A Vietnamese Named Entity Recognition System for COVID-19 Articles

Ngoc Nhu Hoang
New York University Abu Dhabi
ngoc.hoang@nyu.edu

Abstract

This paper presents a named entity recognition system for the specific domain of Vietnamese COVID-19 news articles. By incorporating manually selected and domain-specific features into a simple deep learning architecture, the system can identify a wide range of custom named entities relevant in the context of COVID-19 and future epidemics. Using high-dimensional embedding vectors in combination with part-of-speech tags and additional features, the system achieves an F score of about 90.41%, surpassing or coming close to results by other models that are more complicated or pre-trained and fine tuned.

1 Introduction

The ongoing of the COVID-19 pandemic results in a continuous surge of information produced to cover details of outbreaks and to broadcast safety measures and government mandates. In Vietnam, official reports at the national and local levels and news coverage constitute a source of thorough, in-depth records of new cases and outbreaks with great details on patients' travel histories, contributing significantly to contact tracing and quarantine mandating. The continued growth of the volume of textual information results in a need for a system capable of processing the texts and identifying important information and key parties so that all related people and organizations can be timely informed (Truong et al., 2021). The task can be framed as the problem of recognizing key names, locations, and other relevant noun phrases present in the text. A prominent technique to use in solving such a problem is named entity recognition.

Named entity recognition is among the fundamental tasks in natural language processing, consisting of identifying phrases that constitute proper entities and classifying them into appropriate entity types. Named entity recognition falls under information extraction and is an essential step for further tasks, such as question answering. In Vietnamese, conventional named entity recognition tasks have involved developing systems that train on data extracted from online news sites and focus on identifying a number of generic named entity types, including person, location, organization, and miscellaneous entities (Pham and Le-Hong, 2017; Minh, 2018). However, due to the nature of the types of information proven valuable to contact tracing and outbreak containment, the entity types are to be customized to be more specific and fitting to the COVID-19 context. The research by Truong et al. (2021) presents the first manually annotated named entity recognition dataset that are specific to COVID-19 and includes a wide range of custom named entities, including patient names, patient IDs, any relevant locations or organizations, etc.

Using this annotated dataset as training and testing data, this paper presents a named entity recognition system that concentrates on the specific domain of COVID-19 news articles and on identifying COVID-19 related entities. The system utilizes a neural network architecture and evaluates the performances of different combinations of features extracted from the data. My work has found that with a simple neural network that uses word embeddings, part-of-speech tags, and addition features as input, the system can be trained on domain-specific data and achieve performances comparable to more sophisticated deep learning models and pre-trained language models. The system achieves an overall F score of 90.41% on the test set of the COVID-19 named entity recognition dataset.

The remaining of this paper is structured as follows. Section 2 discusses some previous works related to named entity recognition in Vietnamese. Section 3 describes the dataset published by Truong et al. (2021) and utilized in this paper. Section 4

then discusses in-depth my methodology, including descriptions of the neural network architecture, data preprocessing steps, feature engineering steps, and description of evaluation method. Section 5 outlines the architecture and configurations used for the neural network. Section 6 presents the results of evaluating the system and different combinations of features. Section 7 discusses the system's performances in comparison to experiments done in Truong et al. (2021), and some insights obtained from error analysis. Sections 8 and 9 conclude the paper and present a few points for future work and development.

2 Related works

The work by Truong et al. (2021) presents the COVID-19 named entity data that is used as the training and validating dataset for this project. This is one of the few existing annotated datasets for the named entity recognition task in Vietnamese, and the first to be narrowed down to the domain of COVID-19. The entity types used in the annotation process for the corpus are also newly defined and modified to fit the domain subject, focusing on patient information, relevant locations, organizations, and any present symptoms, rather than only the generic types of entities as often seen in other named entity datasets. Experiments are conducted on the dataset on the basis of the two forms of representations (syllable-level and word-level) using multiple models, including the BiLSTM-CNN-CRF model (Ma and Hovy, 2016) and two pre-trained language models, the multilingual model XLM-R (Conneau et al., 2020), and the monolingual model PhoBERT (Nguyen and Tuan Nguyen, 2020) pre-trained on Vietnamese. Results of the experiments in this paper show that using word-level settings and employing pre-trained multilingual language models lead to better performance on a language-specific task.

