INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Prosodic Boundary Prediction Model for Vietnamese Text-To-Speech

Nguyen Thi Thu Trang[1], *Nguyen Hoang Ky*[1], *Albert Rilliard*[2], *Christophe d'Alessandro*[3]

[1]Hanoi University of Science and Technology, Vietnam
[2]Université Paris Saclay, CNRS, LISN, France
[3]Institut Jean le Rond d'Alembert, Sorbonne Université, UMR7190 CNRS, Paris France

trangntt@soict.hust.edu.vn, hoangky17101996@gmail.com, albert.rilliard@lisn.upsaclay.fr, christophe.dalessandro@sorbonne-universite.fr

Abstract

This research aims to build a prosodic boundary prediction model for improving the naturalness of Vietnamese speech synthesis. This model can be used directly to predict prosodic boundaries in the synthesis phase of the statistical parametric or end-to-end speech systems. Beside conventional features related to Part-Of-Speech (POS), this paper proposes two efficient features to predict prosodic boundaries: syntactic blocks and syntactic links, based on a thorough analysis of a Vietnamese dataset. Syntactic blocks are syntactic phrases whose sizes are bounded in their constituent syntactic tree. A syntactic link of two adjacent words is calculated based on the distance between them in the syntax tree. The experimental results show that the two proposed predictors improve the quality of the boundary prediction model using a decision tree classification algorithm, about 36.4% (F1 score) higher than the model with only POS features. The final boundary prediction model with POS, syntactic block, and syntactic link features using the LightGBM algorithm gives the best F1-score results at 87.0% in test data. The proposed model helps the TTS systems, developed by either HMM-based, DNN-based, or End-to-end speech synthesis techniques, improve about 0.3 MOS points (i.e. 6 to 10%) compared to the ones without the proposed model.

Index Terms: Prosody modeling, prosodic boundary, pause prediction, Text-To-Speech, speech synthesis, Vietnamese

1. Introduction

For a Text-To-Speech (TTS) system, prosodic phrasing plays a significant role in improving the intelligibility and naturalness of synthetic utterances. One of the most important levels in prosodic phrasing is prosodic boundary modeling. It simulates the reader's breathing pattern and the method of reading based on sentence context in practice. To the best of our knowledge, TTS systems currently have been divided into three main approaches: (i) concatenative speech synthesis, (ii) parametric statistical speech synthesis, and (iii) end-to-end speech synthesis systems. In the parametric statistical speech synthesis approach, the acoustic model can be HMM[1]-based [1][2] or DNN[2]-based [3]. However, those models cannot automatically predict pause appearances and need to provide pause positions at the text pre-processing step. In end-to-end TTS systems, such as Tacotron [4][5], Tacotron2 [6], and Wavenet [7], we could use a massive amount of text and audio data pairs to learn the prosodic structure directly during the TTS training process. Nevertheless, corpora are not always designed to support such a purpose. On the other hand, pause analysis during the pre-processing steps also helps end-to-end TTS models be easier to train and faster to converge.

There are many previous works about prosodic structure generation for Chinese [8][9][10], pause modeling for German [11], Russian [12] or prosodic structure for French [13], and other languages. They may use rules or machine learning with lexical information (e.g. Part-of-speech POS) or contextual lengths.

For Vietnamese, a tonal language, there are few works on these problems. The work of [14] presented a rule-based prosodic phrasing model based on syntactic information, with an alignment to the theory of prosodic hierarchy [15][16][17], for a Vietnamese HMM-based TTS system. For the task of predicting the pause appearance, this work had good precision (i.e. 91% and 82% with manual and automatic syntactic parsing), but low recall (i.e. 54% and 27% correspondingly) due to a limited number of syntactic rules. In this paper, we aim on building a prosodic boundary prediction model for Vietnamese TTS systems. Since pause duration has been already well modeled based on the context, we propose a prediction model using syntactic information to figure out the suitable pause positions in sentences.

The rest of this paper is organized as follows. Section 2 describes the dataset and evaluation metrics used in this work. Section 3 presents the proposal of syntactic blocks, syntactic links, and the prosodic boundary prediction model. Our Vietnamese syntactic parser for the proposed model is presented in section 4. Section 5 shows the experiment results for both the prediction model and the contribution of the proposed model to TTS systems with different speech synthesis techniques. The final section presents conclusions and future work.

