features in our NER model include whole bit-string representations of words and their prefixes of lengths 2, 4, 6, 8, 10, 12, 26, 20. Note that, we only extract unigrams for Brown-cluster-based features.

In experiments, we used the Brown clustering implementation of Liang (Liang, 2005) and applied the tool on the raw text data collected through a Vietnamese news portal. We performed word clustering on the same preprocessed text data which were used to generate word embeddings in (Le-Hong et al., 2017). The number of word clusters used in our experiments is 5120.

2.2.4 Word embeddings

Word-embedding features have been used for a CRF-based Vietnamese NER model in (Le-Hong et al., 2017). The basic idea is adding unigram features corresponding to dimensions of word representation vectors.

In the paper, we apply the same word-embedding features as in (Le-Hong et al., 2017). We generated pre-trained word vectors by applying Glove (Pennington et al., 2014) on the same text data used to run Brown clustering. The dimension of word vectors in 25.

3 Evaluation

3.1 Data sets

Table 2 showed the data statistics on training set, development set, and official test set. The number of organization entities (ORG) at level 3 is too small, so we only consider level-1 and level-2 entities in training and evaluation. Level-2 entities are almost of ORG types.

3.2 Evaluation Measures

We used Precision, Recall, F1 score as evaluation measures. Note that, due to the fact that word segmentation may cause boundary conflict between entities and words, we convert words in the data into syllables before we evaluate Precision, Recall, F1 scores.

We consider four entity types: LOCATION, MISCELLANEOUS, ORGANIZATION, and PERSON in evaluation, and use the evaluation script of CoNLL-2013 for evaluation.

3.3 NER models

For evaluation on the development set, we train three NER models as follows on the training data of VLSP 2018 NER task.

- Level-1 model is trained by using level-1 entity tags.
- Level-2 model is trained by using level-2 entity tags.
- Joint model is trained using joint tags which combine level-1 and level-2 tags of each word.

3.4 Results

Table 3 and Table 4 shows the evaluation results on development set of recognizing level-1 and level-2 entities, respectively. The level-1 model obtained slightly better F1 score than joint model in recognizing level-1 entities while joint model outperformed level-2 model in recognizing level-2 entities. We also see that the level-2 model got higher precision than joint model but much lower recall than joint model. A plausible explanation for that phenomena is that information of level-1 tags helps to recognize more level-2 entities.

3.5 Result Submissions

We trained models on the data set obtained by combining provided training and development data and used the trained models for recognizing entities on the test set.

In order to produce submitted results, we use methods as follows.

- Using level-1 and level-2 model for recognizing level-1 and level-2 entities, respectively. We refer this method as **Separated** method.
- We use joint model to recognize joint tags for each word of a sentence, then split joint tags into level-1 and level-2 tags. We refer this method as **Joint** method.
- We use the joint model for recognizing level-2 entities and level-1 model for recognizing level-1 entities. We refer this method as **Hybrid** method.

Type	Train			Dev			Test		
	Level-1	**Level-2**	**Level-3**	**Level-1**	**Level-2**	**Level-3**	**Level-1**	**Level-2**	**Level-3**
LOC	8831	7	0	3043	2	0	2525	2	0
ORG	3471	1655	63	1203	690	14	1616	557	22
PER	6427	0	0	2168	0	0	3518	1	0
MISC	805	1	0	179	1	0	296	0	0
Total	19534	1663	63	6593	694	14	7955	561	22

Table 2: Number of entities of each type in each level in train/dev set

Model	Precision	Recall	F1
Level-1 Model	91.04	84.41	87.6
Joint Model	90.42	84.72	87.47

Table 3: Evaluation Results on dev set of recognizing level-1 entities

Method	Precision	Recall	F1
Level-2	85.81	72.44	78.56
Joint Model	84.36	77.06	80.54

Table 4: Evaluation Results on dev set of recognizing level-2 entities

Runs	Method	Sent Segmentation
Run-1	Hybrid	YES
Run-2	Hybrid	NO
Run-3	Joint	YES
Run-4	Joint	NO
Run-5	Separated	YES
Run-6	Separated	NO

Table 5: Six submitted runs

In recognition, there are some cases that predicted level-1 entities contains level-2 entities inside them. In such cases, we omit predicted level-2 entities inside predicted level-1 entities. The reason is that accuracy of level-1 entity recognition on dev set is much higher than the accuracy of level-2 entity recognition.

We submitted six runs at VLSP 2018 NER evaluation campaign as showed in Table 5. We try two preprocessing approaches: with sentence segmentation and without sentence segmentation. The reason why we try those preprocessing approaches is that we would like to know the influence of sequence lengths on the accuracy of our model.

Table 6 shows the official evaluation results for our six submitted runs. As indicated in the table, run 4 which uses **Joint** model obtained the highest F1 score among six runs. Using **Joint** model or **Hybrid** model obtained better F1 scores than using **Separated** methods. We also see that the difference between a system that performs sentence segmentation and a system that does not perform sentence segmentation is very small.

Run	Precision	Recall	F1
Run-1	76.08	70.68	73.28
Run-2	76.75	70.37	73.42
Run-3	76.32	70.25	73.16
Run-4	76.16	70.98	**73.48**
Run-5	75.70	70.28	72.89
Run-6	76.26	69.90	72.94

Table 6: Official Evaluation Results on Test set, which consider entities at all levels

Category	Precision	Recall	F1
PER	79.30	79.68	79.49
LOC	79.21	79.69	79.45
ORG	66.83	60.17	63.33
MISC	51.40	25.00	33.64
All	76.16	70.98	73.48

Table 7: Evaluation Results of Run-4 on Test set for each entity category

Run	Precision	Recall	F1
Run-1	73.82	79.43	76.52
Run-2	73.45	80.04	76.60
Run-3	73.21	79.56	76.26
Run-4	73.95	79.33	76.55
Run-5	73.80	79.46	76.53
Run-6	73.46	80.08	**76.63**

Table 8: Evaluation Results on Test set for level-1 entities

Table 7 shows the Precision, Recall, F1 scores for each entity category of run 4.

Table 8 and Table 9 showed the evaluation results on test set of six submitted runs for level-1 and level-2 entities, respectively.

Run	Precision	Recall	F1
Run-1	43.24	82.94	56.84
Run-2	43.06	82.59	56.61
Run-3	45.20	81.41	**58.12**
Run-4	44.48	82.51	57.80
Run-5	39.32	83.08	53.38
Run-6	36.83	84.15	51.24

Table 9: Evaluation Results on Test set for level-2 entities

Run-6 (using level-1 and level-2 models separately without sentence segmentation) obtained the best accuracy of recognizing level-1 entities among submitted runs (76.63%) and Run-3 (Joint model, sentence segmentation) obtained the best accuracy of recognizing level-2 entities (58.12%).

Using joint model obtained better F1 scores of recognizing both levels of entities than just those of the model trained on solely on level-1 and level-2 entity tags. That result is consistent with the result on the development set.

4 Conclusions

We haved presented a feature-based model for Vietnamese named-entity recognition and evaluation results at VLSP 2018 NER evaluation campaign. We compared several methods for recognizing nested entities and pointed out that the hybrid method obtained the best result among methods as presented above. As the future work, we plan to investigate deep learning methods such as BiLSTM-CNN-CRF (Ma and Hovy, 2016) for nested named entity recognition.

References

Peter F. Brown, Peter V. deSouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. 1992. Class-based n-gram models of natural language. *Comput. Linguist.*, 18(4):467–479.

Nguyen Thi Minh Huyen and Vu Xuan Luong. 2016. Vlsp 2016 shared task: Named entity recognition. In *Proceedings of Vietnamese Speech and Language Processing (VLSP)*.

John Lafferty, Andrew McCallum, and Fernando Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *ICML*, pages 282–289.

Phuong Le-Hong, Quang Nhat Minh Pham, Thai-Hoang Pham, Tuan-Anh Tran, and Dang-Minh Nguyen. 2017. An empirical study of discriminative sequence labeling models for vietnamese text processing. In *Proceedings of the 9th International Conference on Knowledge and Systems Engineering (KSE 2017)*.

Percy Liang. 2005. *Semi-supervised learning for natural language*. Ph.D. thesis, Massachusetts Institute of Technology.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1064–1074.

Pham Quang Nhat Minh. 2018. A feature-rich vietnamese named-entity recognition model. *arXiv preprint arXiv:1803.04375*.

Dat Quoc Nguyen, Dai Quoc Nguyen, Thanh Vu, Mark Dras, and Mark Johnson. 2018. A Fast and Accurate Vietnamese Word Segmenter. In *Proceedings of the 11th International Conference on Language Resources and Evaluation (LREC 2018)*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Erik F. Tjong Kim Sang. 2002. Introduction to the conll-2002 shared task: Language-independent named entity recognition. *CoRR*, cs.CL/0209010.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *CoNLL*.

Beth Sundheim. 1995. Overview of results of the muc-6 evaluation. In *MUC*.

ZA-NER: Vietnamese Named Entity Recognition at VLSP 2018 Evaluation Campaign

Viet-Thang Luong
Zalo Group
VNG Corporation
Ho Chi Minh City, Viet Nam
thanglv2@vng.com.vn

Long Kim Pham
Zalo Group
VNG Corporation
Ho Chi Minh City, Viet Nam
longpk@vng.com.vn

***Abstract*—This paper describes our system which participates in NER Shared task of VLSP 2018 evaluation campaign. Our system is the combination of Bidirectional Long Short-Term Memory and Conditional Random Field. Moreover, we enhanced word embeddings with information from characters. With this system, we achieve a comparative results with 74% (Level 1) and 68% (Nested) F1-scores on the standard test set of VLSP 2018.**

***Index Terms*—NER, CRF, Bi-LSTM, named entity recognition, long short-term memory, conditional random field, character LSTM**

I. Introduction

Named Entity Recognition (NER) is a fundamental task in Natural Language Processing (NLP). The role of this task is to identify noun phrases in general text and classify them into predefined categories such as the names of persons, organizations, locations, expressions of times, monetary values. Therefore, NER system can supply useful information to support many NLP applications such as question answering system, automatic summarization, etc.

Basically, most the NER systems focus on applying machine learning approaches, which do not require deep language-specific knowledge. There are two methods to use machine learning in the NER systems. In the first method, we can select a subset of hand-crafted features and combine it with a sequence labeling algorithm such as Conditional Random Field (CRF) [8], Hidden Markov Model (HMM) [11], or Maximum Entropy Markov Model (MEMM) [2]. This method took a lot of attention from researchers in a long time and is especially suitable for the low-resource languages like Vietnamese. However, the accuracy of it is greatly influenced by carefully choosing hand-crafted features.

In the past few years, several neural architectures have been proposed for sequence labeling problems in the development of deep learning. With these architectures, we have the second method to solve NER task. The advantage of deep neural networks is the end-to-end learning ability which eliminates the process of designing hand-crafted features. On the other hand, the main disadvantage of neural architectures is the need of large labeled dataset in order to achieve the high accuracy. One solution to overcome this drawback is to use a good word embedding which is trained in a huge unlabeled data.

Recently, the Vietnamese Language and Speech Processing (VLPS) community has organized an evaluation campaign to systematically compare NER systems for Vietnamese language. In this shared task, NER systems are evaluated based on the ability to recognize four named entity types: persons (PER), organizations (ORG), locations (LOC), and miscellaneous entities (MISC). The data are collected from electronic news papers published on the webs.

In this paper, we apply a state-of-the-art sequence labeling neural network for the NER task. This network combines Bi-LSTM and CRF into the end-to-end learning system without using any hand-crafted features. Furthermore, we carefully investigate the effects of word embedding and how to enhance the word embeddings with information from characters of words. Our best system achieved an accuracy of 98.60% and an overall F1 score of 89.20% in VLSP NER Development set.

The remainder of the paper is organized as follows: in Section 2 we introduce our models for VLSP NER Shared task in this year. Section 3 describes the experiments and results. Finally, we draw some conclusions in Section 4.

II. Models

In this section we briefly outline fundamental concepts of recurrent neural networks such as LSTM and BiLSTM models. We also describe a hybrid architecture which combines Bi-LSTM with a CRF layer for Vietnamese Named Entity Recognition task as well as some extensions to this baseline architecture.

A. Bidirectional Long Short-Term Memory

Recurrent Neural Network (RNN) is a kind of neural networks where every node in hidden layers has a self-connection [4]. This creates an internal state of the network which enables it to model variable-length sequences. Because an RNN can take a sequence as input and produces a sequence of labels, it is very suitable for Named Entity Recognition Task. However, the traditional RNN is still hard to apply due to the vanishing and exploding gradient problems [1], i.e. the update values of network weights computed via error back-propagation may be too small (vanishing) or too big (exploding) when modeling long sequences.

Long Short-Term Memory (LSTM), a well-known variation of RNN, has been proposed as a solution to overcome the mentioned problems [6]. The key in LSTM network is the memory cells with gated access to store and get information. In addition, LSTM also includes input, forget, and output gates to interact with the memory cells.

Correct recognition of named entity in a sentence not only depends on the previous information but also future information. However, a traditional LSTM with a single layer can only predict the output at the current time based on the previous information. Therefore, **Bidirectional LSTMs (BiLSTM)** are designed to overcome this problem [9]. A Bidirectional LSTM contains two single LSTM networks including forward and backward LSTMs. The general architecture of a BiLSTM is illustrated in Figure 1.

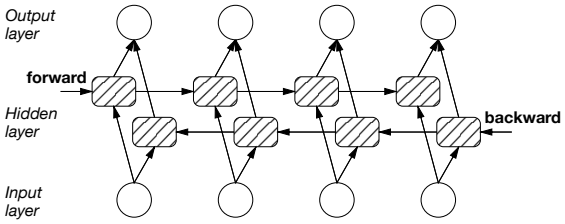

Fig. 1. BiLSTM architecture

B. Conditional Random Field

Conditional Random Field (CRF) is a probabilistic model for structured prediction which has been successfully applied to a variety of fields, such as computer vision, natural language processing. In CRF model, the inputs and outputs are directly connected, as opposed to LSTM and Bidirectional LSTM networks where memory cells components are employed. CRF can be used independently to solve the NER task [8].

C. BiLSTM-CRF

Inspired by the success of the combination of BiLSTM and CRF in English [9], we expect that it is also suitable to Vietnamese. The architecture of this model is presented in Figure 2. The BiLSTM-CRF network operates in the following process: (1) Word embedding is fed to the BiLSTM layer to extract some useful information about this word and the context around it; (2) Then, the CRF layer takes this information as features to predict NER tag of each word. This combination has been shown that it can take advantage of both these two methods.

The full set of parameters for this model consists of parameters of the BiLSTM layers (weight matrices, biases, word embedding matrix) and the transition matrix of the CRF layer. All these parameters are tuned during the training stage by the back-propagation algorithm with stochastic gradient descent. Dropout is applied to avoid over-fitting and improve the system performance.

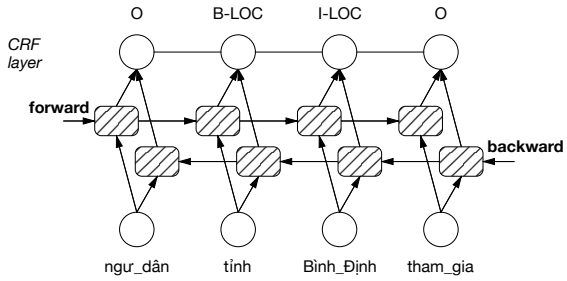

Fig. 2. BiLSTM+CRF architecture

D. Word Embedding

It has been shown in [3] that word embedding plays a vital role to improve sequence tagging performance. There are two mothods of initializing the values of word embedding for neural networks. Firstly, we can train the word embedding from a large unlabeled dataset in the same field with the labeled one. Secondly, we initialize the embedding matrix randomly and let the system learn this matrix during the training phase. In this paper, we try to use both methods.

While word embeddings capture the semantics of words to some degree, they may still suffer from the data sparsity problem. For example, they can not account for out-of-vocabulary words, misspellings, etc. We address this problem by using character-based word embeddings, which incorporate each individual character of a word to generate its vector representation [7]. The character-based word embeddings can obtained by using a BiLSTM network. Figure 3 illustrates the architecture of the word embedding.

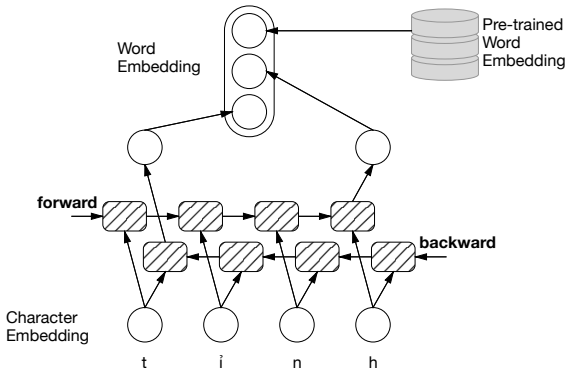

Fig. 3. Word Embedding Architecture

III. Experiments

A. Data sets

1) VLSP NER 2018 data: As mentioned before, we evaluate our neural network models on VLSP 2018. This data includes 1041 text documents in XML format. They contain 21499 sentences with more than 25856 named entities. The statistic summarization of the given data set is described in Table I.

TABLE I
THE STATISTIC OF VLSP 2018 NER DATA SET

	TRAIN	DEV	TEST	TOTAL
LOC	6289	795	2377	9461
MISC	743	63	178	984
ORG	5587	723	2126	8436
PER	4600	492	1883	6975
#total of NE	17219	2073	6564	25856
#sentence	14467	1601	5381	21499

2) BaoMoi data for training word embedding: To create good word embeddings for Vietnamese in in electronic newspapers, we have collected more than 720 thousand articles through BaoMoi news portal[1]. This data contains nearly 0.5 billion words. The text is first tokenized by Vitk[2] toolkit, and then we use FastText[3] and GloVe[4] to train two types of word embeddings. For words that appear in VLSP dataset but do not appear in word embeddings set, we create random vectors for these words by using Xavier Initialization algorithm [5].

B. Preprocessing

Since VLSP NER 2018 data is in XML format, we need to pre-process and convert it into CoNLL format. Firstly, we use some simple rule to split all paragraphs in a documents into a list of sentences. Secondly, we normalize punctuations and run the word segmentation toolkit. Finally, the processed sentences are converted to CoNLL format which is suitable for machine learning algorithms.