The research by Pham and Le-Hong (2017) uses word embeddings and syntactic features to create inputs into a deep learning system for the general Vietnamese named entity recognition task. The paper evaluates performances of the system on a number of model architectures using Recurrent Neural Network, Long Short Term Memory, and Bidirectional Long Short Term Memory and finds that utilizing Bidirectional Long Short Term Memory to capture past and future contexts improves performances for all types of entities. Additionally, automatic syntactic features incorporated into the simple deep learning model also help to improve F score from 74.02% (without syntactic features) to 92.05% (with syntactic features). The general architecture of my system is based upon the model described in this paper, with more emphasis on feature engineering and feature selection more specific to the domain of the project.

A different research by Pham and Le-Hong (2017) introduces a more sophisticated architecture making use of Bidirectional Long Short Term Memory, Convolutional Neural Network, and Conditional Random Field with pre-trained word embeddings as input without any additional syntactic, handcrafted features. The system tackles the task using word-level approach and character-level approach. The system achieves an F score of 88.59%.

The two papers by Pham and Le-Hong (2017) and Pham and Le-Hong (2017) approach the general named entity recognition task without a specific domain subject and evaluate the performances of systems with different attributes: simpler deep learning model with and without syntactic features, and more sophisticated model using only word embeddings and no additional features. Empirical experiments by Truong et al. (2021) are conducted using a sophisticated model whose components are similar to those in Pham and Le-Hong (2017) and two fine tuned language models. My system uses these previous systems as baseline architectures but focuses on using a simple architecture with additional features, emphasizing on building a system that are specific to the domain knowledge and the nature of the entity types present in the dataset.

3 Data

The data used for training and testing in this project is the COVID-19 named entity recognition dataset made public by Truong et al. (2021).

The dataset is annotated using the BIO tagging format and uses ten newly defined or modified entity types specific to the context of COVID-19, including: PATIENT_ID, PERSON_NAME, AGE, GENDER, OCCUPATION, LOCATION, ORGANIZATION, SYMPTOM&DISEASE, TRANSPORTATION, and DATE. These ten entity types, together with 'O', make up 20-21 unique named entity tags in the dataset (20 for word-level set and 21 for syllable-level set), with each entity type includes B- and I- tags. The dataset is comprised of 10027 sentences, split into training/validation/test

sets with a 5/2/3 ratio (Truong et al., 2021).

The dataset comes in two forms, word-level representation and syllable-level representation. The two sets are identical in terms of content, but differ in that the word-level set treats multi-syllabic words as individual units, with underscores between the syllables. Table 1 shows parts of a sample sentence, its sequence of label tags, and its representations on the word level and syllable level.

Words	Tags	Syllables	Tags
cách_ly	O	cách	O
tại	O	ly	O
Bệnh_viện	B-LOC	tại	O
Bệnh	I-LOC	Bệnh	B-LOC
Nhiệt_đới	I-LOC	viện	I-LOC
Trung_ương	I-LOC	Bệnh	I-LOC
		Nhiệt	I-LOC
		đới	I-LOC
		Trung	I-LOC
		ương	I-LOC

Table 1: Parts of a sample sentence in word-level and syllable-level representations. "quarantine(d)", "at", "hospital", "disease", "tropical", "national", or "quarantine(d) at the National Hospital of Tropical Diseases"

4 Methodology

4.1 Neural network architecture

The baseline architecture used in this paper is the architecture described in Pham and Le-Hong (2017) in which the input layer takes in word embeddings and additional syntactic features, and feeds to a 2-layer Bidirectional Long Short Term Memory, and then a Softmax layer to make predictions. This model is relatively less complex than other deep learning models previously built for the same task in Vietnamese like Pham and Le-Hong (2017).

Based on this baseline architecture and the goal of using a simple deep learning model, my system consists of a general input layer, whose details will be discussed in the following section, one Bidirectional Long Short Term Memory layer, and a TensorFlow's TimeDistributed layer to produce the final sequence predictions. The detailed architecture of the model is visualized in Section 5.

4.2 Data preprocessing

The words in the sentences are first encoded into numerical values using a pre-defined vocabulary dictionary obtained over all words present in the dataset. After that, the encoded sentences undergo padding so that they have the same length. Two thresholds chosen for maximum sentence length are 70 words (for word-level representation) and 90 syllables (for syllable-level representation), since an inspection into the distribution of sentence lengths shows that these thresholds effectively accounts for a large majority of the dataset. Sentences under these thresholds are padded at the end, while sentences above the thresholds are cut off.