2. Datasets and Metrics

2.1. Datasets

The VDTO dataset included 5,338 utterances for a duration of about 7.7 hours. The audio was recorded at LIMSI-CNRS Laboratory, France, by a female semi-professional native speaker from Hanoi, aged 31. The supervision during the recording sessions and verification after the recording sessions were well performed. Audio files in this dataset were automatically segmented at a phoneme level by EHMM labeler. Phonemes were then automatically grouped into syllables and semi-automatically perceived pauses. Text files were automatically parsed into syntactic trees by our Vietnamese parser presenting in section 4. For the evaluation phase, we extracted randomly 10% of the sentences in VDTO as a test dataset, called VDTO-Testing, the rest constituting the training dataset, called VDTO-Training.

[1] Hidden Markov Model
[2] Deep Neural Network

http://dx.doi.org/10.21437/Interspeech.2021-125

2.2. Evaluation metrics

For the prosodic boundary prediction, Precision (P) is defined as the probability that a (randomly selected) predicted pause corresponds to an actual (correct) pause in the dataset, while Recall (R) is defined as the probability that a (randomly selected) actual pause in the dataset is predicted. A measure that combines $Precision$ and $Recall$ is the harmonic mean of $Precision$ and $Recall$, the F-score, illustrated in Formula 2.

$$Precision = \frac{CP}{PP} \qquad Recall = \frac{CP}{AP} \tag{1}$$

$$F-score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{2}$$

where

- PP: Number of **P**redicted **P**auses
- CP: Number of **C**orrect predicted **P**auses
- AP: Number of **A**ctual **P**auses in dataset

3. Our proposed model

3.1. Proposal of syntactic blocks

In the syntactic trees, syntactic phrases, i.e. ancestors of syllables, can be effective cues to predict prosodic boundaries. However, the sizes of these phrases (i.e. the numbers of syllables) vary, in the dataset, from 1 to 70 syllables due to the structure and complexity of the sentence. We hence proposed that the levels of ancestors should be flexible, and the size of these phrases should be bounded. These “bounded” syntactic phrases were called “syntactic blocks”.

Figure 1 illustrates the examination of actual pauses in the middle of utterances after syntactic blocks. Pause presence could be roughly predicted by syntactic block size, i.e. at the end of blocks having at least 5 syllables. Based on this raw model, we empirically found the optimized value for n of 10, i.e. maximal syllable number of all syntactic blocks, in which F-score was maximal.

Our proposed algorithm for dividing syntactic blocks of a sentence is illustrated in Algorithm 1. Let n be the bounded size of syntactic blocks. We extracted the first children of the root node. If any of them had more than n syllables, we kept dividing them to lower children until syllable numbers of all syntactic blocks were not above n. There existed a number of single-syllable syntactic blocks in the datasets, which may affect the block predictor. We then proposed several strategies for a combination of single-syllable syntactic blocks. The base strategy is that a single-syllable syntactic block will be combined with its succeeding block. However, if the single-syllable block is the last syllable in the sentence, it will be combined with the previous one.

3.2. Proposal of syntactic links

The syntactic link of two words is a syntax tree-based relationship between them. Four special values for this predictor “l1”, “l2”, “h1”, “h2” show that if the current word was lower (l) or higher (h) one (1) or two (2) levels in the same branch. In other cases, the distance between two nodes in a syntax tree is used to determine this relationship. The distance $D(W_i, W_{i-1})$ between two words W_i and W_{i-1}, shown in Formula 3, is calculated as an average of the number of transitions from W_i and W_{i-1} to the lowest common ancestor. If the distance is “1”, two nodes were siblings. The syntactic link was the ceiling of this distance excluding the distance of over 3. For all distances over “3”, one value “4” was assigned for the syntactic link. The syntactic link $L(W_i)$ of a word W_i (with the previous word W_{i-1}) is defined in Formula 4.

$$Distance(W_i, W_{i-1}) = D(W_i, W_{i-1}) = \frac{T_i + T_{i-1}}{2} \tag{3}$$

$$L(W_i) = L(W_i, W_{i-1}) =$$
$$\begin{cases} l1 \; or \; l2 & \text{if } W_i \text{ was 1 or 2 level lower than } W_{i-1} \\ h1 \; or \; h2 & \text{if } W_i \text{ was 1 or 2 level higher than } W_{i-1} \\ 1 & \text{if } D(W_i, W_{i-1}) = 1 \text{ (siblings)} \\ 2 & \text{if } 1 < D(W_i, W_{i-1}) \leq 2 \\ 3 & \text{if } 2 < D(W_i, W_{i-1}) \leq 3 \\ 4 & \text{if } D(W_i, W_{i-1}) > 3 \end{cases} \tag{4}$$

where

- W_i: The ith word in sentence
- W_{i-1}: The (i-1)th word in sentence
- T_i: Number of branch transitions from W_i to the lowest common ancestor of W_i and W_{i-1}
- T_{i-1}: Number of branch transitions from W_{i-1} to the lowest common ancestor of W_i and W_{i-1}
- L: Syntactic link