C. Evaluation metric

The performance is measured in Precision (P), Recall (R) and F1 score. Precision is the percentage of named entities found by our system that are corrected. Recall is the percentage of named entities in the test set that are recognized correctly by our system. F1 score is calculate by $\frac{2*P*R}{P+R}$. In the evaluation stage, we use a third-party evaluation tool, called *conlleval*[5], which is provided by CoNLL 2003 Shared Task [10].

D. Experiment settings

1) Overall Setting: Based on some experiments, we set the dimension of word and character embeddings to 100 and 25 respectively. Two BiLSTMs have the same dimension as these inputs. We train all our BiLSTM-CRF models using the back propagation method algorithms and update parameters on every training example using stochastic gradient descent (SGD) with a learning rate of 0.005. We also set dropout rate of the networks to 0.5. After 100 epochs, we choose the best model if they produce the best result on the development set. In the inference stage for test set, we use the Viterbi decoder to choose the best NER tag sequence.

[1]https://baomoi.com
[2]https://github.com/phuonglh/vn.vitk
[3]https://github.com/facebookresearch/fastText
[4]https://github.com/stanfordnlp/GloVe
[5]http://www.cnts.ua.ac.be/conll2003/ner/

*2) **Baseline**: Original BiLSTM and CRF architecture:* In the baseline system, we use BiLSTM to extract useful features from word and context, and feed them to CRF to predict NER tag. The word embedding matrix is initialized randomly and updated in the training process. The text input for Baseline system is not put through word segmentation.

*3) **Sys. 1**: Adding Character BiLSTM Layer:* Similar with Baseline system, we use BiLSTM and CRF to predict output tags. Besides, we add character BiLSTM layer to encode information from characters of words. The output of the character BiLSTM is concatenated with word embedding before putting to the BiLSTM-CRF network. We set the value of word and character embedding matrices randomly.

*4) **Sys. 2**: Effect of Word Segmentation:* We copy the network architecture of Sys. 1, but the text input is tokenized into the "true" word.

*5) **Sys. 3**: FastText Word Embedding:* In this system, we use the same model configuration as Sys. 2. However, we use FastText word embedding maxtrix that is trained in unlabelled BaoMoi dataset. The character embedding is still initialized randomly.

*6) **Sys. 4**: GloVe Word Embedding:* Instead of using FastText word embedding in Sys. 3, we change to using GloVe word embedding.

Table 2 summarizes the major different aspects of all systems.

TABLE II
SHORT DESCRIPTION OF EXPERIMENTAL MODELS

Model	Word Segmentation	Sub-word Level	Word Embedding
Baseline	No	No	No
Sys. 1	No	Character	No
Sys. 2	Yes	Character	No
Sys. 3	Yes	Character	FastText
Sys. 4	Yes	Character	GloVe

E. Experimental results

Table III shows the experimental results of all our systems on Precision, Recall and F1-Score. It is apparent from this table that the four modified systems outperformed the baseline system. The F1-score and Recall of Sys. 1 increased significantly compared to the baseline system. This indicates that the information from character is useful for NER task. Applying word segmentation to input in Sys. 2 also resulted in higher scores in all metrics than Sys. 1. Another significant improvement in F1-score is achieved by using the pre-trained word embeddings in Sys. 3 and 4 in comparison with Sys.2 (89.20, 88.63 vs 81.20, respectively).

F. VLSP Official Test Results

Table IV shows the official test results of VLSP 2018 NER Shared Task. Based on the previous experiments, we submitted Sys.3 and Sys.4 to the VLSP organizers.

TABLE III
THE OVERALL RESULTS

Model	Precision	Recall	F1
Baseline	81.14	73.03	76.98
Sys. 1	81.98	79.49	80.72
Sys. 2	82.49	80.15	81.30
Sys. 3	**89.02**	**89.38**	**89.20**
Sys. 4	88.19	89.08	88.63

TABLE IV
THE VLSP 2018 OFFICIAL TEST RESULTS

	Level 1 Evaluation			Nested Evaluation		
Model	**Precision**	**Recall**	**F1**	**Precision**	**Recall**	**F1**
Sys. 3 (1st)	77	70	74	71	65	68
Sys. 4 (2nd)	76	72	74	70	66	68

IV. CONCLUSION

In this work, we have adapted some parts of the well-known architecture combination of BiLSTM and CRF to the Vietnamese Named Entity Recognition Task at the VLSP 2018 evaluation campaign. Our best system uses BiLSTM to build a part of word embedding from the characters, then concatenate it with pretrained word embedding based on the FastText algorithm. These new word embeddings are fed to BiLSTM and CRF to predict the NER tags. This system achieved an overall F1 scores of 74% (Level 1) and 68% (Nested) on the standard VLSP NER Test set.

ACKNOWLEDGMENT

This work is sponsored in part by the Zalo NLP Research Program. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessary reflect the views of Zalo Group (VNG Corp.).

REFERENCES

[1] Yoshua Bengio, Patrice Simard, and Paolo Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, 5(2):157–166, 1994.

[2] Andrew Eliot Borthwick. *A Maximum Entropy Approach to Named Entity Recognition*. PhD thesis, New York, NY, USA, 1999. AAI9945252.

[3] Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537, 2011.

[4] Jeffrey L Elman. Finding structure in time. *Cognitive science*, 14(2):179–211, 1990.

[5] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256, 2010.

[6] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[7] Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. Character-aware neural language models. In *AAAI*, pages 2741–2749, 2016.

[8] John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, pages 282–289, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.

[9] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*, 2016.

[10] Erik F Tjong Kim Sang and Fien De Meulder. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003-Volume 4*, pages 142–147. Association for Computational Linguistics, 2003.

[11] GuoDong Zhou and Jian Su. Named entity recognition using an hmm-based chunk tagger. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, pages 473–480, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.

An investigation of Vietnamese Nested Entity recognition Models

Ngan T.Dong
FPT Technology Innovation Department,
17 Duy Tan, Ha Noi, Vietnam
Ngandt3@fpt.com.vn

***Abstract*— Many named entities contain other named entities inside them. Nested entity recognition is the task of identifying all named entities as well as all of their nested entity. The VLSP NER Evaluation Campaign 2018 (VLSP NER Eval 2018) is a nested entity recognition problem. In this report, we summarize our experiments, gains and insights when participating in the VLSP NER Eval 2018. The report is divided into 4 sections. The first section presents a short introduction about the task. The second section gives an overview of the data given. Section 3 presents our experimental approaches, experimental setup and the experimental results. Section 4 is for conclusion.**

Keywords—NER, nested entity, LSTM, CRF

I. INTRODUCTION

Many entities contains other named entities inside them. An ORGanization name might contains a PERson name or a LOCation, a LOCation name might contain other LOCation name, etc. Examples of the nested entities types in the VLSP 2018 NER Evaluation Campaign 2018 can be found in table 4. While many research group with many powerful systems have been proposed for the Named Entity Recognition problem, the problem of identifying named entities inside named entities is totally ignore. We believe VLSP 2018 triggers the first moves in the field of nested named entity recognition in Vietnamese text. The VLSP 2018 NER Eval 2018 task can be formulated as an annotation problem. Given an input text, annotate all person, organization, location names in that text. Data contains nested entities and requires those nested entities to be annotated. Since most state-of-the-art systems in Vietnamese NER recognition task are based on CRF and LSTM or both, we decided to construct our experiments on those two methods.

II. THE VLSP 2018 DATASET

The data is given in plain text format where each entity is encapsulated by its appropriate annotation. The data set contains 4 tags: PER for person, LOC for location, ORG for organization and MISC for miscellaneous. We are given the training and development set. We further spend 10% of the training set for testing purpose. Looking at the training and development data sets we can observe that there are at most three layers nested in a Named Entity. The summarization of all the nested types and their frequencies is presented in table 4. We can see that there is only a small number of nested entities compared to one layer entity. And the number of two-layer entities is much bigger than the number of three-layer one. Statistics about the training/testing/development data is given in table 3.

III. APPROACHES AND EXPERIMENTAL RESULT

A. Approaches

Since most of state-of-the-art systems for NER recognition task are based on CRF or LSTM or both, we decided to construct our experiments based on those. We tried three approaches: the first one is based on CRF alone, the second one and the third one are based on both LSTM and CRF. All three methods use word, POS features as well as features based on Word Embedding. Details about the methods and experimental setup is presented below.

1) The CRF-based approach

The proposed system is an ensemble of classifiers, one for each type of entity. Since we have 4 types of entity, we have 4 classifiers. The output of the system is a combination of output from the four classifiers. All classifiers use the same engine as described in [1] with different training data. We use 10% of the given training dataset for testing. 90% of the given training data and the development data is used for training.

2) The LSTM-based approach for each type of entity

The second proposed system is based on LSTM and CRF. We use the same method as described in [2]. We also have 4 classifiers based on the same engine but with different training set. However, in this experiment, we use only 90% of the training data fort training, the remaining 10% is for testing and the provided development set is used for validation purpose. Beside word, word embedding and POS features, this method also takes word shape, character embedding features into account. Details about network architecture, the features used can be found in [2].

3) The joint model approach

We use the multi-lstm model as described in [2] with 3 output layers. In this experiment we also use 10% of the given training set for testing, the development set for validation. The model also use character embedding feature. Details about the network architecture can be found in [2].

B. The Experimental setup

In all three experiments, the data will first be preprocessed to remove special character, standardize its diacritics. After that, they will be fed into the sentence, word segmentation and POS tagging modules. We use the VnCoreNLP[4] for the segmentation and POS tagging tasks. The data is then converted to CONLL format. For the Brown clusters and word vectors, we use the same resources as described in [1]. Since we observe that each entity has at most 3 nested layers, input to the joint model (the third approach) is a CONLL formatted file with 5 columns. The first two columns are for words and POS tags. The next three columns are for nested entity tags where the fifth column is for the outer most entity level. We use the same setup as described in [1] for the CRF-based approach. For the second and third approaches, we use stochastic gradient descent (SGD) with a learning rate of 0.02 to train the network, word dimension set to 25, use a 25 dimension pre-trained Glove word vectors, dimension for POS tags in the second experiment is set to 30 and in the third experiment is set 10, dimension for character embedding is set to 25.

C. Experimental result

Since two out of three proposed systems are ensembles of one-class classifiers and we don't have time to combine the results to get the system overall performance on our test set, we only report our result on the VLSP NER Eval 2018 test set. Details about the results can be found in table 1 and 2.

TABLE 1. OVERALL RESULTS ON VLSP NER EVAL 2018 TEST SET

	Precision	Recall	F1
CRF-based	0.74	0.51	0.61
1-layer LSTM	0.66	0.46	0.54
3 layers LSTM	0.64	0.44	0.52

We can see that approach 1 perform the best, even on the nested entity. That might be because word and word shape features play an important role in NER task. We have expected the third one to perform better on nested entity. We haven't had time to tune up our model in the second and third approaches due to the time constraint and the time needed to train a new model. After looking at the data statistics, we have to admit that the training set is too small to train a good joint LSTM model. The precision and recall rate of the first approach are still lower than we had expected. That might be because of the way we create the training data. We annotate a word with PER tag whenever we see a PER tag, not concerning whether it is encapsulated by an ORG tag or not. That might cause noise in training data and result in wrong prediction.

TABLE 2. DETAILS RESULT ON THE VLSP NER EVAL 2018 TEST SET

	PER	LOC	ORG	MIS.	PER -nest	LOC -nest	ORG -nest	MIS. -nest
CRF	0.77	0.78	0.66	0.63	0.00	0.00	0.64	0.00
	0.60	0.65	0.28	0.13	0.00	0.00	0.34	0.00
	0.67	0.71	0.40	0.22	0.00	0.00	0.45	0.00
LSTM1	0.70	0.74	0.52	0.56	0.00	0.00	0.51	0.00
	0.60	0.53	0.22	0.11	0.00	0.00	0.30	0.00
	0.64	0.61	0.31	0.19	0.00	0.00	0.38	0.00
LSTM2	0.68	0.69	0.49	0.43	0.00	0.00	0.59	0.00
	0.50	0.55	0.27	0.23	0.00	0.00	0.25	0.00
	0.58	0.61	0.35	0.30	0.00	0.00	0.35	0.00

IV. CONCLUSION

In this report we have presented our experiments in the VLSP NER Eval 2018. The best performed system achieves a 74% of overall precision on the test set. However, the performance of the system on nested entity is not that high. We observe a lot of wrong prediction and the highest recall rate documented is only 34 %. Though the highest documented system is a one-class classifier based on CRF we still question whether it is the best approach for nested entity recognition problem. One notable problem with that kind of system is that for nested entity of the same type like LOC-LOC, it will fail to annotate the nested one. We can understand why the performance of the third proposed system is not as expected. The two reasons might be because of the number of tags in layer 1 and 2 are not much, plus we use a bi-direction LSTM which requires large training dataset. All thing considers, though NER recognition has long been researched, further research efforts still needed in solving the nested entity recognition problem.

V. REFERENCES

[1] Le-Hong, P., Pham, Q.N.M., Pham, T.H., Tran, T.A., Nguyen, D.M.: "An empirical study of discriminative sequence labeling models for vietnamese text processing". In: Proceedings of the 9th International Conference on Knowledge and Systems Engineering (KSE 2017). (2017)

[2] Nguyen Truong Son, Nguyen Le Minh, "Nested named entity recognition using multilayer recurrent neural networks", PACLING 2017, August 16 - 18, 2017, Sedona Hotel, Yangon, Myanmar

[3] Pennington, J., Socher, R., Manning, C.D.: "Glove: Global vectors for word representation". In: Empirical Methods in Natural Language Processing (EMNLP). (2014) 1532–1543

[4] Thanh Vu, Dat Quoc Nguyen, Dai Quoc Nguyen, Mark Dras and Mark Johnson. 2018. "VnCoreNLP: A Vietnamese Natural Language Processing Toolkit". *arXiv preprint* arXiv:1801.01331.

TABLE 3. STATISTICS ABOUT NESTED ENTITY BY LAYER

		No. of PER tag	No. of ORG tag	No. of LOC tag	No. of MISC tag
Train	Layer1	3	1	47	0
	Layer2	54	82	1395	0
	Layer3	5732	4545	6446	720
Dev	Layer1	2	0	12	0
	Layer2	40	24	657	2
	Layer3	2126	1883	2377	178
Test	Layer1	0	1	11	0
	Layer2	7	17	153	0
	Layer3	631	543	786	86

TABLE 4. A SUMMARIZATION OF NESTED ENTITY TYPES AND THEIR FREQUENCY

Pattern	Train Freq.	Dev Freq.	Test Freq.	Example
LOC-LOC	14	3	0	<LOC>Sở LĐ-TB-XH <LOC>Bình Định</LOC></LOC>
LOC-ORG	13	1	0	<LOC><ORG>Shanghai</ORG> <LOC>Shenhua</LOC></LOC>
LOC-PER	3	1	0	<LOC>phường <PER>Tăng Nhơn Phú</PER> A</LOC>
MISC-LOC	2	3	0	CLB "<MISC>Về với quê mình <LOC>Quảng Ngãi</LOC></MISC>"
ORG-LOC-LOC	3	0	0	<ORG>Phòng Cảnh sát môi trường, <LOC>Công an <LOC>tỉnh Đắk Lắk</LOC></LOC></ORG>
ORG-LOC	2906	1356	318	<ORG>Sở LĐ-TB-XH <LOC>Bình Định</LOC></ORG>
ORG-MISC	0	5	0	<ORG>Nhà máy thép <MISC>Việt Trung</MISC></ORG>
ORG-ORG-LOC	104	29	23	<ORG>khoa Giáo dục tiểu học, <ORG>đại học Sư phạm <LOC>Hà Nội</LOC></ORG></ORG>
ORG-ORG-ORG	1	0	1	<ORG>Cục Trinh sát và Văn phòng <ORG>Bộ Tư lệnh <ORG>BĐBP</ORG></ORG></ORG>
ORG-ORG-PER	6	6	0	<ORG>Hội sinh viên <LOC>Việt Nam</LOC> tại <ORG>đại học <PER>La Trobe</PER></ORG></ORG>
ORG-ORG	179	55	45	<ORG>Trung tâm Ung bướu <ORG>Bệnh viện Quân y 175</ORG></ORG>
ORG-PER	131	92	21	<ORG>Trường THPT <PER>Lê Viết Thuật</PER></ORG>

VLSP 2018 Shared Task: Aspect Based Sentiment Analysis

Ngo Xuan Bach*, Tran Mai Vu† Le Anh Cuong‡, Vu Xuan Luong§, Nguyen Thi Minh Huyen¶
*Department of Computer Science,
Posts and Telecommunications Institute of Technology, Hanoi, Vietnam
bachnx@ptit.edu.vn
†Faculty of Information Technology,
VNU University of Engineering and Technology, Hanoi, Vietnam
vutm@vnu.edu.vn
‡Faculty of Information Technology,
Ton Duc Thang University, Ho Chi Minh City, Vietnam
leanhcuong@tdt.edu.vn
§Vietnam Lexicography Center, Hanoi, Vietnam
vuluong@vietlex.com
¶Faculty of Mathematics, Mechanics, and Informatics,
VNU University of Science, Hanoi, Vietnam
huyenntm@hus.edu.vn

***Abstract*—This paper describes the VLSP 2018 shared task on Aspect Based Sentiment Analysis (ABSA) for Vietnamese, a continuation and extension of the sentiment analysis task of VLSP 2016. This year, the task provided training, development, and test datasets in restaurant and hotel domains, as well as a common evaluation procedure. The task attracted submissions from 3 teams with promising results.**

***Keywords*—VLSP Workshop, Shared task, Aspect Based Sentiment Analysis, Opinion Mining.**

I. Introduction

With the development of technology and the Internet, different types of social media such as social networks and forums have allowed people to not only share information but also to express their opinions and attitudes on products, services and other social issues. The Internet becomes a very valuable and important source of information. People nowadays use it as a reference to guide their decisions when buying a product or using a service. Moreover, this kind of information also lets manufacturers and service providers receive feedback about limitations of their products and therefore improve them to better meet their customers needs. Furthermore, it can also help authorities know the attitudes and opinions of their residents on social events so that they can make appropriate adjustments.