4.3 Feature engineering

4.3.1 Word embeddings

Word embeddings are a form of representing words using high dimensional vectors such that words with similar meanings tend to have similar representations in the vector space. My system utilizes the package of Vietnamese word vectors in Facebook's fastText library which are pre-trained on Common Crawl and Wikipedia pages (Grave et al., 2018). The vectors are in dimension 300 and can be adapted to a lower dimension.

To obtain the vectors, the fastText library needs to be used with the Vietnamese word vectors package (7GB) available. I used the vocabulary dictionary obtained over the entire dataset as mentioned in the previous section to create in advance an embedding matrix in which row i contains the word vector for the word with index i in the dictionary. An extra zero vector is added to the end of the matrix, representing the padded positions in a sentence. This embedding matrix is used in the embedding layer in the neural network to transform arrays of (encoded) words in the input layer into arrays of word vectors. The same method is used for syllable-level data to create a second embedding matrix. Since multi-syllabic words in Vietnamese are comprised of several syllables which are highly capable of standing on their own as separate meaningful words, it is possible to obtain word vectors for these syllables as well.

4.3.2 Part-of-speech tags

To maintain context, a sentence is fed to a Vietnamese part-of-speech tagger as a whole, instead of as individual words. This project examines the use of two Vietnamese NLP toolkits to obtain part-of-speech tags: VnCoreNLP (Vu et al., 2018) and Undertheseа[1]. It is found that for the majority of

[1] github.com/undertheseanlp/undertheseа

the dataset, VnCoreNLP's part-of-speech tagger detects and tags the correct words as they are represented in the named entity dataset, which shows a high level of word segmentation agreement between the toolkit and the named entity data's annotation. Though there are cases of conflicts that are mainly due to language ambiguities, such cases are in the order of 10 and can be manually rectified. Underthesea's part-of-speech tagger, on the other hand, seems to disagree more with the annotated data in terms of word segmentation, producing conflicts in the order of 100, which are more difficult to manually fix. Therefore, VnCoreNLP is chosen for part-of-speech tagging in all experiments.

The part-of-speech tags of the words are converted to numerical values and then turned into sparse vectors of dimension 23 using one-hot encoding, with 23 being the total number of unique tags present in the dataset.

When dealing with syllables, it is not feasible to obtain part-of-speech tags for each syllable individually. Since sentences first undergo word segmentation and then part-of-speech tagging, syllables that make up the same word are grouped together by the algorithm, giving only tags for the words and not the syllables. If syllables are fed to the tagger individually, the absence of context will very likely result in a high margin of errors. This system uses a workaround for this issue: the syllables are obtained from the word-level dataset by manually splitting multi-syllabic words (i.e. words with underscores in the data files) into their smaller syllables, with each syllable inheriting the part-of-speech tag from the higher word.

4.3.3 Manually extracted features

The system extracts a selected number of features from the words, many of which are general word shape features, while some others are more narrowed down features that utilize custom word lists to search for some specific patterns derived from error analysis. Table 2 summarizes the general word shape features used in this system.

The specific, customized features are as follows:

- containsLastName: whether the word/syllable contains a word that belongs to a list of common last names.
- isInJobs: whether the word/syllable appears in a list of common occupations.

Features	Examples
isNotAlNum	Punctuations: , . ’ " etc.
isLower	bệnh_nhân, thành_phố
isAllCap	UBND, THPT
isMixed	iPhone
isTitle	Bệnh_viện, Đa_khoa, Y_tế
isTitleMul	Quảng_Trị, Đà_Nẵng

Table 2: Word shape features used and examples. isTitleMul is used only for words and not syllables, since only words can have consecutive capitalized syllables.

- isNoun: whether the word/syllable has a part-of-speech tag as one of the noun types.
- isPatientId: whether the word/syllable follows the pattern of "BN123" which the model failed to pick up as patient IDs in error analysis.
- isPatientAbbv: whether the word/syllable includes initials of a person's name, usually in the form of capital letters followed by periods.
- isFollowedAge: whether the word/syllable is immediately followed by the word "tuổi" (age), which the model was observed to overlook as age entities.

The features are extracted for unigram, using only the current word/syllable.