3.3. Proposal of prosodic boundary prediction model

Beside the two above proposed predictors, we did find that POS of the last word (current POS) and that of the next word (next POS) of syntactic blocks could be used to predict pauses with ambiguity. We then applied a machine-learning algorithm to evaluate these predictors' performance. As a result, the prosodic boundary prediction model was considered as a classification task of two labels “Yes” or “No” for the target value “Has pause”. We experimented with some state-of-the-art classification algorithms in section 5.

4. Unnamed Vietnamese syntactic parser

This section presents our proposal for a Vietnamese syntactic parser, which is necessary for our prosodic boundary prediction model. Despite a considerable gap in quality to other common languages such as Chinese, Japanese, or English [18] (e.g. F-score for English = 95%-96%), there have been several popular constituent syntactic parsers for Vietnamese, whose best performance was at 81.19% [19] (F-score) in the VietTreeBank dataset [20].

In this work, we built a Vietnamese syntactic parsing using a state-of-the-art deep neural architecture, i.e. self-attentive architecture [21] and applied the pre-trained Vietnamese language model, PhoBERT-large [22], to embed the input texts. This model, trained with the VietTreeBank dataset [20] with some corrections, received the best performance at 84.4% accuracy, compared to other well-known Vietnamese parsers.

In our proposal of the prosodic boundary prediction model, phrase names, e.g. noun phrase (NP), verb phrase (VP), were not used. As a result, to improve the quality of the syntactic parser, we propose to build an unnamed constituency parser, in which all phrases above word will have only one label XP. As a result, sentences are parsed into syntax trees, leaves of these

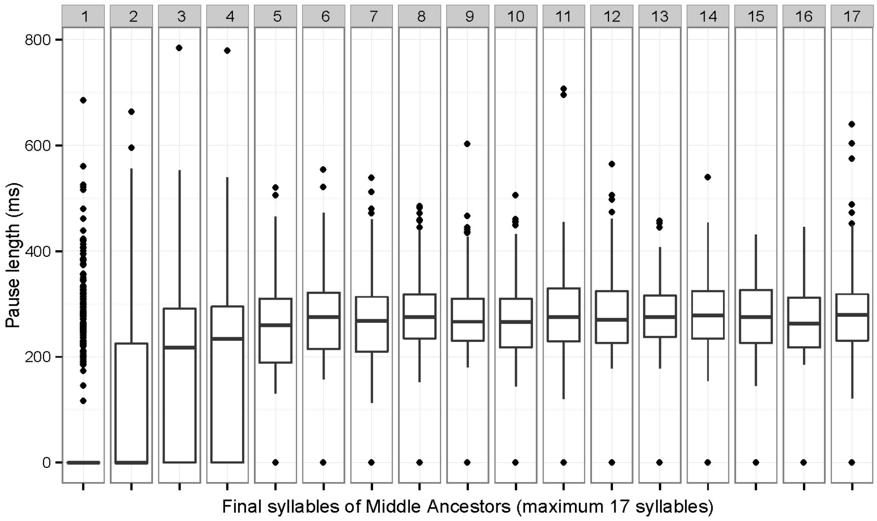

Figure 1: *Distributions of pause length of final syllable of syntactic blocks with a maximum of 17 syllables, factored by syllable numbers of these blocks.*

trees are grammatical words named by POS categories while ancestor nodes are unnamed syntactic phrases. The unnamed constituency parsing improved the syntactic model's accuracy to 87.0%.

5. Experiments

5.1. Experiment for the prosodic boundary model

After a random division, the VDTO-Training dataset contained nearly seven hours of speech while the testing one had nearly one hour. Necessary features such as POS, next POS, syntactic block size, the position in syntactic block, syntactic link, etc. were then extracted for the dataset at syllable level using our proposal. In this work, the C4.5 algorithm, an improvement of the ID3 [23], was adopted for experimenting with different predictors due to its simplicity and effectiveness. This algorithm can generate a decision tree from a dataset with extracted features for a classification "Yes" or "No" for the target value "Has pause".