Since the early 2000s, opinion mining and sentiment analysis [1] have become a new and active research topic in Natural language processing and Data mining. Topics in this field include:

- Subjective classification: this is the task of detecting whether a document contains personal opinions or not (only provides facts).
- Polarity classification (Sentiment classification): classify the opinion expressed in a document into one of three types, which are "positive", "negative" and "neutral".
- Spam detection: detect fake reviews and reviewers.
- Rating: reflect the personal opinion expressed in a document as a rating from 1 star to 5 stars (very negative to very positive).
- Opinion summarization: generate effective summaries of opinions so that users can get a quick understanding of the underlying sentiments.

Recently, Aspect Based Sentiment Analysis (ABSA), which is the task of mining and summarizing opinions from text about specific entities and their aspects, has been attracting more research. ABSA for English and other languages (but not Vietnamese) was introduced as a SemEval task in 2014 [4], 2015 [5], and 2016 [6]. These ABSA tasks provide benchmark datasets of reviews and evaluation frameworks in which the datasets were annotated with opinion target expressions and sentiment polarities.

The first related campaign for Vietnamese language sentiment analysis was organized in VLSP 2016, which only focused on polarity classification. The dataset consisted of short reviews annotated with one of three labels: "positive", "negative" and "neutral". The sentiment analysis (SA) task this year at VLSP 2018 addresses the problem of ABSA for Vietnamese, in which we are given a review and the task is how to determine aspects assigned with the corresponding sentiment polarities. We call this task VABSA (i.e. Vietnamese Aspect Based Sentiment Analysis).

The rest of this paper is structured as follows. Section 2 introduces the task. Datasets and evaluation measures are presented in Section 3 and Section 4, respectively. Section 5 describes submissions and results of participants. Finally, conclusions are given in Section 6.

Example 1: Restaurant domain.
#lozi #lozisaigon #chaovit #anchinh ngon - bổ - rẻ. Khuyết là hơi xa trung tâm
(delicious - good for health - cheap, far from center)
{FOOD#PRICE, positive},
{FOOD#QUALITY, positive},
{LOCATION#GENERAL, negative}

Example 2. Hotel domain.
Phòng ốc sạch, giường thoải mái, nhân viên thân thiện.
(clean rooms, comfortable beds, friendly staffs)
{ROOMS#CLEANLINESS, positive},
{ROOMS#COMFORT, positive},
{SERVICE#GENERAL, positive}

Fig. 1: Examples of input reviews and expected outputs.

II. Task Description

This task is similar to the Subtask 2 (slot 1 and slot 3) of the SemEval 2016 Task 5. Given a customer review about a target entity, the goal is to identify a set of {*aspect*, *polarity*} tuples that summarize the opinions expressed in the review. *aspect* is a pair of *entity–attribute*, while *polarity* can be "positive", "negative" or "neutral".

The task considers reviews in two domains: Restaurant and Hotel. Figure 1 shows two examples of input reviews in the two domains and expected outputs. In Example 1, the goal is to recognize the following three tuples:

1) {*aspect* = FOOD#PRICE, *polarity* = positive},
2) {*aspect* = FOOD#QUALITY, *polarity* = positive},
3) {*aspect* = LOCATION#GENERAL, *polarity* = positive}.

Similarly, in Example 2, we aim to extract the following three tuples:

1) {*aspect* = ROOMS#CLEANLINESS, *polarity* = positive},
2) {*aspect* = ROOMS#COMFORT, *polarity* = positive},
3) {*aspect* = SERVICE#GENERAL, *polarity* = positive}.

The task is divided into two subtasks (two phases):

- **Phase A (Aspect)**: The participants are required to identify aspects (entity-attribute) only.
- **Phase B (Aspect-Polarity)**: The participants are required to identify both aspects and sentiment polarities.

III. Datasets

A. Data Collection

Raw data were crawled from https://lozi.vn/ (for restaurant) and https://www.booking.com/ (for hotel). We collected reviews from hotels in Ha Noi, Da Nang, and Ho Chi Minh City (150 hotels in each city). We got 4751 reviews for restaurant domain and 5600 reviews for hotel domain.

	GENERAL	PRICES	QUALITY	STYLE& OPTIONS	MISCELLANEOUS
RESTAURANT	✓	✓	✗	✗	✓
FOOD	✗	✓	✓	✓	✗
DRINKS	✗	✓	✓	✓	✗
AMBIENCE	✓	✗	✗	✗	✗
SERVICE	✓	✗	✗	✗	✗
LOCATION	✓	✗	✗	✗	✗

Fig. 2: Possible entity-attribute pairs for restaurant domain.

B. Annotation Procedure

Data were annotated by three people. For each domain, we divided the dataset into two subsets. First, two annotators were asked to identify aspects and polarities in two subsets (each annotator for one subset). Then, the third annotator checked labeled data. If annotators disagreed on an assignment, three people were asked to examine and make the final decision.

In the following, we describe the set of aspects for each domain.

- **Aspects for restaurant domain**: entities can be RESTAURANT (in general), AMBIENCE, LOCATION, FOOD, DRINKS, or SERVICE; attributes can be GENERAL, QUALITY, PRICE, STYLE_OPTIONS, or MISCELLANEOUS.
  The possible combinations of these entities and attributes are given in Figure 2. Totally, we have 12 aspect categories for restaurant domain.
- **Aspects for hotel domain**: entities can be HOTEL (in general), ROOMS, ROOM_AMENITIES, FACILITIES, SERVICE, LOCATION, or FOOD&DRINKS; attributes can be GENERAL, PRICES, DESIGN&FEATURES, CLEANLINESS, COMFORT, QUALITY, STYLE&OPTIONS, or MISCELLANEOUS.
  The possible combinations of these entities and attributes are given in Figure 3. Totally, we have 34 aspect categories for hotel domain.

C. Training, Development, and Test

For each domain, data were divided into three datasets: training, development, and Test. Training and development datasets were used to build participating systems. Test dataset was used for evaluation purpose. Table I shows the number of reviews and aspects in each dataset.

IV. Evaluation Measures

The performance of participating systems will be evaluated in two phases.

	GENERAL	PRICES	DESIGN& FEATURES	CLEANLINESS	COMFORT	QUALITY	STYLE& OPTIONS	MISCELLANEOUS
HOTEL	✓	✓	✓	✓	✓	✓	✗	✓
ROOMS	✓	✓	✓	✓	✓	✓	✗	✓
ROOM_ AMENITIES	✓	✓	✓	✓	✓	✓	✗	✓
FACILITIES	✓	✓	✓	✓	✓	✓	✗	✓
SERVICE	✓	✗	✗	✗	✗	✗	✗	✗
LOCATION	✓	✗	✗	✗	✗	✗	✗	✗
FOOD& DRINKS	✗	✓	✗	✗	✗	✓	✓	✓

Fig. 3: Possible entity-attribute pairs for hotel domain.

TABLE I: Statistical information of training, development, and test datasets

Domain	Dataset	#Reviews	#Aspects
Restaurant	Training	2961	9034
	Development	1290	3408
	Test	500	2419
Hotel	Training	3000	13948
	Development	2000	7111
	Test	600	2584

A. Phase A: Aspect (Entity-Attribute)

The F_1 score will be calculated for aspects only. Let A be the set of predicted aspects (entity-attribute pairs), and B be the set of annotated aspects, precision, recall, and the F_1 score can be computed as follows:

$$Precision = \frac{|A \cap B|}{|A|},$$

$$Recall = \frac{|A \cap B|}{|B|},$$

$$\text{and } F_1 = \frac{2 * Precision * Recall}{Precision + Recall}.$$

B. Phase B: Full (Aspect-Polarity)

The F_1 score will be calculated for both aspects and sentiment polarities. Let A be the set of predicted tuples (entity-attribute-polarity), and B be the set of annotated tuples, the precision, recall, and the F_1 score can be computed in a similar way as in Phase A.

V. Submissions and Results

We received submissions from 3 teams. Among them, two teams submitted technical reports and the other one sent us a short description. All teams considered the task as classification problems and exploited statistical machine learning algorithms to solve. In the next section, we summarize methods and results of 3 participating systems: SA1 from Van et al. [3], SA2 from Nguyen and Minh [2], and SA3 from Vu and Anh.

TABLE II: Learning algorithms and features used in participating systems

System	Learning Algorithms	Features
SA1	Linear SVM (sklearn-toolkit)	**Aspect**: n-grams, words, POS tags **Polarity**: n-grams, words, Elongate, Aspect Category, Count of the hagtags, Count of POS tags, Punctuation Marks
SA2	Multilayer Perceptron (scikit-learn library)	n-grams, tf-idf
SA3	Linear SVM	Count features (n-grams), tf-idf

A. Methods

While SA2 and SA3 considered the task as a multi-class classification problem (each label is a pair of *aspect-polarity*) and built only one classifier to solve the task, SA1 treated the task as multiple binary classification problems and built a single binary classifier for each aspect. To identify polarities of reviews, SA1 modeled the problem as a classification with three classes, i.e. positive, negative, and neutral.

Table II summarizes learning algorithms and features used in participating systems. While SA1 and SA3 used SVM with linear kernel, SA2 exploited multilayer perceptron algorithm. SA2 and SA3 built only one multi-class classifier with basic features, including n-grams and tf-idf scores. SA1 used more sophisticated features, such as elongate features, hagtags, punctuation marks. SA1 also conducted some preprocessing steps before training classification models.

B. Results

Tables III and IV summarize results of participating systems on development and test datasets, respectively[1]. For both domains, SA1 achieved the best F_1 scores on both development and test datasets. The results showed the effectiveness of sophisticated features used in SA1. Using linear SVM, SA1 and SA3 outperformed SA2 with multilayer perceptron significantly.

VI. Conclusion

We have presented the VLSP 2018 shared task on Aspect Based Sentiment Analysis for Vietnamese. The task attracted 3 teams, which is much smaller than the number of participants in the previous VLSP workshop, 8 teams. The reason might be that the task this year is more difficult than the previous one. We believe that the shared task provides useful resources for academic research as well as for building Vietnamese sentiment analysis systems. We hope to receive more attention from the research community and companies in the next VLSP workshop.

Acknowledgment

The organization of this shared task was partially supported by the following sponsors: Alt Vietnam, InfoRe and Zalo Careers. Great thanks to them! We would like to address our special thanks to Dr. Hoang Thi Tuyen Linh and Mr. Vu Hoang for their contribution to the data annotation process. And we thank to all the research teams for their participation to this competition.

[1]We did not receive precision and recall from SA2 team.

TABLE III: Results on development datasets

		Phase A (Aspect)			**Phase B (Aspect-Polarity)**		
Domain	**Team**	**Precision**	**Recall**	**F_1**	**Precision**	**Recall**	**F_1**
Restaurant	SA1	0.75	0.85	**0.79**	0.63	0.71	**0.67**
	SA2						0.59
	SA3	0.78	0.65	0.71	0.71	0.59	0.64
Hotel	SA1	0.75	0.64	**0.69**	0.67	0.58	**0.62**
	SA2						0.56
	SA3	0.83	0.51	0.63	0.78	0.48	0.60

TABLE IV: Results on test datasets

		Phase A (Aspect)			**Phase B (Aspect-Polarity)**		
Domain	**Team**	**Precision**	**Recall**	**F_1**	**Precision**	**Recall**	**F_1**
Restaurant	SA1	0.79	0.76	**0.77**	0.62	0.60	**0.61**
	SA2	0.88	0.38	0.54	0.79	0.35	0.48
	SA3	0.62	0.62	0.62	0.52	0.52	0.52
Hotel	SA1	0.76	0.66	**0.70**	0.66	0.57	**0.61**
	SA2	0.85	0.42	0.56	0.80	0.39	0.53
	SA3	0.83	0.58	0.68	0.71	0.49	0.58

REFERENCES

[1] B. Liu. Sentiment Analysis and Opinion Mining. Morgan & Claypool Publishers, 2012

[2] Tuan Anh Nguyen, Pham Quang Nhat Minh. Using Multilayer Perceptron for Aspect-based Sentiment Analysis at VLSP-2018 SA Task. In *Proceedings of the Fifth International workshop on Vietnamese Language and Speech Processing (VLSP)*, 2018.

[3] Thin Dang Van, Kiet Van Nguyen, Ngan Luu-Thuy Nguyen. NLP@UIT at VLSP 2018: A Supervised Method for Aspect Based Sentiment Analysis. In *Proceedings of the Fifth International workshop on Vietnamese Language and Speech Processing (VLSP)*, 2018.

[4] Pontiki et al. SemEval-2014 Task 4: Aspect Based Sentiment Analysis. In *Proceedings of SemEval-2014*, pp. 27–35, 2014.

[5] Pontiki et al. SemEval-2015 Task 12: Aspect Based Sentiment Analysis. In *Proceedings of SemEval-2015*, pp. 486–495, 2015.

[6] Pontiki et al. SemEval-2016 Task 5: Aspect Based Sentiment Analysis. In *Proceedings of SemEval-2016*, pp. 19–30, 2016.

NLP@UIT at VLSP 2018: A SUPERVISED METHOD FOR ASPECT BASED SENTIMENT ANALYSIS

Thin Dang Van
NLP@UIT Research Group*
Multimedia Communication Laboratory
University of Information Technology - VietNam National University Ho Chi Minh City
Ho Chi Minh, VietNam
thindv@uit.edu.vn

Kiet Van Nguyen
NLP@UIT Research Group*
Department of Information Science and Engineering
University of Information Technology - VietNam National University Ho Chi Minh City
Ho Chi Minh, VietNam
kietnv@uit.edu.vn

Ngan Luu-Thuy Nguyen
NLP@UIT Research Group*
Faculty of Computer Science
University of Information Technology - VietNam National University Ho Chi Minh City
Ho Chi Minh, VietNam
ngannlt@uit.edu.vn

***Abstract*—** In recent years, Sentiment Analysis has became one of the interesting research fields in Natural Language Processing. In this paper, we present the description of our system to address problem in Sentiment Analysis task at the VLSP shared task 2018: Aspect-based sentiment analysis, for two main sub-tasks on the different domain datasets. We used a supervised method based on the SVM classifier combined with a variety of features for both aspect detection and aspect polarity for two domains - the restaurant and hotel domain. Our results in aspect detection task is 77% of F1-score for the restaurant, while achieving 70% of F1-score for the hotel domain. In the task of aspect - polarity detection, we obtained the F1-score of 61% for two domains.

I. INTRODUCTION

Nowadays, the rapid development of Internet brings many opportunities and challenges for organizations which provide various products or services for people. In addition, the volume of Internet-users is growing rapidly and tends to increase more in near future. It is common that people refer the product (laptop, phone, etc.) or service (hotel, restaurant, etc.) reviews on the Internet before making decision. Besides, there are many individuals who are influenced by other people's reviews. Moreover, knowing user's likes and dislikes can be of great help in developing new products. For that reason, users' reviews play an important role and become a valuable information for companies, providers who are interested in users' opinions (Liu, 2012).

The website is one of the great platforms for users to immediately share their comments or experiences about several subjects. People can freely express their opinions about what they want to say on websites such as comment, evaluation on electrical devices, etc. Hence, the number of reviews of users is increasing day by day. For e-commerce companies, taking care of user feedback is a necessity and they usually have a team to analyze and evaluate users' reviews. However, with a large source of data currently available, the manual analysis is not feasible. Sentiment Analysis (SA) is the sub-field of Natural Language Processing that aims to extract and analyze subjective information from opinions, comments or reviews shared by human. Therefore, sentiment analysis has been studied very early in the world (Turney, 2002; Pang et al., 2002) and on Vietnamese, this research topic has became a trend since 2010 (Bang and Pham et al. 2010).

Aspect-based sentiment analysis (ABSA) is the sub-field of Sentiment Analysis which allows us deeply understand and determine sentiment in term of different aspects of an topic. An ABSA system must be able to classify each opinion according to the aspect categories and its polarity for the certain domain. Recently, this task was researched by scientists in the field of natural language processing via many shared tasks such as SemEval 2014 (Task 1), 2015 (Task 12) and 2016 (Task 5). These shared tasks focus on addressing the problem of aspect-based sentiment analysis for many languages such as English, Chinese, Arabic, etc.

For the low-resource languages such as Vietnamese, previous studies only focused on sentiment analysis in Vietnamese using methods such as rule based (Kieu and Pham et al. 2010) , Lexicon (Vu et Park et al. 2014a, Nguyen et al. 2014b, Trinh et al., 2016), semi-supervised learning (Ha et al. 2011, Le et al. 2015a), supervised learning (Phan and Cao 2014, Duyen et al. 2014, Nguyen et al. 2014a, Bach et al. 2015). In 2016, the VSLP community organized the first campaign related to Sentiment Analysis focused on the sentiment classification which contains three class polarity: positive, negative and neutral. This year 2018, ABSA standard datasets are annotated and released for two domains - the restaurant and the hotel.

In this paper, we describe our system to address aspect-based sentiment analysis problem at VLSP shared-task competition on two domains. The paper is structured as follows: the next section introduces the ABSA task in this campaign; Section 3 presents our approach while section 4 presents our experimental results. Section 5 concludes the

(*) NLP@UIT is a scientific research group on Natural Language Processing in University of Information Technology, Vietnam National University - Ho Chi Minh City. You can access information about our group at https://sites.google.com/uit.edu.vn/uit-nlp/.

work and describes the future enhancement directions to improve the classifiers on the two datasets.

II. ABSA SHARED-TASK

A. *Task Description*

This task is similar to the SemEval workshop 2016 Task 5 - slot 1 and slot 3 in the subtask 2. Given a set of customer reviews about a target entity (e.g. a hotel or a restaurant), the goal is to identify a set of tuples {aspect, polarity} that summarize the opinions expressed in each review. The aspect is identified by the tuple (entity, attribute). The polarity labels include the classes positive, negative and neutral. For example, in the restaurant dataset, "Khẩu vị khá ngon, không gian quán rộng view cũng tạm được. Đặc biệt là phục vụ rất nhiệt tình và vui vẻ" ("Taste is quite good, spacious space view is also temporary. Especially served very enthusiastic and fun"), the team's system have to return the list of {A#E, P} tuples as below: {E : FOOD # A: QUALITY, P : positive}, {E: SERVICE # A: GENERAL, P : positive}, {E: AMBIENCE # A: GENERAL, P : neutral}. Otherwise, in hotel domain, consider an example as follow: "Vị trí khách sạn đẹp; Có quán bar view đẹp; Nhân viên thân thiện" (Hotel's location is nice, there is a great bar's view; staff is friendly) should be assigned the tuples: "{LOCATION#GENERAL, positive}, {FACILITIES#GENERAL, positive}, {SERVICE#GENERAL, positive}". ure In this task, we participate in both two phrases and two domains.