All of the features are formed as functions that give boolean values, so a vector (of length 11 for syllable-level and length 12 for word-level) of 0s and 1s are extracted for each word/syllable.

4.4 Evaluation method

Since all input sentences are either padded up or cut off to be exactly 70 words/90 syllables each, the neural network also outputs a sequence of 70/90 predicted named entity tags for each sentence, which might belong to padded positions instead of real words. At the evaluation stage, based on the legitimate words/syllables in the sentences, model predictions are stripped of all labels in padded positions, and the remaining labels are used to calculate precision, recall, and F score. Predictions are aggregated by entity types, for example, all B-DATE and I-DATE predictions are used to evaluate system performance for the DATE entity type.

5 Experiments

Experiments are conducted on a number of models with different inputs to test performances of differ-

ent combinations of features. Figure 1 depicts the general architecture of the system.

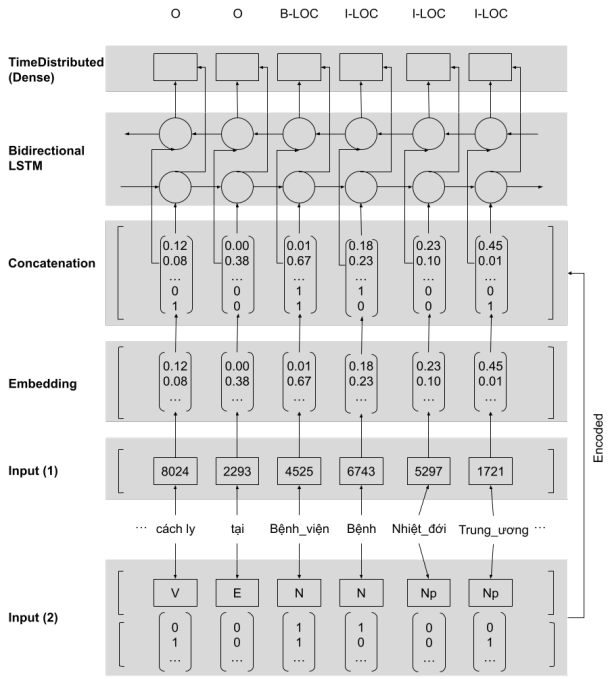

Figure 1: General architecture of the system

Input (1) is the word/syllable input layer. This layer includes words/syllables in sentences that have been padded and encoded, which are then passed to the **Embedding** layer, which is configured with the corresponding embedding matrix to turn each word/syllable index into a word vector. Depending on the combination of features used, the embedding vectors are concatenated with the vectors of part-of-speech tags and additional features (supplied in **Input (2)** if needed) in the **Concatenation** layer. This layer outputs the finalized representation of each word/syllable, which is then passed to the **Bidirectional LSTM** and then the **TimeDistributed** wrapper of Dense layers to produce the final predictions.

The hyper-parameters are maintained in all experiments and are described in Table 3.

Hyper-parameter	Value
BiLSTM hidden units	200
BiLSTM dropout	0.3
Activation function	Softmax
Optimizer	Adam
Learning rate	0.01
Decay	$1e$-6

Table 3: Hyper-parameters used in the model

Two callback functions, `EarlyStopping` and `ReduceLROnPlateau`, are also set up to monitor the loss on the validation set while training the model.

6 Results

The system is trained and tested on word-level representation and syllable-level representation with embedding vectors of dimensions 300 and 100. 100-dimensional vectors are added to, in parts, bridge the balance gap between the dimensions of the embedding vectors and additional features.

Performances of the system with word-level representation are summarized in Table 4, and performances with syllable-level representation are in Table 5.

With both word-level and syllable-level, performance of the system improves with the addition of part-of-speech tags and manually extracted features, and models with 300-dimensional embedding vectors overall achieve better results than models with 100-dimensional embedding vectors. The best performing model is one where all investigated features are combined: embedding vectors of high dimension combined with part-of-speech tags and extracted features.

Though the addition of part-of-speech tags and extracted features improves overall F score of the system by only 1-2%, they have bigger impacts on some specific entity types:

- F score in JOB entity type improves from 57% (only 300-dimensional embeddings) to 63% (embeddings + part-of-speech tags + features).
- F score in NAME entity type improves from 82% (only 300-dimensional embeddings) to 91% (embeddings + part-of-speech tags + features).