Table 1 shows the performance of the models using C4.5 in 10-fold cross-validation. The "Syntactic block" was the most efficient predictor in both Precision (83.4%) and Recall (71.1%). The Precision of the model using "POS" alone (73.4%) was higher than that of the one using "Syntactic link" alone (64.4%). However, "Syntactic link" (Recall=43.7%) could predict more pauses than the "POS" predictor (Recall=31.0%). Using only "POS", the F-score of the model was only 43.6%, that is 9% lower than that using only "Syntactic link". The combination of two out of the three predictors gave better results in Precision and/or Recall. The model with "Syntactic block" and "Syntactic link" had the same Precision as that with only "Syntactic block", but Recall increased nearly 6%. Whereas, the model with "Syntactic block" and "POS" almost had no progress on Recall, but about 4% higher on Precision. We assumed that the "Syntactic link" helped us improve the Recall while "POS" gave effective information to increase the Precision. The complete model including the three predictors had the best results with Precision=89.0%, Recall=74.6%, hence F-score=81.2%. Using a separate test set (VDTO-Testing), the

Algorithm 1: Proposal algorithm for dividing syntactic blocks bounded by a limit number of syllables

```
Input  : Node $phrase$ with limit length $n$
Output: List of syntactic blocks $blocks$
Function getBlocks ($phrase, n$):
    if $size(phrase) \leq n$ then
        return $phrase$
    else
        foreach $child \in phrase$ do
            $blocksOfChild \leftarrow getBlocks(child, n)$
            $blocks \leftarrow blocks + blocksOfChild$
        end
    end
    $blocks \leftarrow combineSingleBlocks(blocks)$
    return $blocks$
End Function
Function combineSingleBlocks ($blocks$):
    foreach $block \in blocks$ do
        if $isSingle(block)$ then
            if $isLast(block)$ then
                $combine\ block\ to\ last\ block\ of\ blocks$
            else
                $newBlock \leftarrow newBlock + block$
                if $isNotSingle(newBlock)$ & $isNotSingle(nextBlock)$ then
                    $add\ newBlock\ to\ blocks$
                    $newBlock \leftarrow null$
                end
            end
        else
            $add\ block\ to\ newBlock$
        end
    end
    return $blocks$
End Function
```

Table 1: *Performance of prosodic boundary prediction models with C4.5 algorithm using different features*

Features	**Precision** (%)	**Recall** (%)	**F-score** (%)
10-fold cross-validation			
Syntactic block	83.4	71.1	76.8
Syntactic link	65.4	43.7	52.6
POS	73.4	31.0	43.6
Syntactic-block+link	83.4	76.8	80.0
Syntactic block+POS	87.2	71.4	78.6
Syntactic link+POS	70.6	58.7	61.4
Syntactic-block+link+POS	**89.0**	**74.6**	**81.2**
VDTO-Testing			
Syntactic-block+link+POS	**87.6**	**75.9**	**81.4**

F-score was approximate to the F-score in the best results of 10-fold cross-validation.

Table 2: *Performance of prosodic boundary predictive models with different classifier algorithms*

Model name	**Precision** (%)	**Recall** (%)	**F-score** (%)
C4.5	87.6	75.9	81.4
Adaboost	84.6	79.9	82.2
XgBoost	85.5	83.1	84.3
RandomForest	87.1	83.3	85.1
LightGBM	**88.1**	**86.1**	**87.0**

We took the same dataset extracted with all three predictors to experiment with some other classification algorithm. C4.5, a fast decision tree algorithm, provides a good precision but not comparative recall, hence the lowest F-score among classification algorithms. Ensemble models are very simple and effective, they group weak individual classifiers into a single strong classifier. The tree-based model RandomForest [24] has an extremely fast training time due to the parallel training process. Boosting models, such as Adaboost [25], XgBoost [26], and LightGBM [27], improve every training steps by creating a next model that attempts to correct the errors from the previous model. Table 2 shows the performance of those five models. LightGBM gave the best results in all Precision, Recall, and F-score (87.0%), increase 5.6% compare to the F-score of the C4.5 algorithm.