B. *Datasets and Measures*

The dataset in ABSA task includes three parts: training, development and test dataset. All samples in this dataset is document-level reviews which is composed of many sentences with different length. This is a challenge that needs to be addressed in this year's shared task. Table 1 and Table 2 show the statistics of the dataset for the restaurant and hotel domains in detail.

Table 1: The statistics of datasets for the restaurant domain.

Dataset	#Reviews	#Aspect
Train	2961	9297
Development	1290	3443
Test	500	2419

Table 2: The statistics of datasets for the hotel domain

Dataset	#Reviews	#Aspect
Train	3000	13949
Development	2000	7111
Test	600	2584

To evaluate the performance of final system, we use the micro-averaged method which is distributed from the VLSP organizer. There are two phrases to measure the participating systems: Phrase A for aspect detection and Phrase B for aspect - polaririty. The precision, recall, and F1-score is calculated in a similar way in two phrases.

III. SYSTEM OVERVIEW

In this section, we make a brief description of how to build our ABSA system for VLSP shared-task.

A. *Preprocessing*

Preprocessing is one of the key components in a typical text classification framework. With dataset about reviews of the restaurant or hotel, people often do not pay attention to forms and sentences in writing, so there are many minor errors in original review (e.g the words stick together, special character, etc.). First of all, each review must be processed by various step as follow:

- **Step 1**: We observe that there are a lot of different money value characters which indicate money value in whole dataset. Therefore, we decide to replace symbols or words which refer the same object such as the value of money: 100k, 200d is replaced by special character as "giá_tiền" (money), "#lozi" is replaced by 'hag_tag', etc.
- **Step 2**: Delete special character (= , < , @ , $) in reviews expect punctuation mark. If following it is an upper word, we will insert dot before the word.
- **Step 3**: Split words which stick together "tả dcVới giá" and icon with a word (□Nem).
- **Step 4**: Change elongate word by true word ('ngoooon' (tasty) is normalized by 'ngon').
- **Step 5**: Because this dataset is crawled directly from the website, there are many freestyle letters in reviews. However, we just decide to replace one negation word which is used frequently in the whole training dataset. The word "không / no" can be written as '‘khong’, ‘ko’, ‘khg’, ‘k’, etc'.
- **Step 6**: After that, each review is broken into tokens and POS tagging by using Pyvi library[1].

On the other hand, we also manually create the food and drink dictionary based on part of speech of word. We extract all nouns in training dataset, then filter them in order to remove noisy words.

B. *Feature Selection*

B1. Aspect detection

The aim of this task is to assign to each review the list of entity - attribute pairs. Give a review, the system predicts the E#A pairs for that review. There are 12 different pairs in the restaurant domain and 34 pairs of entity#attribute in the hotel domain. Therefore, the problem in this task is the multi-label classification which each review can belong to one or more aspect categories. Grigorios T. et, al (2007) has grouped the methods for multi-label problem into two main categories: problem transformation methods and algorithm adaptation

[1] *Python Vietnamese Toolkit. https://pypi.python.org/pypi/pyvi*

methods [6]. To address it, we approach the method of problem transformation to transfer multi-label classification into multiple binary classification. Hence, we developed 12 binary classifications corresponding to 12 E#A pairs for restaurant domain and 34 binary classifications for hotel domain by using the linear SVM classifier. To implement the SVM classifier, we used the following features for each review:

- **Ngram**: Since proposed by Pang et al [2], n-grams have been used as basic feature in text classification. Therefore, unigram, bigram and trigram are extracted as feature for classifier.
- **Word**: All nouns, verbs and adjectives which appear in review are extracted.
- **POS tag**: Part-of-speech of all words - nouns, verbs or adjectives.

Then, we use the TF-IDF model to convert all features into numerical representation. Finally, we apply the linear SVM classifier to build the model for each category.

B2: Aspect Polarity

With each identified pair (entity#attribute) has to be assigned one of polarity labels: "positive", "negative" and "neutral". To tackle that challenge, we apply a supervised method to evaluate the efficient on development dataset and analyse errors in the result. In supervised learning, we continually choose the linear SVM classifier with diversity of features as follow:

- **N-gram**: In this feature, we also choose bigram, trigram as feature of n-gram.
- **Word feature** : we select all nouns, verbs, adjectives which appear in reviews is used as a feature.
- **Elongate Feature**: words contain a vowel that is repeated more than twice (example "ngooonnnn") are adopted.
- **Aspect Category:** we use the entity, attribute and the entity#attribute pair as a feature.
- **Count of the hagtag:** we calculate the number of hagtag words in review.
- **Count the POS feature:** we also calculate number of nouns, verbs, adjectives.
- **Punctuation Marks:** True if punctuation (!,?) marks present in reviews.

We consider this to be a multi-class problem which contains three class - positive, negative and neutral. Therefore, for each aspect category, if review contains that aspect, we will automatically assign its polarity as label of review. We adopt the linear SVM with the multi-class parameter "ovr" in sklearn-toolkit.

IV. EXPERIMENTS & DISCUSSION

In this section, we describe our experiments and results for the different tasks - aspect category detection and aspect polarity in two domain. The performance of system is evaluated in two phases : Phrase A: aspect (entity # attribute) and Phase B: full aspect-polarity (entity#attribute, polarity). We show the result of our system for aspect-based sentiment analysis in Table 4 and Table 5.

Table 4: The performance of our method for two domains on development dataset.

Domain	Phase	Precision	Recall	F1
Restaurant	A: E#A	0.75	0.85	**0.79**
	B:E#A, P	0.63	0.71	**0.67**
Hotel	A:E#A	0.75	0.64	**0.69**
	B:E#A, P	0.67	0.58	**0.62**

Table 5: The performance of our method for two domains on test dataset.

Domain	Phase	Precision	Recall	F1
Restaurant	A:E#A	0.79	0.76	**0.77**
	B:E#A, P	0.62	0.60	**0.61**
Hotel	A:E#A	0.76	0.66	**0.70**
	B:E#A, P	0.66	0.57	**0.61**

Table 4 presents the precision, recall and F-score of our system is measured in the development dataset, whereas Table 5 is the results on the test dataset for two domains. As can been seen, our system performs better for the restaurant domain on the development dataset but we get the same results for two domains in phrase B. As revealed by the results, in both Table 4 and Table 5, we notice that the recall is greater than the precision in the development, but the results are contradictory in the test. It depends on the length of the reviews in the development and the test dataset. By our observation, the length of review in test dataset is much longer than in development dataset (average length of reviews in development dataset is 58, while in test dataset is 158). For that reason, there is a big difference between two datasets for the restaurant domain. Figure 1 and Figure 2 illustrate our results on two dataset for two domains.

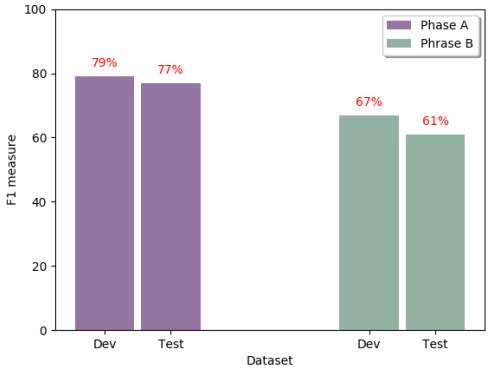

Figure 1: Performance of our system on the development and test dataset for the restaurant domain in two phrases.

When analysing the result of our system in aspect detection task for restaurant domain, we found that the entities {food, drink} and its attribute {quality, prices,styles&option} is concurrently assigned in many reviews. We predict that our drink and food dictionary is the main reason for this result in the test dataset. Therefore, our official results did not achieve the best results. For the hotel domain, there are 34 aspect categories which include the entity and the attribute. However, there are many aspects that appear very little in the entire training dataset but exist in test dataset. This is also one of the reasons why our results are not high on this domain.

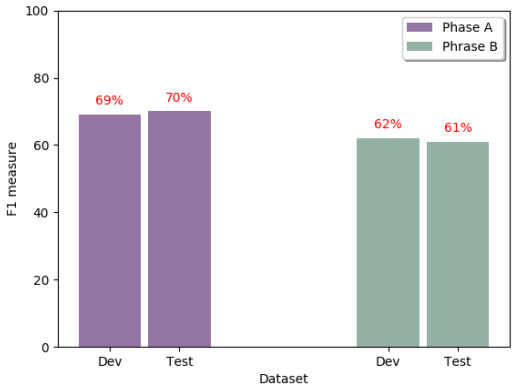

Figure 2: Performance of our system on the development and test dataset for the hotel domain in two phrases.

Finally, we achieved the F1-score of 77 % and the F1-score of 70% for the restaurant and hotel domain in phrase A, while in the phrase B, we also achieved the F1-score of 61% in two domains.

V. CONCLUSION & FUTURE WORK

In this paper, we have described our approach to solve the aspect-based sentiment analysis task proposed at the Sentiment Analysis Task of VLSP campaign 2018. We develop the ASBA system using supervised approach for aspect categories and its polarities. We participate in the two sub-tasks and evaluate the performance of our system for two domains. Our official result is 77% and 70% of F1-score in the aspect detection task for the restaurant and hotel domain, respectively. We also achieved the same F1-score of 61% for two domains in aspect and its polarity task.

As future works, we plan to exploit this problem in different ways to improve performance. We will investigate directions both in feature engineering and types of neural network models for this problem.In addition, we also analyse those dataset for two domains to select the efficient approach such as the hybrid approach which combines supervised method and rule heuristic to improve the result of classification.

ACKNOWLEDGMENT

We would like to thank the VLSP 2018 organizers for their sustained hard work, and providing datasets for this project.

REFERENCES

[1] Grigorios Tsoumakas and Ioannis Katakis . “Multi-Label Classification: An Overview”. International Journal of Data Warehousing and Mining 3, 1–13 (2007).

[2] B. Pang, L. Lillian, and V. Shivakumar, “Thumbs up?: Sentiment classification using machine learning techniques,” in Proc. ACL-02 Conf.Empirical Methods in Natural Language Process., vol. 10, pp.79–86, July, 2002.

[3] Maria Pontiki, Dimitrios Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. “Semeval-2014 Task 4:Aspect based sentiment analysis”. Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014), pages 27–35,Dublin, Ireland, August 23-24, 2014.

[4] M. Pontiki, D. Galanis, H. Papageorgiou, S. Manandhar and I. Androutsopoulos. “SemEval-2015 Task 12: Aspect Based Sentiment Analysis”. Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015) pp. 719 - 724, Denver, Colorado, June 4-5, 2015.

[5] Tamara Alvarez-Lopez, Jonathan Juncal-Martınez, Milagros Fernandez-Gavilanes, Enrique Costa-Montenegro and Francisco Javier Gonzalez-Castano. “GTI at SemEval-2016 Task 5: SVM and CRF for Aspect Detection and Unsupervised Aspect-Based Sentiment Analysis”. Proceedings of SemEval-2016, pages 306–311, San Diego, California, June 16-17, 2016.

[6] Maria Pontiki,Dimitrios Galanis,Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeny Kotelnikov, Nuria Bel,Salud María Jiménez-Zafra and Gülşen Eryiğit. “SemEval-2016 Task 5: Aspect Based Sentiment Analysis”. Proceedings of SemEval-2016, pages 19–30, San Diego, California, June 16-17, 2016.

[7] Kim Schouten and Flavius Frasincar. “Survey one Aspect-Level Sentiment Analysis”. IEEE Transactions on Knowledge and Data Engineering. DOI 10.1109/TKDE.2015.2485209.

[8] Liu. B 2011. "Opinion mining and sentiment analysis". In Web Data Mining. pages 459-526. Springer Berlin Heidelberg. 2009.

[9] Le Anh Cuong, Nguyen Thi Minh Huyen and Nguyen Viet Hung. "VLSP 2016 Shared Task: Sentiment Analysis". Fourth International Workshop on Vietnamese Language and Speech Processing (VLSP 2016).

[10] Toma˘s Hercig, Toma˘sBrychcın, Luka˘s Svoboda and Michal Konkol. “UWB at SemEval-2016 Task 5: Aspect Based Sentiment Analysis”. Proceedings of SemEval-2016, pages 342–349, San Diego, California, June 16-17, 201.

[11] Xinjie Zhou, Xiaojun Wan and Jianguo Xiao. “Representation Learning for Aspect Category Detection in Online Reviews”. Proceedings of the Twenty Ninth AAAI Conference on Artificial Intelligence. 2015.

Using Multilayer Perceptron for Aspect-based Sentiment Analysis at VLSP-2018 SA Task

Tuan Anh Nguyen and **Pham Quang Nhat Minh**
Alt Vietnam Co., Ltd
{nguyen.tuan.anh, pham.minh}@alt.ai

Abstract

In this report, we describe our system for sentiment analysis (SA) task at VLSP 2018 evaluation campaign. The goal of SA task is to classify documents, articles, or product reviews into classes that reflect their sentiments about some subject matters. We propose a Multiple Layers Perceptron (MLP) model for the task. We use n-gram features (n = 1,2,3,4) with TF-IDF weighting scheme to train this classifier. Our system obtained 48% F1-score on restaurant development dataset and 53% F1-score on hotel development dataset.

1 Introduction

Sentiment analysis is the task of mining opinions in sentences, users' reviews, or articles according their sentiment towards some subject matters such as products, services, etc. Sentiment analysis is useful in many business intelligence applications. For instance, sentiments of product reviews give us a quick summary of users' opinions about products. The task has received extensive attentions of natural-language-processing and data mining communities since early 2000s (Pang et al., 2002).

In Vietnamese Language and Speech Processing (VLSP) 2016, the SA shared task's goal is to classify Vietnamese reviews/documents into one of three categories: positive, negative, or neutral. This year, the SA task is more complicated, the goal is to classify Vietnamese reviews into one or more aspects and output the opinions respectively. We propose a Multilayer Perceptron (MLP) model trained with n-grams and TF-IDF features. We also discuss some errors that our system made and directions for future work.

The rest of the paper is organized as follows. In section 2, we describe our system. In section 3, we present our evaluation results on the test set. In section 4, we discuss about the errors and directions for further improvements.

2 System Description

We observe that the data given by VLSP is raw data, therefore preprocessing data is an important step. The main problem that we need to deal with is the inconsistent of the unicode characters in the data, specifically, some characters have the same appearance but different unicode. Figure 1 describes the differences in details.

Native	Symbola [1]	Code
t	t	0074
r	r	0072
ắ	ắ	1EAF
n	n	006E
g	g	0067

Native	Symbola [1]	Code
t	t	0074
r	r	0072
ă	ă	0103
´	´	0301
n	n	006E
g	g	0067

Figure 1: Example of words that have characters with same appearance but different code

These differences can be very noisy in extracting n-grams and TF-IDF features. Therefore we convert these characters to a same format. Word-tokenizing is not used in our system, we only use the syllables.

Our approach is pretty straight forward. We formalize the task as a multi-labeling task, we use a multi-label MLP for our system. There are 34 and 12 categories in the hotel and restaurant datasets, respectively, each with 3 opinions, so

in total we have 102 classes for the hotel dataset and 36 classes for the restaurant dataset. Our model is trained with n-grams and TF-IDF features in which n = 1, 2, 3, 4. The scikit-learn library (Pedregosa et al., 2011) is used for the implementation.

3 Evaluation results

We evaluate our system with the data provided by VLSP, we report recall, precision and F1 score for aspect and both aspect-opinion. The results is presented in Table 1.

		Hotel dataset	Restaurant dataset
Aspect	Recall	0.42	0.38
	Precision	0.85	0.88
	F1-score	0.56	0.54
Aspect-Opinion	Recall	0.39	0.35
	Precision	0.8	0.79
	F1-score	0.53	0.48

Table 1: Evaluation results

4 Error analysis

A problem with our system is that it does not handle the opposition between opinions. For example, our system might give both "positive" and "negative" for a same aspect at the same time. This is possibly caused when a review has positive and negative factors for a same aspect. For example: "Nhin chung thi phong sach se, nhung tu quan ao lai co bui.", this review shows both positive and negative side of the aspect ROOMS#CLEANLINESS. A temporary solution that we use is convert a "positive-negative pair" to "neutral". "Neutral" reviews are also harder for our system to recognize. We hypothesize that it is because "neutral" reviews contain both positive and negative opinions and "neutral" reviews are cases where even annotators find they are difficult to decide whether it is positive or negative.

Dealing with slang, teen codes, emojis is a pain in analyzing reviews on social media, these parts of a review may give noises to the model, but some of them can be very useful. For instance, a heart emojis or a smiley face may be used to express satisfaction, while a thumb down might express the opposite meaning. We will handle this problem in future work. Some examples of cases that mentioned above are presented in Figure!2.

Review that gives both positive and negative opinions for the same aspect	Nhan vien phuc vu nhiet tinh. That vong
Reviews with emojis and "teencodes"	😍 rẻ giã mannn cả chỗ này chưa đến 100k
	Cay quá không nuốt nổi 😂 😂

Figure 2: Some reviews that our system missclassified

Conclusion

In this report, we present our participant system for the SA task at VLSP 2018 evaluation campaign. We adopt a simple multilabel Multilayer Perceptron trained with n-grams and TF-IDF features. The method is straight forward and easy to implement with the scikit-learn library. The system obtained 0.54 F1-score on hotel dataset and 0.48 F1-score on restaurant dataset. We also analyse errors made by our system and discuss some difficulties of the task.

References

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics.