The system is also evaluated in its ability to identify the entity boundaries (the spans of the entities) accurately. In this regard, the system shows to improve with additional features.

7 Discussion

7.1 Comparison with previous works

Along with publishing the dataset, the paper by Truong et al. (2021) conducts experiments on the data using the BiLSTM-CNN-CRF model (Ma and Hovy, 2016) and two pre-trained language models

Models	Unweighted			Weighted			Correct boundaries
	Prec.	Recall	F	Prec.	Recall	F	
300D	93.51%	85.19%	88.79%	**95.62%**	90.83%	93.06%	90.59%
300D+POS	**93.53%**	87.10%	89.94%	95.50%	91.68%	93.46%	92.08%
300D+POS+F	93.42%	**88.02%**	**90.41%**	95.23%	**92.24%**	**93.64%**	**92.34%**
100D	91.74%	84.54%	87.66%	94.80%	90.43%	92.46%	90.34%
100D+POS	91.29%	86.81%	88.76%	94.68%	91.08%	92.78%	91.27%
100D+POS+F	92.28%	87.15%	89.46%	94.66%	91.77%	93.13%	92.03%
BiL-CNN-CRF			87.5%			91%	
PhoBERT base			92%			94.2%	
PhoBERT large			93.1%			94.5%	

Table 4: Performances on word-level representation. 300D, 100D = embedding vectors. F = features. The last 3 rows are baseline models from Truong et al. (2021).

Models	Unweighted			Weighted			Correct boundaries
	Prec.	Recall	F	Prec.	Recall	F	
300D	93.07%	85.77%	89%	95.08%	91.76%	93.30%	91.15%
300D+POS	92.62%	86.91%	89.54%	95.12%	91.09%	93.52%	91.55%
300D+POS+F	**93.95%**	**87.67%**	**90.45%**	**95.28%**	**92.27%**	**93.67%**	**92.30%**
100D	92.29%	85.59%	88.51%	94.60%	91.22%	92.79%	90.41%
100D+POS	91.59%	86.24%	88.67%	94.44%	91.34%	92.81%	90.84%
100D+POS+F	93.38%	86.63%	89.65%	94.81%	91.79%	92.24%	91.64%
BiL-CNN-CRF			85.8%			90.6%	
XLM-R base			87.9%			92.5%	
XLM-R large			91.1%			93.8%	

Table 5: Performances on syllable-level representation. 300D, 100D = embedding vectors. F = features. The last 3 rows are baseline models from Truong et al. (2021).

with fine tuning, the multilingual model XLM-R (Conneau et al., 2020) (used for the syllable-level representation), and the monolingual model PhoBERT (Nguyen and Tuan Nguyen, 2020) pre-trained on Vietnamese (used for the word-level representation).

The system shows performances better than the BiLSTM-CNN-CRF model and close to those of fined tuned pre-trained language models XLM-R and PhoBERT.

Table 6 shows the performances by entity type of the best achieving model in this paper in comparison to best baseline models. This paper's model shows a similar tendency to perform weaker in identifying entities of types JOB, NAME, ORGANIZATION, and SYMPTOM&DISEASE.

7.2 Error analysis

Error analysis is conducted on the development set using the model with 100-dimensional word embeddings and part-of-speech tags. This part elaborates on a few highlights from the analysis.

Many entities of either type LOCATION or ORGANIZATION are detected with the correct boundaries, but with mistaken type (i.e. labeled as LOCATION while it is ORGANIZATION and vice versa). This is largely due to context ambiguities, for the same entity, usually a name of a hospital, can belong to either of the two types depending on the context and/or bias of the annotator. Truong et al. (2021) notes that this is a foreseeable type of errors, and that mislabeling an ORGANIZATION entity as a LOCATION can still be useful in contact tracing.

Many entities are detected with the correct types, but the wrong boundaries (missing the leading word, or including one extra word that is not included in the labels). This type of error is typically observed in the SYMPTOM_AND_DISEASE group, where the correct labels focus more on the specific symptom/disease while the model sometimes includes an extra leading word in the

		AGE	DAT	GEN	JOB	LOC	NAM	ORG	ID	SYM	TRA	U. F	W. F
Word	*	96	99	93	63	95	91	88	98	87	94	90.41	93.64
	P-B	96.7	98.9	96.8	79.1	94	94.4	87.6	98	88.5	96.7	93.1	94.5
Syl.	*	96	99	93	65	96	86	89	98	87	95	90.45	93.67
	X-R	96.2	98.7	95.8	69.2	94.3	93.3	85.3	98.2	85.4	94.3	91.1	93.8

Table 6: Performances of this system compared to best baseline systems in different entity types when tested on word-level and syllable-level representations. * = system using 300-dimensional embedding vectors, POS tags, and additional features. P-B = PhoBERT large, X-R = XLM-R large. U. F = unweighted F score. W. F = weighted F score. Unit: %.

phrase.