5.2. Experiment for applying the proposed model to TTS

The purpose of this test is to see how the prosodic boundary model affects the naturalness of the TTS output voices, synthesized by different technologies: HMM-based [28], DNN-based (built with model [3] using Vietnamese linguistic features), and End-to-end (i.e. Tacotron [4]+Wavenet [7]). The HMM-based TTS system was trained with the VDTO dataset, which was recorded by a female speaker from Northern Vietnam (SPK01). The DNN-based TTS system was trained with another corpus of a male speaker from Northern Vietnam (SPK02), while the end-to-end TTS system uses the voice of another female from Northern Vietnam (SPK03) as the training dataset. Subjects were asked to assess the speech they had heard. The question presented to subjects was "How do you rate the naturalness of the sound you have just heard?", with the maximal score of 5.0.

Table 3 shows our MOS Test results on two main types of TTS systems: (i) TTS system without the proposed prosodic boundary prediction model (i.e. Baseline), (ii) TTS system with the proposed model (i.e. Boundary). In all speech synthesis techniques, the output voices with prosodic boundary prediction of the corresponding TTS systems were evaluated about 0.3 points better than the ones without the proposed model (i.e. baseline). In other words, the synthetic voices with the prosodic boundary prediction model improved about 6 to 10% compared to the one without the model. The results of two-factorial ANOVA show that all factors (i.e. TTS system and sentence) and their interactions have a highly significant effect ($p < 0.001$).

Table 3: *MOS Test for naturalness of voices without (Baseline) or with (Boundary) prosodic boundary prediction model*

Voice	**Synthesis Technology**	**MOS Score**
SPK01-Baseline	HMM	3.12
SPK01-Boundary	**HMM**	**3.48**
SPK02-Baseline	DNN	3.64
SPK02-Boundary	**DNN**	**3.98**
SPK03-Baseline	End-to-End	3.98
SPK03-Boundary	**End-to-End**	**4.21**

6. Conclusion

In this work, we proposed a prosodic boundary prediction tree-based model using three predictors: (i) syntactic blocks, (ii) syntactic links, and (iii) POS. Syntactic blocks are syntactic phrases whose sizes are bounded. The syntactic link of a word was a syntax tree-based relationship with the previous word. A predictive model was evaluated with 10-fold cross-validation using these three predictors using C4.5, a decision tree classification algorithm. The syntactic block was the most important predictor since the model with only this predictor had the best precision (83.4%), compared to the models with only POS (F-score=43.6%) or syntactic link (F-score=52.6%) alone. The syntactic link predictor helped the model improve the Recall (6% improved) while POS gave effective information to increase the Precision (4% improved). We experimented with different state-of-the-art classification algorithms using the three predictors. The model with the LightGBM algorithm had the best result of F-score at 87.0% in test data. The proposed model could help to increase around 0.3 MOS points (i.e. 6 to 10%) when applying for all HMM-based, DNN-based, and end-to-end state-of-the-art TTS systems. In the future, we will extend this work to other languages to explore similarities and differences as well as to define common patterns for a multi-lingual prosodic boundary prediction model.

7. Acknowledgements

This work was funded by Gia Lam Urban Development and Investment Company Limited, Vingroup and supported by Vingroup Innovation Foundation (VINIF) under project code DA116_14062019.

8. References

[1] M. Schröder, M. Charfuelan, S. Pammi, and I. Steiner, "Open source voice creation toolkit for the MARY TTS Platform," in *Twelfth annual conference of the international speech communication association*, 2011.

[2] M. Charfuelan, "MARY TTS HMM-based voices for the Blizzard Challenge 2012," in *Blizzard Challenge*, 2012.

[3] Z. Wu, O. Watts, and S. King, "Merlin: An Open Source Neural Network Speech Synthesis System." in *SSW*, 2016, pp. 202–207.

[4] Y. Wang, R. Skerry-Ryan, D. Stanton, Y. Wu, R. J. Weiss, N. Jaitly, Z. Yang, Y. Xiao, Z. Chen, S. Bengio *et al.*, "Tacotron: Towards end-to-end speech synthesis," *arXiv preprint arXiv:1703.10135*, 2017.

[5] R. Skerry-Ryan, E. Battenberg, Y. Xiao, Y. Wang, D. Stanton, J. Shor, R. J. Weiss, R. Clark, and R. A. Saurous, "Towards end-to-end prosody transfer for expressive speech synthesis with tacotron," *arXiv preprint arXiv:1803.09047*, 2018.