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. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

VLSP 2018 Shared Task: Aspect Text-To-Speech Evaluation

Viet Son Nguyen, Nhut Pham Minh,
Chi Mai Luong, Quan Vu Hai, Thi
Minh Huyen Nguyen

***Abstract*—This paper summarizes the text-to-speech evaluation campaign of the workshop Vietnamese language and speech processing VLSP2018.**

Keywords—Vietnamese speech, text to speech, speech synthesis

I. INTRODUCTION

The evaluation campaign in the 5th international workshop on Vietnamese Language and Speech Processing VLSP 2018 deals with two tasks for speech processing: Automatic speech Recognition (ASR) and Text-To-Speech (TTS). The Vietnamese speech synthesis could be one or more regions (Northern, Central and Southern of Vietnam).

II. PREPARATION

A. TTS test set preparation

In order to evaluate the quality of TTS systems, the test set contains 30 numbered sentences in the news domain. These sentences have different length, and contain some information on date, personal name, foreign location name, and some Vietnamese popular abbreviations, etc. The test set are collected from titles of e-newspapers such as dantri.com.vn, vnexpress.net, vietnamnet.vn, etc. on March 7th, 2018 (see Table I). The TTS test set is released on March 8th, 2018 at 8:30 am. Each registered team can choose to generate speech in one or more dialects of the three regions: Northern, Southern and Central.

TABLE I. THE TEST SET FOR TTS EVALUATION

No. ID	Test set
TTS1	Sẽ không còn đường lùi về thời gian cho dự án đường sắt Cát Linh - Hà Đông.
TTS2	Độc đáo nghề dệt thổ cẩm của người Xtiêng ở Bình Phước (Tin chiều 5-3-2018).
TTS3	Đắk Lắk: Nóng chuyện phá rừng, trách nhiệm thuộc về ai? (Thời sự tối ngày 05 tháng 03 năm 2018).
TTS4	Khu Công nghệ cao Hòa Lạc đẩy nhanh thu hút các dự án có hàm lượng KHCN cao (Bản tin sáng ngày 8.3).
TTS5	Đi taxi 3km, du khách người Mỹ bị chặt chém gấp 10 lần.
TTS6	Netanyahu và Trump: Cuộc gặp của những người cùng cảnh ngộ.
TTS7	Cần giải quyết những bất cập để thực hiện Luật Bình đẳng giới đối với phụ nữ trong lao động việc làm (Bản tin 19h ngày 07/03/2018).
TTS8	Toà án nhân dân tỉnh Phú Thọ đẩy mạnh cải cách tư pháp nâng cao chất lượng xét xử (Bản tin sáng lúc 7h45, ngày 6/3/2018).
TTS9	Mời quý vị tiếp tục xem chương trình Bótay.kom chiếu lúc 16:30 hôm nay.
TTS10	Chính phủ đồng hành cùng người dân và doanh nghiệp phát sóng trong chương trình Thời sự lúc 19g30.
TTS11	Thông điệp liên bang của Tổng thống Nga Vladimir Putin.
TTS12	Cuộc chiến thương mại đe dọa quan hệ giữa Mỹ với các đối tác.
TTS13	Thành phố HCM: Cuộc chiến lập lại trật tự vỉa hè cần kiên quyết và bền bỉ.
TTS14	Người dân vây nhà máy thép gây ô nhiễm ở Đà Nẵng vì chậm di dời: Tại sao chính quyền lúng túng xử lý vụ việc? (Chương trình cafe sáng ngày 7/03/2018)
TTS15	Có nên tiếp tục triển khai tuyến buýt nhanh BRT ở Hà Nội hay không?
TTS16	Ngành nông nghiệp: Bước khởi động tốt từ kết quả 2 tháng đầu năm.
TTS17	Việt Nam được gì khi tham gia Hiệp định đối tác toàn diện và tiến bộ xuyên Thái Bình Dương?
TTS18	Việt Nam được gì khi tham gia Hiệp định đối tác toàn diện và tiến bộ xuyên Thái Bình Dương?
TTS19	Gia Lai: Hệ lụy của tình trạng cho thuê đất sản xuất trong vùng đồng bào dân tộc thiểu số.
TTS20	Ghép tạng: Sự chuyển biến đã đến từ cộng đồng?
TTS21	Nguy cơ cuộc chiến thương mại mới khi Mỹ áp đặt thuế lên nhôm và thép nhập khẩu.
TTS22	Về việc xét công nhận đạt tiêu chuẩn chức danh GS, PGS đợt năm 2017, Thường trực Chính phủ yêu cầu Bộ trưởng Bộ GD&ĐT Phùng Xuân Nhạ rút kinh nghiệm một cách nghiêm khắc.
TTS23	Công nghiệp cơ khí: Làm gì để trở thành ngành công nghiệp then chốt tại Việt Nam.
TTS24	Việc tăng phí liệu có giúp hạn chế phương tiện cá nhân lưu thông vào khu vực tuyến phố trung tâm như cơ quan chức năng kỳ vọng?
TTS25	Phòng chống tham nhũng - Những tín hiệu tích cực.
TTS26	Thành phố HN thí điểm xây dựng khu nhà ở xã hội cho 11.000 người tại huyện Đông Anh.
TTS27	Campuchia thúc đẩy phát triển du lịch bền vững.
TTS28	Chủ tịch Liên đoàn bóng đá Việt Nam nhiệm kỳ 8: Người sẽ đưa bóng đá Việt Nam đi lên? (Chương trình Nhịp đập 360 độ thể thao, kênh VTV3, ngày 7/3/2018).
TTS29	Tin tức về chủ đề Thần tượng Bolero: Ca sĩ Như Quỳnh quay trở lại showbiz Việt sau 24 năm vắng bóng.
TTS30	Thành lập trường THPT trong trường ĐH Lâm Nghiệp Việt Nam.

B. Result and technical report submission

After receiving the test set, teams are not allowed to modify or change any information in the test set (word, date, personal/foreign location name, abbreviation, etc.). The synthesized speech for each sentence must be generated automatically, it should be saved as a WAV format (16kHz sampling frequency and 16 bits). All 30 synthesized speech files must be saved under the name format <team's name><sentence's number>.wav, then they must be compressed in a zip file named <team's name>.zip. Each team

has only one key given by the committee to upload one or many results by using the webservice[1] (see Figure 1). Each team can submit more than one result, but the later submission will be considered as the official result. All results must be submitted before 6:00 pm on March 9th, 2018. After this time, any submission will be refused. After submitting the result, a technical report of each participant's system should be submitted before 6:00 pm on March 15th, 2018 by using the EasyChair system.

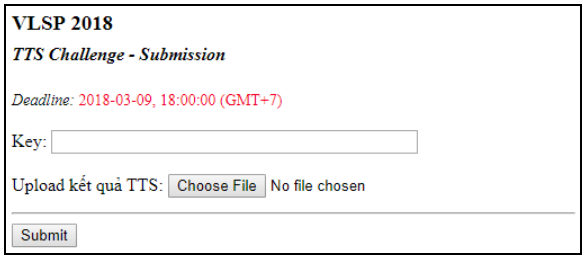

Fig. 1. The interface of the web service for TTS challenge - Submission

III. EVALUATION

The organization have received totally six registrations but at the end of submission results, there are only three of them who participated and submitted the results: TTS system of institute MICA (HUST), TTS system of VAIS and TTS system of Viettel Cyber Space Center. Twenty subjects are invited to evaluate all the synthesized speech results of the teams (30 files x 3 teams) by using the web service[2]. Figure 2 shows the interface of the web service for TTS evaluation. Before evaluating the results, each subject must use private key given by the organization to fill some personal information such as full name, sex, dialect, phonetician or non-phonetician.

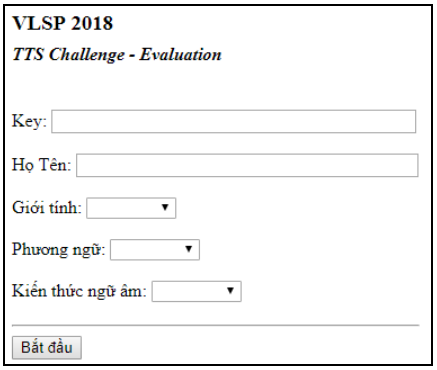

Fig. 2. The interface of the web service for TTS challenge - Evaluation: Fill some personal information of subject

Table II presents the subject information analysis. There are 20 subjects (12 females, 8 males) who participate in the evaluation. Although, all teams generate their speech in Northern dialect, we would like to invite some subjects who come from the Central (07 people), and the South (03 people) of Vietnam. For the subject's experience, the subjects are not only ordinary people but also phoneticians.

TABLE II. SUBJECT INFORMATION ANALYSIS

Subject information						
Sex		Dialect			Phonetician	
F	M	North	Central	South	YES	NO
12	08	10	07	03	13	07

After filling all the personal information, each subject will start an evaluation test. Each subject sees the text of each sentence and listens the corresponding synthesized speech. The subject will give his/her score for three criteria: Naturalness, Intelligibility and MOS. The subjects can listen again the sound if they want. Each subject carries out her/his evaluation one time. Each evaluation test spends about 20 minutes.

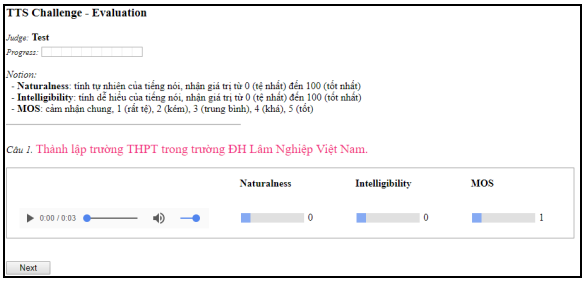

Fig. 3. The interface of the web service for TTS challenge - Evaluation with the three tests: Naturalness test, Intelligibility test and MOS test.

Table III presents the average results of the Naturalness test, Intelligibility test and MOS test. These results are calculated for each team and for twenty subjects. It can be noted that in all tests, the result of Viettel team are always higher than the others. The result of MICA team are better than the VAIS's one.

TABLE III. THE AVERAGE RESULTS OF NATURALNESS TEST, INTELLIGIBILITY TEST, AND MOS TEST

Test	Average results of all subjects		
	VAIS	Viettel	MICA
Naturalness	65.50	**90.54**	72.69
Intelligibility	72.59	**93.02**	78.94
MOS	3.48	**4.66**	3.79

TABLE IV. THE AVERAGE RESULTS OF NATURALNESS TEST, INTELLIGIBILITY TEST, AND MOS TEST FOR MALE SUBJECTS

Test	Average results of male subjects		
	VAIS	Viettel	MICA
Naturalness	65.96	**91.58**	76.06
Intelligibility	72.51	**94.18**	82.10
MOS	3.46	**4.73**	3.90

Table IV and Table V show the average results of the tests for male and female subjects. The results show that for all tests, male subjects often give a higher score than female, but the

[1] https://ailab.hcmus.edu.vn/tools/ttsSubmission/ttsSubmission.html

[2] https://ailab.hcmus.edu.vn/tools/ttsEvaluation/ttsEvaluation.html

difference is not considerable. The results of Viettel team still get the highest score.

TABLE V. THE AVERAGE RESULTS OF NATURALNESS TEST, INTELLIGIBILITY TEST, AND MOS TEST FOR FEMALE SUBJECTS

Test	Average results of female subjects		
	VAIS	Viettel	MICA
Naturalness	65.19	**89.86**	70.24
Intelligibility	72.64	**92.24**	76.84
MOS	3.49	**4.61**	3.71

Table VI and Table VII present the average results of the tests for phoneticians and non-phoneticians. There are also small different between the given score between two groups. The phoneticians give higher score than the non-phoneticians in the case of Viettel and MICA systems, but the results of VAIS systems do not satisfy the phoneticians (it ~~get~~gets higher score from the non-phoneticians). With the both kind of subjects (phonetician and non-phonetician), Viettel team keeps the best results for all the tests.

TABLE VI. THE AVERAGE RESULTS OF NATURALNESS TEST, INTELLIGIBILITY TEST, AND MOS TEST FOR PHONETICIAN SUBJECTS

Test	Average results of phonetician subjects		
	VAIS	Viettel	MICA
Naturalness	63.85	**91.56**	76.73
Intelligibility	70.37	**93.36**	83.02
MOS	3.39	**4.66**	3.92

TABLE VII. THE AVERAGE RESULTS OF NATURALNESS TEST, INTELLIGIBILITY TEST, AND MOS TEST FOR NON- PHONETICIAN SUBJECTS

Test	Average results of non-phonetician subjects		
	VAIS	Viettel	MICA
Naturalness	68.55	**88.65**	63.94
Intelligibility	76.70	**92.38**	71.38
MOS	3.63	**4.67**	3.55

TABLE VIII. THE AVERAGE RESULTS OF NATURALNESS TEST, INTELLIGIBILITY TEST, AND MOS TEST FOR NORTH DIALECT

Test	Average results of north dialect		
	VAIS	Viettel	MICA
Naturalness	67.15	**93.99**	71.49
Intelligibility	76.5	**96.16**	80.75
MOS	3.55	**4.79**	3.73

TABLE IX. THE AVERAGE RESULTS OF NATURALNESS TEST, INTELLIGIBILITY TEST, AND MOS TEST FOR CENTRAL DIALECT

Test	Average results of central dialect		
	VAIS	Viettel	MICA
Naturalness	62.67	**82.52**	68.91
Intelligibility	68.76	**86.31**	72.69
MOS	3.26	**4.33**	3.52

TABLE X. THE AVERAGE RESULTS OF NATURALNESS TEST, INTELLIGIBILITY TEST, AND MOS TEST FOR SOUTH DIALECT

Test	Average results of south dialect		
	VAIS	Viettel	MICA
Naturalness	65.59	**93.96**	81.04
Intelligibility	68.58	**95.22**	83.80
MOS	3.61	**4.84**	4.33

Although, all teams generate their synthesized speech in Northern dialect, we want to try to evaluate the three tests with different dialect subjects. The Table VIII, Table IX and Table X present the average results for three main dialects in Vietnam (North, Central and South, respectively). It seems that the Central dialect subjects give a lower score than the North and the South. But in all cases, Viettel system always get the best score, better than the VAIS and MICA system.

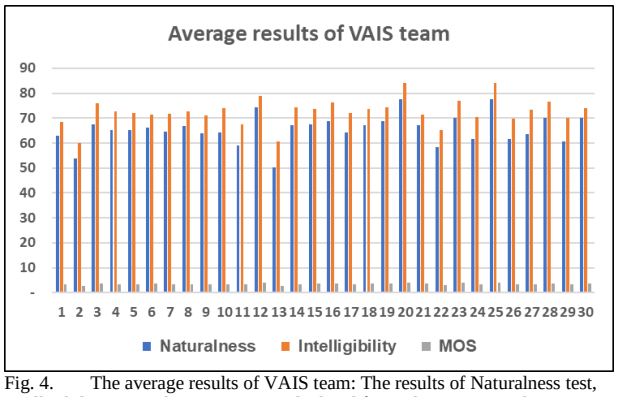

Fig. 4. The average results of VAIS team: The results of Naturalness test, Intelligibility test and MOS test are calculated for each sentence in the test set.

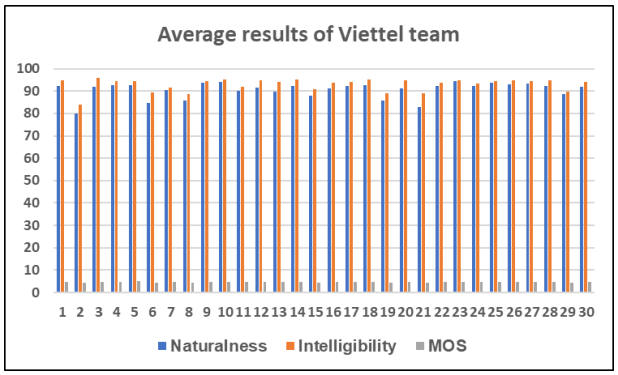

Fig. 5. The average results of Viettel team: The results of Naturalness test, Intelligibility test and MOS test are calculated for each sentence in the test set.

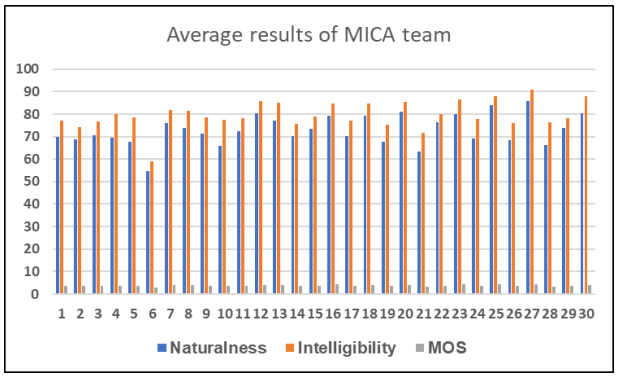

Fig. 6. The average results of MICA team: The results of Naturalness test, Intelligibility test and MOS test are calculated for each sentence in the test set.

In order to evaluate more detailly and independently each team's system, we try calculating the average results of the

tests for each sentence in the test set. Figure 4, Figure 5, and Figure 6 illustrate the average results of VAIS team, Viettel team and MICA team, respectively. They are calculated for the result of each team, and for each sentence in the test set. It can be noted that for all sentences, the result of Viettel system is always higher and more stable than the one of VAIS and MICA systems. It seems that the scores of Naturalness test and Intelligibility test of Viettel system are the same, but for VAIS and MICA systems, the scores between Naturalness test and Intelligibility test are different.

IV. CONCLUSIONS

Paper presents all tasks of the TTS Evaluation Campaign of VLSP 2018. After the preparation process, the TTS Evaluation Campaign is carried out with three teams MICA (HUST), VAIS and Viettel. The evaluation process is done with twenty subjects including male, female, different dialect in Vietnam, phonetician and non-phonetician. The different analysis results show that the TTS system of Viettel team always gets the best score of subjects, the TTS system of VAIS team gets the worse score. By analyzing the score of each sentence, it can be noted that the quality of the Viettel system are more stable and more robustness.

ACKNOWLEDGMENT

The organization would like to thank to laboratory AILab of Vietnam National University Ho Chi Minh city who helps us to prepare the web services for our campaign. We would like to thank all subjects who spend their precious time to carry out all the tests.