The model's recall in identifying entities of type JOB is quite poor compared to precision in this type and with recall metrics in other entity types, with the best achieving model scoring only 53% in recall for JOB entities. Through manual inspection, it is observed that the JOB entities identified by the model are mostly legitimate occupations. However, due to annotation guidelines in the original dataset, eligible JOB entities are those that represent occupations of some specific PATIENT_ID or NAME entities. In many sentences, a lot of occupations are mentioned but since they are not directly linked to specific people, they do not bare the labels for JOB entities, and thus contributing to the low recall.

8 Conclusion

This paper presents a named entity recognition system for the Vietnamese language that focuses on the specific domain of COVID-19 news articles. The system utilizes a neural network architecture with Bidirectional Long Short Term Memory at its core and a number of features on the word-level and syllable-level representations. The system performs better when the words or syllables are represented using higher dimensional embedding vectors. It is also shown that adding part-of-speech tags and a number of manually extracted features, including word shape features and custom word lists, contributes to a 1-2% improvement in the overall average F score, and higher improvements (6-9% increase) in specific entity types. The best performing system uses a combination of 300-dimensional embedding vectors, part-of-speech tags, and additional features and achieves unweighted F scores of 90.41% (in the word-level settings) and 90.45% (in the syllable-level settings). The system results show to be higher than a more complicated baseline model, and lower than but close to results of pre-trained language models that have undergone fine tuning.

The system is among the first to focus on Vietnamese named entity recognition with the specific scope of COVID-19 news articles. I hope that the work will serve as an early research into the effectiveness of using a simple neural network architecture with combined specific features for named entity recognition when the domain subject is narrowed and the language is one with lower resources.

9 Future work

In the future, the system can be extended to make use of higher n-gram models. The current system is unigram since a vector representation considers only the current word/syllable with little information extracted using a context window (except for one entry on the additional feature vector which inspects whether a word is followed by "tuổi"). It is also useful to conduct experiments using different dimensions for embedding vectors to further study the impacts of other features when there is a better balance between the length of the embedding vectors and the length of the additional features. Additionally, it will be helpful to be able to obtain domain-specific word embeddings in order to reflect the relationships and similarities between words/syllables more accurately.

The system can be implemented in a number of downstream tasks, including automatic contact tracing using online news articles or relationship extraction for clustering patients by locations, among others, to contribute to the fight against the COVID-19 pandemic.

References

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised

cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC 2018)*.

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)*, pages 1064–1074, Berlin, Germany. Association for Computational Linguistics.

Pham Quang Nhat Minh. 2018. A feature-rich vietnamese named-entity recognition model. *CoRR*, abs/1803.04375.

Dat Quoc Nguyen and Anh Tuan Nguyen. 2020. PhoBERT: Pre-trained language models for Vietnamese. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1037–1042, Online. Association for Computational Linguistics.

Thai-Hoang Pham and Phuong Le-Hong. 2017. End-to-end recurrent neural network models for vietnamese named entity recognition: Word-level vs. character-level. *CoRR*, abs/1705.04044.

Thai-Hoang Pham and Phuong Le-Hong. 2017. The importance of automatic syntactic features in Vietnamese named entity recognition. In *Proceedings of the 31st Pacific Asia Conference on Language, Information and Computation*, pages 97–103. The National University (Phillippines).

Thinh Hung Truong, Mai Hoang Dao, and Dat Quoc Nguyen. 2021. COVID-19 named entity recognition for Vietnamese. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2146–2153, Online. Association for Computational Linguistics.

Thanh Vu, Dat Quoc Nguyen, Dai Quoc Nguyen, Mark Dras, and Mark Johnson. 2018. VnCoreNLP: A Vietnamese natural language processing toolkit. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 56–60, New Orleans, Louisiana. Association for Computational Linguistics.