[6] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Skerrv-Ryan *et al.*, "Natural TTS synthesis by conditioning wavenet on mel spectrogram predictions," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 4779–4783.

[7] A. v. d. Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "Wavenet: A generative model for raw audio," *arXiv preprint arXiv:1609.03499*, 2016.

[8] F.-C. Chou, C.-y. Tseng, and L.-S. Lee, "Automatic generation of prosodic structure for high quality Mandarin speech synthesis," in *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP'96*, vol. 3. IEEE, 1996, pp. 1624–1627.

[9] J. Tao, H. Dong, and S. Zhao, "Rule learning based Chinese prosodic phrase prediction," in *International Conference on Natural Language Processing and Knowledge Engineering, 2003. Proceedings. 2003*. IEEE, 2003, pp. 425–432.

[10] C. Lu, P. Zhang, and Y. Yan, "Self-attention based prosodic boundary prediction for Chinese speech synthesis," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 7035–7039.

[11] J. Apel, F. Neubarth, H. Pirker, and H. Trost, "Have a break! Modelling pauses in German speech," in *KONVENS*, 2004, pp. 5–12.

[12] P. Chistikov and O. Khomitsevich, "Improving prosodic break detection in a Russian TTS system," in *International Conference on Speech and Computer*. Springer, 2013, pp. 181–188.

[13] D. Doukhan, A. Rilliard, S. Rosset, and C. d'Alessandro, "Modelling pause duration as a function of contextual length," *13th Annual Conference of the International Speech Communication Association 2012, INTERSPEECH 2012*, vol. 1, pp. 658–661, 01 2012.

[14] T. T. T. Nguyen, A. Rilliard, D. D. Tran, and C. d'Alessandro, "Prosodic phrasing modeling for Vietnamese TTS using syntactic information," in *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.

[15] E. O. Selkirk, *On prosodic structure and its relation to syntactic structure*. Indiana University Linguistics Club, 1980, vol. 194.

[16] M. Nespor and I. Vogel, "Prosodic structure above the word," in *Prosody: Models and measurements*. Springer, 1983, pp. 123–140.

[17] B. Hayes, "The prosodic hierarchy in meter," in *Rhythm and meter*. Elsevier, 1989, pp. 201–260.

[18] M. Zhang, "A survey of syntactic-semantic parsing based on constituent and dependency structures," *Science China Technological Sciences*, pp. 1–23, 2020.

[19] T.-V. Tran, X.-T. Pham, D.-V. Nguyen, K. Van Nguyen, and N. L.-T. Nguyen, "An Empirical Study for Vietnamese Constituency Parsing with Pre-training," *arXiv preprint arXiv:2010.09623*, 2020.

[20] P. T. Nguyen, X. L. Vu, T. M. H. Nguyen, H. P. Le *et al.*, "Building a large syntactically-annotated corpus of Vietnamese," in *The Third Linguistic Annotation Workshop-The LAW III*, 2009, p. 6p.

[21] N. Kitaev and D. Klein, "Constituency parsing with a self-attentive encoder," *arXiv preprint arXiv:1805.01052*, 2018.

[22] D. Q. Nguyen and A. T. Nguyen, "PhoBERT: Pre-trained language models for Vietnamese," *arXiv preprint arXiv:2003.00744*, 2020.

[23] J. R. Quinlan, "Induction of decision trees," *Machine learning*, vol. 1, no. 1, pp. 81–106, 1986.

[24] P. Bonissone, J. M. Cadenas, M. C. Garrido, and R. A. Díaz-Valladares, "A fuzzy Random Forest," *International Journal of Approximate Reasoning*, vol. 51, no. 7, pp. 729–747, 2010.

[25] R. E. Schapire, "Explaining AdaBoost," in *Empirical inference*. Springer, 2013, pp. 37–52.

[26] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho *et al.*, "XGBoost: Extreme gradient boosting," *R package version 0.4-2*, vol. 1, no. 4, 2015.

[27] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.-Y. Liu, "LightGBM: A highly efficient gradient boosting decision tree," *Advances in neural information processing systems*, vol. 30, pp. 3146–3154, 2017.

[28] T. T. T. Nguyen, C. d'Alessandro, A. Rilliard, and D. D. Tran, "The HMM-based TTS for Hanoi Vietnamese: Issues in design and evaluation," in *Annual Conference of the International Speech Communication Association (INTERSPEECH 2013)*, 2013, pp. 2311–2315.