Report on the Vietnamese ASR task

Nguyen Van Huy
Thai Nguyen University of Technology

***Abstract*— This paper describes the ASR task for VLSP-2018, and an overview of participated systems such as feature extraction, acoustic model, language mode, and decoding process.**

Keywords—VLSP 2018, ASR, VAIS, Viettel.

I. DEFINITION

In the ASR task, participants were asked to transcribe automatically Vietnamese audio files into the spoken word sequences. The committee provided the test set only, while the training data for the acoustic and language models was developed by the teams themselves. The test set was delivered on March 8, 2018 via email, and then each team had two days to recognize it. The final result had to submit by 16-PM on March 10, 2018. The Word Error Rate (WER) was measured with references which were human transcripts of the audio files.

II. TEST DATA

The test set was composed of 796 continuous wav files of news speech for a total duration of two hours, without any information on the sentence segmentation. The speech was recorded in a non-noisy environment, and available in three dialects: Northern, Southern and Central with respectively proportion of 50%, 40% and 10%.

III. SUBMISSION

We received two submissions which were from VAIS and Viettel-CSC.

IV. OVERVIEW OF PARTICIPATED ASR SYSTEMS

	VAIS	**Viettel-CSC**
Resources		
Acoustic Data	1200h, 3 dialects	500h, mostly is Northern
Text data	...	900MB, online newspapers
Words in Lexicon	6,5k	11k (6k Vietnamese + 5k foreign)
System		
Feature	MFCC + Pitch	MFCC + Pitch + Bottleneck
Acoustic model (AM)	7 layers DNN	Combination DNN model including two sub-models (TDD and BLSTM)
Language model (LM)	N-gram	N-gram, RNN-LM
Decoding process	- 1st decoding: AM + 4-gram LM - Re-scoring: 5-gram	- 1st decoding: TDNN + 4-gram, BLSTM + 4 gram - Rescoring: RNN-LM - Combining: Using two 1st-decoding outputs - LM Adaptation: Topic adaptation - 2nd decoding: Re-decoding with Adapted-LM

V. EVALUATION RESULT

Scores in WER and Sentence Error Rate (SER) for the submissions of VAIS and Viettel-CSC are shown in Table 1.

TABLE I. ASR EVALUATION RESULTS FOR VLSP-2018

Team	**WER**	**SER**
VAIS	**6.29%**	75.50%
Viettel-CSC	7.40%	75.38%

VI. CONCLUSIONS

This year is the first time VLSP starts the ASR task. There were six registrations, but only two submissions. The best team in WER is VAIS, but in SER is Viettel-CSC.

VAIS-Speech: An Overview of Automatic Speech Recognition and Text-to-speech Development at VAIS

Quoc Truong Do
Vietnam Artificial Intelligent System
Email: support@vais.vn

***Abstract*—In this paper, we describe the development of automatic speech recognition (ASR) and text-to-speech (TTS) systems at VAIS. Our speech engines utilized the state-of-the-art technologies that have been used in many popular languages such as English and Japanese. Moreover, we also designed many features of the core engine that are highly optimized for Vietnamese. We evaluated our ASR engine on large test sets including news (9.3 hours) and mobile phone (10 hours) domains spoken by both Northern and Southern accents. As the result, we achieved 5.58% WER on the news test set and 7.99% on the mobile phone set.**

I. Introduction

Automatic speech recognition (ASR) and speech synthesis technologies are two important components for a speech based human-machine interaction. Both ASR and TTS are active research areas that has been well studied for few decades. And thank to the powerful of deep learning, recently, these technologies are deployed in many softwares and companies making the human-machine communication at remarkable high accuraries. In this paper, we present our deep learning based ASR and TTS engines for Vietnamese language. Our ASR engines can recognize words spoken by both north, south and center regions of Vietnam and the TTS engines support both north and south accents.

II. Automatic speech recognition

In a conventional Gaussian Mixture Model - Hidden Markov Model (GMM-HMM) acoustic model, the state emission log-likelihood of the observation feature vector o_t for certain tied state s_j of HMMs at time t is computed as

$$\log\, p(\mathbf{o}_t|s_j) = \log \sum_{m=1}^{M} \pi_{jm} \mathbb{N}(\mathbf{o}_t|s_j) \tag{1}$$

where M is the number of Gaussian mixtures in the GMM for state j and π_{jm} is the mixing weight. As the outputs from DNNs represent the state posteriors $p(s_j|\mathbf{o}_t)$, a DNN-HMM hybrid system uses pseudo log-likelihood as the state emissions that is computed as

$$\log\, p(\mathbf{o}_t|s_j) = \log\, p(s_j|\mathbf{o}_t) - \log\, p(s_j), \tag{2}$$

where the state priors $\log p(s_j)$ can be estimated using the state alignments on the training speech data. In our experiment, we used an deep neural network (DNN) with 7 layers to generate the state posteriors $p(s_j|\mathbf{o}_t)$. Our language model use 4-gram model and it is trained from news and conversation dataset.

III. Text-to-speech

A. Speech concatenation approach

Speech concatenation [1] is an approach that synthesize audios by concatenate speech segments from a database. The idea is as follows, first, audio and text are aligned at the phoneme level to provide information about where the phoneme is located in the speech sound. Second, the text is also analyze to provide linguistic information, such as phonetic context, part-of-speech tags, word position. Finally, a database is built which contains all speech segments and phonetic information.

At the synthesis time, the speech segment is chosen to minimize the combination of the target cost which measured by a heuristic distance between contexts and the concatenation cost which measured by speech parameter distortion (Fig. 1).

Because the audio is synthesized using the speech segment extracted directly from original audios, it provides very high quality speech waveform signal. If the model can choose correctly speech segments, it is very difficult for human to distinguish the synthetic voice and the natural voice. However, allthough the concatenation approach can produce high quality speech waveform, it comes with some disadvantages. First, it has large footprints because all speech segments are stored in the model. Our actual model size is approximately 1GB when trained on 6k utterances. Second, the sound sometime has unstable quality due to wrong alignment and mistake during segment selection.

B. HMMs approach

The HMM approach models speech by using Gaussian mixture models. Each phoneme is modeled by an HMM instead of many speech segments in concatenation approach. At the synthesis time, a sentence HMM is constructed from the given text. Then the speech parameter is predicted to maximize the likelihood probability,

$$\hat{\boldsymbol{O}} = \underset{\boldsymbol{O}}{\operatorname{argmax}}\, P(\boldsymbol{O}|\lambda, T), \tag{3}$$

where $\boldsymbol{O}$ is speech parameters, λ is the model parameters, and T is the length of speech that we want to generated.

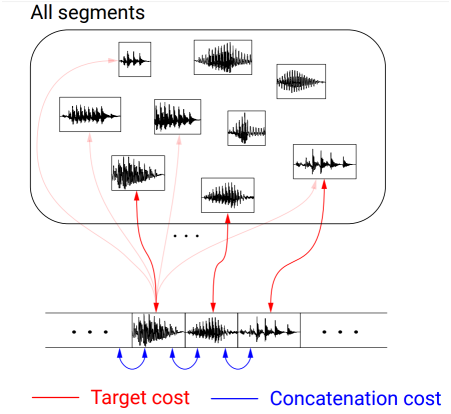

Figure 1. Speech concatenation procedure.

Unlike unit concatenation approach where we need to collect large amount of speech data to have good quality, the HMM approach can trained a model with just few hundreds phonetic-balanced utterances. This allows us to quickly train a fairly good model given small amount of data.

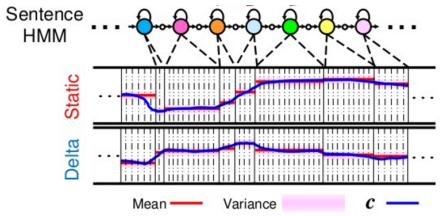

Figure 2. HMM-based speech synthesis

The HMM voice has very low footprints, in fact, our model trained on 6k utterances takes only 5MB of storage and 10MB of memory when fully loaded. The approach can also provide very flexible voices by changing model parameters [2], and also be able to adapted to someone else voices with minimal data collection required[3].

C. Front-end text processing

Vietnamese is a complex language where one word can be pronounced in different ways depending on the context. Another problem is abbreviations that is very often being used in newspaper, such as TPHCM, VKSND. The list of abbreviation is endless and have no rules to pronounced.

To make the speech synthesis more useful for general tasks, we define a set of regression rules for date, numbers, date-of-birth, time, units (such as currencies, temperature, weights), and develop a toolkit for define abbreviation words. The processing procedure is illustrated in Fig. 3.

D. Available voices

At the current stage, we provide 2 speech concatenation female voices with Northern and Southern accent and 1 HSMM voice. All voices are made available for online access

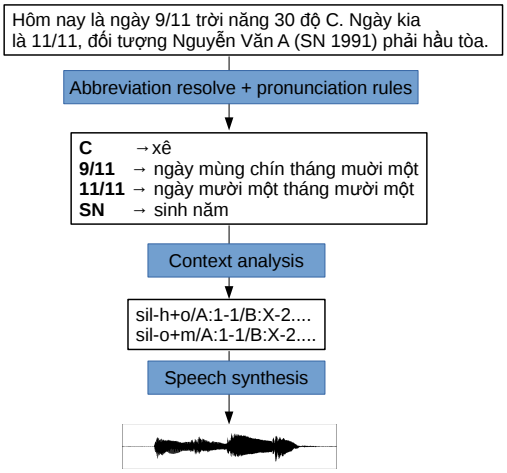

Figure 3. The text analysis front-end procedure.

Table I
Word Error Rate (WER) Evaluation on news and mobile test set

Model	News	Mobile
DNN 7x1024	5.58	7.99

and the HSMM voice is also available for offline access including PC and mobile platforms.

The dataset used for training models is described as follows:

- **Southern accent:** 5.8k utterances of female voice collected from VOV audio speech. The length of each utterance varies from 5 to 15 words.
- **Northern accent:** 6k utterances of female voice. The length of each utterance varies from 14 to 17 words.

IV. Experimental and Results

A. ASR evaluation

In this section, we describe our ASR training setup and evaluation. Our acoustic model is trained using 1200 hours including North, South and Central regions. Audio speech signals are sampled at 16kHz sampling frequency. The acoustic feature includes both Mel Frequency Ceptral Coefficients (MFCC) feature and pitch feature. The MFCC feature is extracted using an windown of 25ms and a shift of 10ms.

In our experiments, a deep neural network (DNN) with 7 layers is used to generate the state posteriors. Each layer has 1024 nodes. To train the DNN, the cross-entropy objective funtion is used. We used the 4-gram model language model in which has vocabulary size of 6500 words. And the output lattices is re-scored with a 5-gram language model.l

The result is shown in Fig. I. As we can see, our sytem perform very well on the news dataset when the audio quality is good and clean. While on the mobile test set where the audio is noisy and include more conversation and fast-speaking style, the performance is dropped by approximately 2%.

B. TTS training setup

Our TTS system is trained on 5,000 utterances spoken by a female speaker with Northern accent using the HTS toolkit. Speech features are 40 dimension mel-generalized cepstral coefficients (MGC), 1 dimension band apperiodic feature and 1 dimension log F0 extracted with a windows of 25 frames and 5 frames shift.

V. Conclusion

In this paper, we briefly describe our ASR and TTS development. Both engines achieved high performance and can be used in general domain. The ASR engine works well not only under clean but also very noisy and far-field environment. More detail and desmonstration are available on our website: https://vais.vn.

References

[1] A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *In Proceeding of ICASSP*, 1996, pp. 373–376.

[2] N. Takashi, J. Yamagishi, T. Masuko, and T. Kobayashi, "A style control technique for HMM-based expressive speech synthesis," *IEICE*, vol. 90, no. 9, pp. 1406–1413, 2007.

[3] J. Yamagishi and T. Kobayashi, "Average-voice-based speech synthesis using HSMM-based speaker adaptation and adaptive training," *IEICE*, vol. 90, no. 2, pp. 533–543, 2007.

MICATTS: Non-uniform unit selection speech synthesis system for Vietnamese

Tien-Thanh NGUYEN
International Research Institute MICA, HUST-CNRS/UMI 2954-Grenoble INP
Hanoi, Vietnam
Email: tien-thanh.nguyen@mica.edu.vn

Dang-Khoa MAC
International Research Institute MICA, HUST-CNRS/UMI 2954-Grenoble INP
Hanoi, Vietnam
Email: dang-khoa.mac@mica.edu.vn

Do-Dat TRAN
Office of National Science and Technology Research Program, MOST
Hanoi, Vietnam
Email: tddat@most.gov.vn

***Abstract*—This paper describes the development of a Vietnamese Text-To-Speech (TTS) system, which uses non-uniform unit selection technique. To deploy a full-functional TTS system on limited resource such as mobile devices, we uses a syllable dictionary to index and link all speech units in the database**

Keywords— text-to-speech, speech synthesis, non-uniform unit selection, unit selection, indexed syllable dictionary

I. INTRODUCTION

Text-to-speech technology has been developed for decades with many different techniques [1]. Recently, speech synthesis systems are developed by two main approaches: unit selection speech synthesis [2] and statistical parametric speech synthesis [3],[4]. In unit selection synthesis, synthesized speech is concatenated by the segments of speech units. These segments were selected from a speech corpus that contain a large amount of recorded speech, along with its appropriate transcription and annotation as shown in Figure 1.

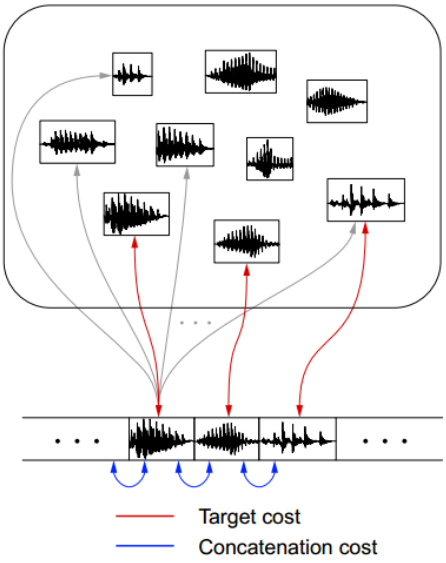

Figure 1. Unit selection scheme [5]

By contrast, statistical parametric approach generates output speech by using parametric models, which were trained from the database [3]. Each of the two technologies above has its own strengths and weaknesses. The strong point of unit selection is the natural-like sounding of output voice. However, its weakness is the decoding process with large run-time data. Statistical parametric approach has some advantages, such as being highly customizable, having a smoother output than unit selection approach, while not needing much data for the decoding process. However, the drawback is the robot-like output speech [5].

This TTS system use non-uniform unit selection approach because of the easy implementation and the ability to produce good synthesized speech results

II. TECHNOLOGIES

A. Overview

MICATTS is an unit selection based TTS system for Vietnamese. This system employed a large database which contains many segments of recorded speech with the transcription. The synthesis process is described in Figure 2.

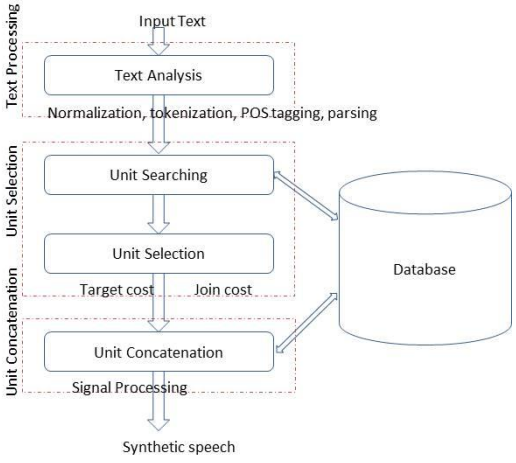

Figure 2. Non-uniform units selection model [9]

The first step performs text processing tasks such as text normalization, word segmentation, part-of-speech tagging and syntax parsing. In the second step, based on the result of text processing, the unit searching process will be carried out to choose potential candidates in the speech database, and the most suitable units will be selected among these best candidates. Finally, the final process will concatenate all these units to generate the output speech signal.

B. Text processing

Before converted to the speech, all the text must be processed, in order to :

- Normalize all special text, such as: number, date, letter, email, URL, etc. into readable text.
- Detect the unknown words (abbreviation, loan word, private name, etc.). and convert to readable text.

1) Text normalization process

The input sentence will be processed word by word. Firstly, all the words in three categories above (Number, Letter, URL etc.) will be extracted and normalized using defined regular expression rules. The rest words of input sentence will be converted to readable text using 3 pronunciation dictionaries. . Each dictionary contain word and its corresponding pronunciation. For example: word "WTO" and the corresponding pronunciation "vê kép tê ô". There are 3 type of dictionaries:

- Common dictionary: contains common readable word list
- Packages Dictionary: contains specially readable word list for each voice package
- User dictionary: the dictionary defined by the user

After "checking" in 3 dictionary, if there is still unreadable word (unknown word), the TTS system will spell this unknown word by character. For example, the word "abc" which does not exist in 3 dictionaries will be read as "a bờ cờ".

2) Unknown word management

One of the biggest issues of Text to speech is the unknown words (i.e loan word, acronym, private name etc.). One of our solution is to provide an User dictionary for this case (the third dictionary mentioned above). With this dictionary, user can define their own pronunciation for their word.

Whenever TTS system reads out the input text, it priority checks User Dictionary first to read the words the way that user wants.

C. Data organization

The proposed speech database structure consists of two components:

- Audio files: stored in *wav* format, recorded by two professional native speakers, one male and one female. In our TTS system, the total number of files is 3085 (corresponding to 3085 sentences or paragraphs). The total duration of audio signal is approximately 4 hours for each speaker voice.
- A metadata file (in *xml* format) contains all necessary information for all the unit selection process. The necessary information for each syllable (the smallest unit) is stored in the xml tags shown in Table 1.

Table 1: XML Tags and meaning

Tag	Meaning
Name	Name of syllable
Start_index	Start index of syllable
End_index	End index of syllable
Initial	Initial phone of syllable
Middle	Middle phone of syllable
Nucleus	Nucleus phone of syllable
Final	Final phone of syllable
Type	Type of syllable
LeftSyl	Syllable in the left side
RightSyl	Syllable in the right side
Tone	Tone of syllable
finalPhnm	Final phone
initialPhnm	Initial phone

The data structure is also represented in this Metadata file. This structure consists of the following components (see Figure 3):

- List of sentences: Including all sentences in the database. Each sentence has a distinguished ID.
- Syllable dictionary: Including all the distinctive syllables in the database. In our case, the total number of distinctive syllable is 8175. This dictionary is stored as a single list, where each element corresponds to one syllable, and is arranged in the alphabetical order. Each element consists of two components: syllable name and a list of sentence ID of the sentence in database containing this syllable.

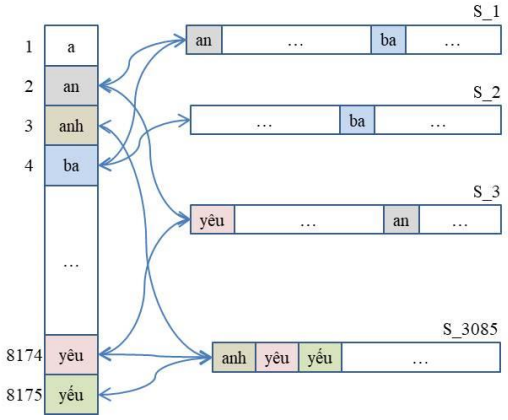

Figure 3. Data structure using syllable dictionary

Based on the proposed data organization, the unit searching process will be done primarily on the dictionary which is considered as an one dimensional array. It reduces the

complexity of the searching process compared to the implementation of syntactic tree as the conventional approachnt designations.

D. Unit Searching

1) Finding the longest matching unit

The idea of this process is to find the longest phrase in the input sentence, which exists in the database.

Algorithm:

- Step 1: Decomposing the input sentence into individual syllables:

S = {"syl(1)", "syl(2)",…,"syl(i)",…, "syl(n)"}
*S is ordered, T = corpus, i = 1, phrase = {syl(1)}

- Step 2: Checking value of i. If i<n, continue to step 3. Else, going to Step 5
- Step 3: Exhaustively searching in T to get a list of sentences T' containing phrase. T := T'.
- Step 4: Checking whether T empty or not. If it is not, concatenating current "phrase" with the next syllable in S, phrase = {"phrase" + "syl(i+1)"}. i++. Going back to step 2.

If T is empty, save previous phrase. New phrase = syl(i). Going back to step 2

- Step 5: Finishing search

An example for finding the longest matching unit with the input sentence "*Trường đại học bách khoa hà nội*" (The university of science and technology), is described in Figure 4:

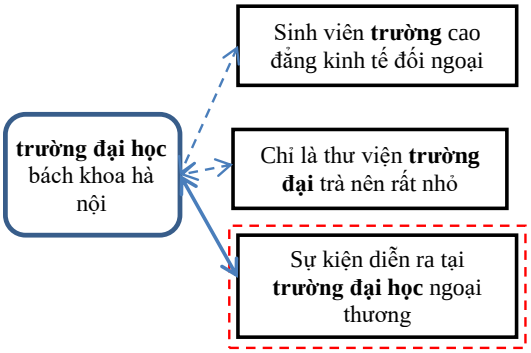

Figure 4. Finding the longest phrase in database

Step 1: Decomposing the input sentence into individual syllables.

1.1. Starting the searching process with the first syllable "trường".

1.2. Getting a set of sentences containing syllable "trường" (D1).

Step 2: Checking whether the syllable exists in D1. If it does, then concatenating it with the next syllable into the phrase "*trường đại*".

Step 3: Searching for new segment in the database D1, if one exists, continue to concatenative one more next syllable, get new phrase "*trường đại học*".

Step 4: Until there is no occurrence of the phrase in the database D1 (The "*trường đại học bách*" cluster is not included in database), save phrase "*trường đại học*". Back to step 1 with the new syllable "bách".

After finishing this process, the input sentence will be split into consecutive smaller unit segments. For each segment, a list of candidates was found in the database. These candidates will be used in the next step of "best candidate selection".

2) The best candidate selection

The best candidate is selected based on two costs:

- Target cost $C^t(t_i, u_i)$: a cost or distance between the target s_t and a candidate unit in the database u_t. In this system, the target cost is calculated based on the differences in context such as: The difference in position of units, preceding syllable and following syllable of units, preceding phoneme and following phoneme of units, preceding phoneme type and following phoneme type of units, preceding tone and following tone of units.

- Concatenation cost $C^c(u_{i-1}, u_i)$: a measure of how well two units join together (low values mean good joins). In this work, the concatenation cost comprises differences between the right and left segment of unit

The total combined cost for a sequence of n units is given by: [2]

$$C(t_1^n, u_1^n) = \sum_{i=1}^{n} C^t(t_i, u_i) + \sum_{i=2}^{n} C^c(u_{i-1}, u_i) + C^c(S, u_1) + C^c(u_n, S)$$

Where S denotes silence, $C^c(S, u_1)$ and $C^c(u_n, S)$ are costs given by the concatenation of the first and last units to silence. Result of selection is the set of units $\bar{u}_1^n$ which minimizes this cost: [3]

$$\bar{u}_1^n = \min_{u_1, \ldots u_n} C(t_1^n, u_1^n)$$

III. PERFORMANCE EVALUATION

For this purpose, the VNTTS2 system with the proposed method is deployed on two platforms:

- PC: (Intel® Core ™ i5-2400 Processor, 6M Cache, 3.40 GHz, DDR3 1333 RAM).
- Android devices (Galaxy J7 2015, S4 mini, Motorola X gen2).

The test dataset contains 100 input sentences with length from 1 to 30 syllables, divided into 3 groups by sentence length:

- Short sentences: 1-10 syllables
- Average sentences: 11-20 syllables
- Long sentences: 21-30 syllables

These sentences were randomly chosen from popular sources such as news, books and novels. These sentences were also carefully chosen to avoid similarity to the sentences in the speech database of TTS system. They will be put into the system to generate the corresponding synthesized speech. The test was performed by a specific application, which measured the responding time of the system from the time of receiving the input text to the starting time of speech output playing.

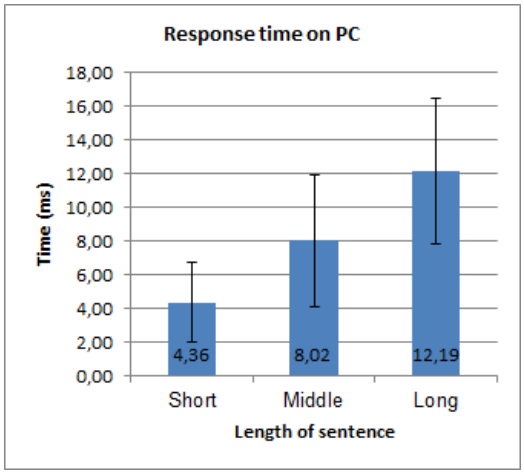

Figure 5. Average response time (with standard deviation) of proposed system on personal computer

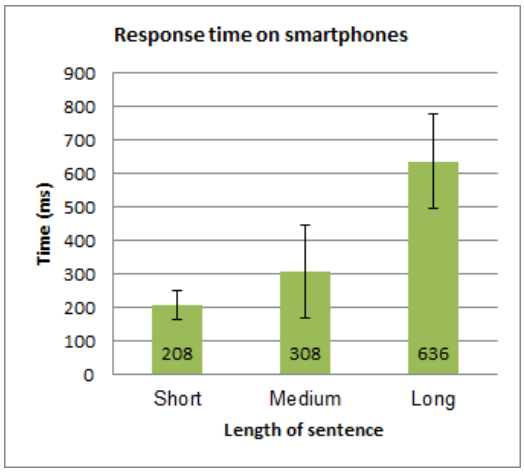

Figure 6. Average response time (with standard deviation) of proposed system on smartphones

Figure 5 shows the response time in average (in milliseconds) for the testing sentences with the different lengths on PC platform (Short: 1-10 syllables, Medium: 11-20 syllables, Long: 20-30 syllables). It can be clearly seen that the new system responds quite quickly (about a few tens of milliseconds). It can also be notable that the result varies depending on the length of the test sentences. This is because with the longer sentences, to find the best unit which fits the given context well, the TTS system must search in a larger searching space. The searching time is also much longer when the number of syllables increases.

Figure 6 shows the response time in average (in milliseconds) of the testing sentences on Android devices. It can be seen that this response time is acceptable for ordinary applications with most cases having a synthesis time less than one second.

IV. CONCLUSION

In this paper, we describe our speech synthesis system for Vietnamese language for Text to Speech task of VLSP 2018 evaluation campaign. Our system uses the approach of Non-uniform unit selection and this approach yields promisingly a good result with enough large database.

In the future, we are expecting to have more voices for more regions in Vietnam. We are also developing new speech synthesis system using other approaches (statistical parametric)

REFERENCES

[1] P. Taylor, *Text-to-Speech Synthesis*, 1 edition. Cambridge, UK ; New York: Cambridge University Press, 2009.

[2] A. J. Hunt and A. W. Black, "Unit selection in a concatenative speech synthesis system using a large speech database," in *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, 1996, vol. 1, pp. 373–376.

[3] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," in *Sixth European Conference on Speech Communication and Technology*, 1999.

[4] H. Zen, T. Toda, M. Nakamura, and K. Tokuda, "Details of the Nitech HMM-based speech synthesis system for the Blizzard Challenge 2005," *IEICE transactions on information and systems*, vol. 90, no. 1, pp. 325–333, 2007.

[5] K. Tokuda, H. Zen, and A. W. Black, "An HMM-Based Speech Synthesis System Applied To English," *IEEE Speech Synthesis Workshop*, pp. 227–230, 2002.

[6] D.-D. Tran, "Synthèse de la parole à partir du texte en langue vietnamienne," Grenoble, INPG, 2007.

[7] H.-Q. Vũ and X.-N. Cao, *Tổng hợp tiếng nói tiếng Việt, theo phương pháp ghép nối cụm từ*. Tập, 2009.

[8] T. T. Vu, M. C. Luong, and S. Nakamura, "An HMM-based Vietnamese speech synthesis system," in *2009*

Oriental COCOSDA International Conference on Speech Database and Assessments, 2009, pp. 116–121.

[9] T. Van Do, D.-D. Tran, and T.-T. T. Nguyen, "Non-uniform Unit Selection in Vietnamese Speech Synthesis," in *Proceedings of the Second Symposium on Information and Communication Technology*, New York, NY, USA, 2011, pp. 165–171.

Development of a Vietnamese Speech Synthesis System for VLSP 2018

Van Thinh Nguyen, Khac Tan Pham, Huy Kinh Phan and Quoc Bao Nguyen
Viettel CyberSpace Center

***Abstract*—This paper describes our deep neural network-based speech synthesis system for high quality of Vietnamese speech synthesis task. The system takes text as input, extract linguistic features and employs neural network to predict acoustic features, which are then passed to a vocoder to produce the speech wave form.**

***Index Terms*—Speech Synthesis, Deepneural Network, Vocoder, Text Normalization.**

I. Introduction

The fifth International Workshop on Vietnamese Language and Speech Processing (VLSP 2018) organizes the shared task of Named Entity Recognition, Sentiment Analysis, Speech Recognition and Speech Synthesis at the first time for Vietnamese language processing. The goal of this workshop series is to attempt a synthesis of research in Vietnamese language and speech processing and to bring together researchers and professionals working in this domain.

In this paper we describe the speech synthesis system which we participated in the TTS track of the 2018 VLSP evaluation campaign.

II. System Architecture and Implementation

A. System Architecture

With the aim of improving the naturalness and intelligibility of speech synthesis system, we propose apply new technologies of speech synthesis for acoustic modeling and waveform generation such as using deep neural network combined with new type of vocoder. To archive to this goal, a new architecture of speech synthesis system is proposed in figure 1. Our proposed system takes text as an input and normalize it into standard text which is readable. Linguistic feature extraction is then applied to extract text's linguistic features as an input for acoustic model. For vocoder parameter generation, Acoustic model used to take given input linguistic feature and generate predicted vocoder parameter. The speech waveform is generated by vocoder, which is new type.

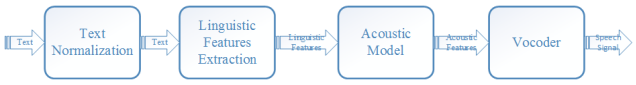

Fig. 1. The proposed speech synthesis system [1]

B. Front End

1) Text Normalization: Text Normalization plays an important role in a Text-To-Speech (TTS) system. It is a process to decide how to read Non Standard Words (NSWs) which can't be spoken by applying letter-to-sound rules such as CSGT (cnh st giao thng), keangnam (cang nam). The process decides the quality of a TTS system. The module implemented and based on using regular expression and using abbreviation dictionary . Regular expression is a direct and powerfull technique to clasify NSWs. We build expressions that describe the date, time, scrore, currency and mesuarment. An abbreviation dictionary containing foregin proper names, acronyms...[2]

2) Linguistics Features Extraction: Linguitics Feature, which was used as input features for the system, had been extracted by generating a label file from linguistics properties of the text (Part-of-Speech tag, word segmentation, and text chunking) and mapping the corresponding information to binary codes presented in a question file. Each piece of information was encoded into an one-hot vector, which was later concatenated horizontally to form a single one-hot vector presenting the text.

C. Acoustic Modeling

Acoustic model is based on deep neural network, specially it is feedforward neural network with enough layers, a simplest type of network. The architecture of network is shown in figure 2. follow this network, The input linguistic features used to predict the output parameter via several layers of hidden units [1]. Each node of the network is called perceptron and each perceptron perform a nonlinear function, as follow:

$$h_t = H(W^{xh}x_t + b^h)$$

$$y_t = W^{hy}h_t + b^y$$

Where H(.) is a nonlinear activation function in a perceptron (in this system, we use TANH function for each unit) [3], W^{xh} and W^{hy} are the weight matrices, b^x and b^y are bias vector.

D. Vocoder

Currently, The speech synthesis system use many type of vocoder and most of vocoder are based on source filter model [4]. In our system we used a vocoder-based speech synthesis system, named WORLD, which was developed in an effort to improve sound quality of real-time application using speech [5]. WORLD vocoder consist of three algorithm for obtaining three speech parameters, which are F0 contour estimated with

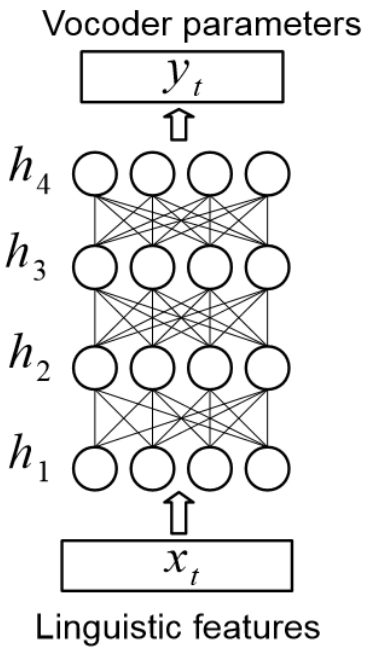

Fig. 2. The feedforward neural network for acoustic modeling

DIO [6], spectral envelop is estimated with CheapTrick [7] and excitation signal is estimated with PLATINUM used as an aperiodic parameter [8], and a synthesis algorithms for obtaining three parameter as an input. With WORLD vocoder, speech parameter predicted from acoustic model which correspond to input text sentence, will be used for produce speech waveform.

III. Experimental Setup and Results

A. Data Preparation

In this section, we describe our effort to collect more than 6.5 hour of high quality of audio for speech corpus which are used to train our acoustic model for speech synthesis system. To archive our target, firstly we are collected 6.5 hour of recordings, but almost our data come from internet such as radio online, because we do not have resource to record audio ourself. Audio data crawled from internet which has much more noise, so the next step we did is apply a noise filter to reduce noise signal. Each audio is very long and the difference in amplitude is very large at different times. For that reason, we cut into small audio file corresponding to text sentence and balanced all these files. And finally, we got a corpus which has more than 3500 audio file corresponding to 6.5 hour of high quality of audio.

B. Experimental Setup

To demonstrate how we archive high quality of speech synthesis, we report experimental setup for this architecture. we used speech corpus collected in previous section.In this data, 3150 utterances were used for training, 175 as a development set, and 175 as the evaluation set.

The input features for neural network, is extracted by frontend, consisted 743 features. 734 of these derived from linguistic context, including phoneme identity, part of speech and positional information within a syllable, word, phrase, etc. The remain 9 features are within phoneme positional information. The speech acoustic features extracted by WORLD vocoder for both training and decoding. Each speech feature vecor contain 60 dimensional Mel Cepstral Coefficients (MFCCs), 5 band aperiodicities (BAPs) and fundamental frequency on log scale (logF0) at 5 milliseconds frame intervals.

Deep neural network is configured with 6 feedforward hidden layers and each layer has 1024 hyerbolic tangent unit.

C. Results

The objective result of the system is presented in table 1. it shown that, MCD: Mel cepstral distortion [9], BAP: distortion of band aperiodicities and V/UV: voice/unvoice error are quite low, that mean we has traned good acoustic model which return the best result. F0 RMSE is caculated on linear scale.

TABLE I
THE OBJECTIVE RESULT OF SPEECH SYNTHESIS SYSTEM

	MCD (dB)	BAP (dB)	F0 RMSE (Hz)	V/UV
DNN system	6	7	22.9	6.15

The subjective results presented in table 2. this table show the comparision of evaluation of deep neural network speech synthesis system with old system based on Hidden markov model. the evaluation of both system is executed by 5 native Vietnamese listener, who evaluated the naturalness and intelligibility of each system on a scale of five. the results shown that, our speech synthesis system based on Deep neural network has better score than old system based on hidden markov model.

TABLE II
THE COMPARISION OF SUBJECTIVE RESULTS

	DNN system	HMM system)
Average score	4.21	3.8

IV. Conclusion

In this paper, our speech synthesis system based deep neural network is shown, and the improvement of this system compared to old system based on hidden markov model which has dominated acoustic modeling for past decade. We hope this system can provide the best speech sysnthesis system for Vietnamse to produce high quality of voice from text. In future work, we want to improve the performace of our system (it still has long time delay for generate an audio from text) by apply parallel computing and the quality by improve quality of data or change neural network architecture.

V. Acknowledgment

This work was supported by Viettel Cybersace Center - Viettel Group.

REFERENCES

[1] Z. Wu, O. Watts, and S. King, "Merlin: An open source neural network speech synthesis system," *Proc. SSW, Sunnyvale, USA*, 2016.

[2] D. A. Tuan, P. T. Lam, and P. D. Hung, "A study of text normalization in vietnamese for text-to-speech system."

[3] J. Jantzen, "Introduction to perceptron networks," *Technical University of Denmark, Lyngby, Denmark, Technical Report*, 1998.

[4] F. Espic, C. Valentini-Botinhao, and S. King, "Direct modelling of magnitude and phase spectra for statistical parametric speech synthesis," *Proc. Interspeech, Stochohlm, Sweden*, 2017.

[5] M. Morise, F. Yokomori, and K. Ozawa, "World: a vocoder-based high-quality speech synthesis system for real-time applications," *IEICE TRANSACTIONS on Information and Systems*, vol. 99, no. 7, pp. 1877–1884, 2016.

[6] M. Morise, H. Kawahara, and H. Katayose, "Fast and reliable f0 estimation method based on the period extraction of vocal fold vibration of singing voice and speech," in *Audio Engineering Society Conference: 35th International Conference: Audio for Games*. Audio Engineering Society, 2009.

[7] M. Morise, "Cheaptrick, a spectral envelope estimator for high-quality speech synthesis," *Speech Communication*, vol. 67, pp. 1–7, 2015.

[8] ——, "Platinum: A method to extract excitation signals for voice synthesis system," *Acoustical Science and Technology*, vol. 33, no. 2, pp. 123–125, 2012.

[9] J. Kominek, T. Schultz, and A. W. Black, "Synthesizer voice quality of new languages calibrated with mean mel cepstral distortion," in *Spoken Languages Technologies for Under-Resourced Languages*, 2008.

VLSP 2018 - Development of a Vietnamese Large Vocabulary Continuous Speech Recognition

Quoc Bao Nguyen, Van Hai Do
Cyberspace Center
Viettel Group
Hanoi, Vietnam
{baonq2, haidv21}@viettel.com.vn

Van Tuan Mai, Quang Trung Le
Cyberspace Center
Viettel Group
Hanoi, Vietnam
{tuanmv2, trunglq12}@viettel.com.vn

Ba Quyen Dam, Manh Dung Do
Cyberspace Center
Viettel Group
Hanoi, Vietnam
{quyendb, dungdm6}@viettel.com.vn

***Abstract*— In this paper, we first present our effort to collect a 500-hour corpus for Vietnamese read speech. After that, various techniques such as data augmentation, RNNLM rescoring, language model adaptation, bottleneck feature, system combination are applied to build the speech recognition system. Our final system achieves a low word error rate at 6.9% on the noisy test set.**

Keywords*— *Vietnamese speech corpus, Vietnamese speech recognition, bottleneck feature, system combination.

I. Introduction

Vietnamese is the sole official and the national language of Vietnam with around 76 million native speakers[1]. It is the first language of the majority of the Vietnamese population, as well as a first or second language for country's ethnic minority groups.

There were several attempts to build Vietnamese large vocabulary continuous speech recognition (LVCSR) system where most of them developed on read speech corpuses [1-4]. In 2013, the National Institute of Standards and Technology, USA (NIST) released the Open Keyword Search Challenge (Open KWS), and Vietnamese was chosen as the "surprise language". The acoustic data are collected from various real noisy scenes and telephony conditions. Many research groups around the world have proposed different approaches to improve performance for both keyword search and speech recognition [5-7]. In 2017, we presented our effort to collect a Vietnamese corpus and build a LVCSR system for Viettel customer service call center [8] and achieved a promising result on this challenging task.

Recently, the Vietnamese Language and Speech Processing (VLSP) community has organized an evaluation campaign for the Vietnamese speech recognition task. The evaluation data were collected mainly from broadcast news such as VOV, VTV. No training or development data was provided. In this paper, we present our effort to collect 500-hour speech corpus and the process to build a Vietnamese LVCSR speech recognition system. Our final system achieves 6.9% word error rate (WER) on our noisy test set.

The rest of the paper is organized as follows. Section II describes our speech corpus. Section III presents the proposed speech recognition system. Section IV shows the experimental results and Section V concludes the paper.

II. Corpus Description

In this paper, we present our effort to collect a 500-hour read speech corpus which will be used to train our speech recognition system.

Previously, several Vietnamese speech corpora were collected by different research groups [1-4]. However, they are relatively small i.e., less than 100 hours while commercial systems normally use thousands of hours of training data. In Viettel, beside building a speech recognition for telephone conversation such as for call center, we also target on building a commercial system for other applications such as virtual assistance, smart home, etc.

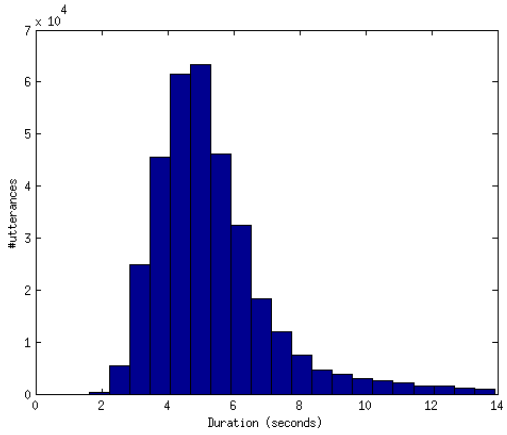

Fig. 1. The distribution of utterance durations.

To achieve this target, in the first phase, we collect 500-hour read speech mainly in the northern dialect. Speakers are recruited from our call center. We first collect text from online newspapers and Wikipedia. After cleaning, sentence segmentation is applied and text is then sent to speakers sentence

[1]https://en.wikipedia.org/wiki/List_of_languages_by_number_of_native_speakers

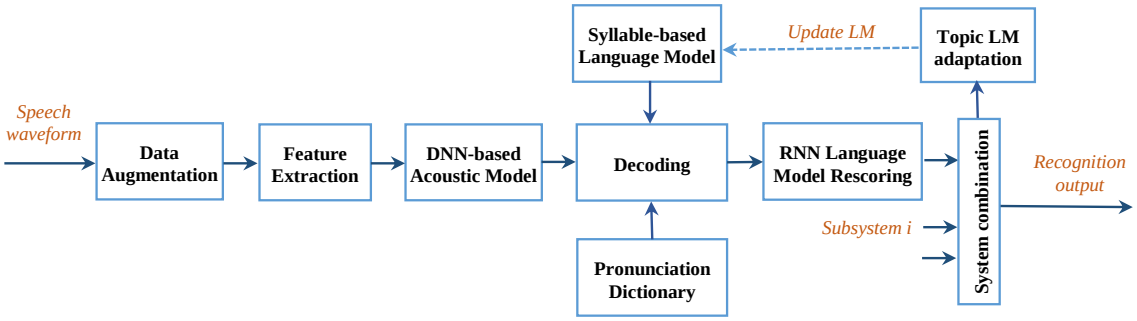

Figure 2. The proposed speech recognition system.

by sentence for speaking and recording. We create a friendly user interface website to help speakers and reviewers to be able to record and supervise easily.

The corpus is recorded with a sampling rate of 16kHz and a resolution of 16 bits/sample. In the corpus, there are 1,424 speakers with totally 343,115 utterances. To improve the corpus quality, each utterance is reviewed by a least one reviewer to warranty speech with good quality and the transcript and speech content are matched.

Figure 1 shows the distribution of utterance durations. The range of duration is from 2 to 14 seconds with the average duration of each utterance is 5.3 seconds.

III. The Proposed System

Our target is to build a speech recognition system which is robust to different recording environments. To achieve to this goal, training data are first augmented by adding various types of noise. Feature extraction is then applied to use for the acoustic model. For decoding, acoustic model is used together with syllable-based language model and pronunciation dictionary. After decoding, recognition output is rescored using RNN language model. The output generated by individual subsystems are combined to achieve further improvement. The recognition output is then used to select relevant text from the text corpus to adapt the language model. The decoding process is then repeated for the second time. In the next subsections, the detailed description of each module is presented.

A. Data Augmentation

To build a reasonable acoustic model, thousands hours of audio recorded in different environments are needed. However, to achieve transcribed audio data is very costly. To overcome this, many techniques have been proposed such as semi-supervised training [9], phone mapping [10], exemplar-based model [11], mismatched crowdsourcing [12]. In this paper, we use a simple approach to simulate data in different noisy environments. Specifically, we collect some popular noise types such as office noise, street noise, car noise, etc. After that noise is added to the clean speech of the original speech corpus with different level to simulate noisy speech. With this approach, we can easily increase the data quantity to avoid over-fitting and improve the robustness of the model against different test conditions.

B. Feature Extraction

We use Mel-frequency cepstral coefficients (MFCCs) [13], without cepstral truncation are used as input feature i.e., 40 MFCCs are computed at each time step which is similar setup in [14]. Since Vietnamese is a tonal language, pitch feature is used to augment MFCC.

Beside MFCC feature, bottleneck feature (BNF) [15] is also considered to build our second subsystem. BNF is generated using a neural network with several hidden layers where the size of the middle hidden layer (bottleneck layer) is very small. With this structure, we can choose an arbitrary feature size without using dimensionality reduction step, independently on the neural network training targets.

C. Acoustic Model

We use time delay neural network (TDNN) and bi-directional long-short term memory (BLSTM) with lattice-free maximum mutual information (LF-MMI) criterion [16] as the acoustic model.

D. Pronunciation Dictionary

Vietnamese is a monosyllabic tonal language. Each Vietnamese syllable can be considered as a combination of initial, final and tone components. Therefore, the pronunciation dictionary (lexicon) needs to be modelled with tones. As in [17], we use 47 basic phonemes. Tonal marks are integrated into the last phoneme of syllable to build the pronunciation dictionary for 6k popular Vietnamese syllables.

In order to build the dictionary for foreign, we select 5k popular foreign words from web newspapers. These words are then manually pronounced in the Vietnamese pronunciation. As a result, the total number of words in our lexicon is about 11k words. This lexicon is used for training as well as decoding.

E. Language Model

A syllable-based language model is built from 900MB web text collected from online newspapers. 4-gram language model with Kneser-Ney smoothing is used after exploring different configuration.

To get further improvement, after decoding, recurrent neural network language model (RNNLM) is used to rescore decoding lattices with a 4-gram approximation as described in [18].

F. System Combination

As described above, we have two subsystems i.e., the first subsystem uses MFCC feature while the second system uses bottleneck feature. The combination of information from different ASR subsystems generally improves speech recognition accuracy. The reason for this advantage is explained by the fact that different subsystems often provide different errors. In this paper, we examine the combination of our two subsystems using the minimum Bayes risk (MBR) decoding method described in [19], which we view as a systematic way to perform confusion network combination (CNC) [20].

G. Language Model Adaptation

The recognition output of our system has a relatively low word error rate (WER). Hence, from decoded text, we can know about the topic of the input utterances. This is especially important when we have no domain information.

Our algorithm is implemented as follows. The in-domain language model is constructed by using the recognition output. After that sentences from the general text corpus (900MB in this paper) are selected based on a cross-entropy difference metric. Detailed description about this selection algorithm can be referred in [21]. Finally, about 200MB text which have the most relevant to the recognition output are selected to build the adapted language model. The decoding process is then repeated with the new language model.

IV. Experiments

To evaluate our system performance, a test set is selected from our 500 hour corpus which is separated from the training set. The test set contains 2000 utterances with around 3 hours of audio. To simulate the real condition, the test set is added different noise with signal to noise ratio (SNR) from 15-40 dB.

A. Data Augmentation

We first examine the effect of data augmentation to the system performance. In this case MFCC feature is used. As shown in Table I, by applying data augmentation brings a big improvement. When the original training data are used only i.e., without data augmentation, the system is only trained with clean speech while test set is noisy. Hence, the model cannot recognize efficiently. By applying data augmentation, the original training data is multiplied by 11 times by adding various types of noise. Obviously, this makes model more robust with noise conditions and hence we achieve a low WER at 10.3%.

TABLE I. Effect of data augmentation to system performance.

Data augmentation	Word Error Rate (%)
No	28.2
Yes	10.3

B. RNNLM Rescoring

As shown in Table II, by applying RNNLM rescoring technique, we can achieve 1.4% absolute improvement.

TABLE II. Effect of RNNLM rescoring to system performance.

RNNLM Rescoring	Word Error Rate (%)
No	10.3
Yes	8.9

C. System Combination

The systems in the previous subsections are trained using MFCC feature. In this subsection, we investigate the effect of using bottleneck feature and its usefulness in system combination.

As shown in Table III, using BNF does not provide a good performance as MFCC. However, it provides complementary information and hence we can gain by combining them.

TABLE III. Bottleneck feature and system combination.

Subsystem	Word Error Rate (%)
Subsystem 1 (MFCC)	8.9
Subsystem 2 (BNF)	9.5
Combined system	8.1

D. Language Model Adaptation

As shown in Table IV, by applying language model adaptation, a significant WER reduction is achieved. It can be explained that the algorithm only chooses relevant (in-domain) sentences, while mismatched (out-domain) sentences which can be harmful to language model are discarded.

TABLE IV. Effect of language model adaptation to system performance.

Language model adaptation	Word Error Rate (%)
No	8.1
Yes	6.9

V. Conclusions

In this paper, we have described our 500-hour speech corpus. Various techniques such as data augmentation, RNNLM rescoring, language model adaptation, bottleneck feature, system combination were then applied. Our final system achieves a low word error rate at 6.9% on the noisy test set.

In the future, we will enlarge the speech corpus to cover most of the popular dialects in Vietnamese with different aging ranges as well as enlarge the text corpus to make our system more robust and achieve even better performance.

References

[1] Thang Tat Vu, Dung Tien Nguyen, Mai Chi Luong, and John-Paul Hosom, "Vietnamese large vocabulary continuous speech recognition," in Proc. *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2005, pp. 492–495.

[2] Quan Vu, Kris Demuynck, and Dirk Van Compernolle, "Vietnamese automatic speech recognition: The flavour approach," in Proc. *the 5th International Conference on Chinese Spoken Language Processing (ISCSLP)*, 2006, pp. 464–474.

[3] Tuan Nguyen and Quan Vu, "Advances in acoustic modeling for Vietnamese LVCSR," in Proc. *International Conference on Asian Language Processing (IALP)*, 2009, pp. 280–284.

[4] Ngoc Thang Vu and Tanja Schultz, "Vietnamese large vocabulary continuous speech recognition," in Proc. *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2009.

[5] Nancy F. Chen, Sunil Sivadas, Boon Pang Lim, Hoang Gia Ngo, Haihua Xu, Bin Ma, and Haizhou Li. "Strategies for Vietnamese keyword search," in Proc. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 4121-4125.

[6] Tsakalidis, Stavros, Roger Hsiao, Damianos Karakos, Tim Ng, Shivesh Ranjan, Guruprasad Saikumar, Le Zhang, Long Nguyen, Richard Schwartz, and John Makhoul. "The 2013 BBN Vietnamese telephone speech keyword spotting system," in Proc. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2014, pp. 7829-7833.

[7] I-Fan Chen, Nancy F. Chen, and Chin-Hui Lee, "A keyword-boosted sMBR criterion to enhance keyword search performance in deep neural network based acoustic modeling," in Proc. *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2014.

[8] Quoc Bao Nguyen, Van Hai Do, Ba Quyen Dam, Minh Hung Le, "Development of a Vietnamese Speech Recognition System for Viettel Call Center," in Proc. *Oriental COCOSDA*, pp. 104-108, 2017.

[9] Haihua Xu, Hang Su, Eng Siong Chng, and Haizhou Li. "Semi-supervised training for bottle-neck feature based DNN-HMM hybrid systems," in Proc. *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2014.

[10] Van Hai Do, Xiong Xiao, Eng Siong Chng, and Haizhou Li, "Context-dependent phone mapping for LVCSR of under-resourced languages," in Proc. *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2013, pp. 500–504.

[11] Van Hai Do, Xiong Xiao, Eng Siong Chng, and Haizhou Li, "Kernel Density-based Acoustic Model with Cross-lingual Bottleneck Features for Re-source Limited LVCSR," in Proc. *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2014, pp. 6–10.

[12] Van Hai Do, Nancy F. Chen, Boon Pang Lim and Mark Hasegawa-Johnson, "Multi-task Learning using Mismatched Transcription for Under-resourced Speech Recognition," in Proc. *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 734-738, 2017

[13] S. B. Davis and P. Mermelstein, "Comparison of parametric representation for monosyllabic word recognition in continuously spoken sentences," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 28, no. 4, pp. 357–366, 1980.

[14] Vijayaditya Peddinti, Daniel Povey, and Sanjeev Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in Proc. *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2015, pp. 3214-3218.

[15] F. Grezl, M. Karafiat, S. Kontar, and J. Cernock, Probabilistic and bottleneck features for LVCSR of meetings," in Proc. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2007, vol. 4 pp. 757-760.

[16] D. Povey, V. Peddinti, D. Galvez, P. Ghahrmani, V. Manohar, X. Na, Y. Wang, and S. Khudanpur, "Purely sequence-trained neural networks for ASR based on lattice-free MMI", in Proc. *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 2751–2755, 2016.

[17] Quoc Bao Nguyen, Tat Thang Vu, and Chi Mai Luong, "The Effect of Tone Modeling in Vietnamese LVCSR System," *Procedia Computer Science* 81 (2016): 174-181.

[18] Xunying Liu, Yongqiang Wang, Xie Chen, Mark Gales, and P. C. Woodland, "Efficent lattice rescoring using recurrent neural network language models," in Proc. *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2014.

[19] H. Xu, D. Povey, L. Mangu, and J. Zhu, "Minimum Bayes Risk Decoding and System Combination Based on a Recursion for Edit Distance," *Computer Speech & Language*, vol. 25, no. 4, pp. 802 – 828, 2011.

[20] G. Evermann and P. C. Woodland, "Posterior Probability Decoding, Confidence Estimation and System Combination," in Proc. *Speech Transcription Workshop*, 2000.

[21] P. Bell, H. Yamamoto, P. Swietojanski, Y. Z. Wu, F. McInnes, C. Hori, and S. Renals, "A Lecture Transcription System Combining Neural Network Acoustic and Language Model," in Proc. *Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2013