Detecting spelling errors in Vietnamese administrative document using large language models

Huan T. Phung¹*, Nghia V. Luong²

¹Thai Nguyen University of Information and Communication Technology, Thai Nguyen, Vietnam
²Pham Van Dong University, Quang Ngai, Vietnam
*Corresponding author: pthuan@ictu.edu.vn

ARTICLE INFO

ABSTRACT

DOI:10.46223/HCMCOUJS.tech.en.14.1.3141.2024

In the context of the emergence of more and more administrative documents, the need to ensure accuracy and improve the quality of these documents becomes increasingly important. This research focuses on applying advanced language models to detect spelling errors in administrative documents. Specifically, in this study, a new method using a language model based on the Transformers architecture is proposed to automatically detect and correct common spelling errors in administrative documents. This method combines the model’s ability to understand context and grammar to identify words or phrases that are likely to be misspelled. The proposed method is tested on a dataset containing real administrative documents, and the experimental results show that the proposed model is capable of detecting spelling errors with significant performance, helping to improve accuracy and improve the quality of administrative documents. This research not only contributes to improving the quality of administrative documents but also opens up new research directions in applying language models to issues related to natural language processing in the field of administration.

1. Introduction

In today’s era of digitalization and information technology, the role of machine learning has spread widely and deeply into many different fields, from business to medicine, from space science to public administration (Anastasopoulos & Whitford, 2019). One of the important applications of machine learning is in natural language processing, opening up new possibilities in optimizing and improving workflows, especially for administrative documents (Daniel, Julian, & Jugal, 2020). Administrative documents include documents and papers used in the process of managing and performing administrative work of organizations and agencies (Nguyen & Nguyen, 2021). In a context where the drafting and processing of administrative documents plays an important role in maintaining the accuracy and professionalism of information, detecting spelling errors is an extremely big and unavoidable challenge can be ignored. Spelling errors can cause misunderstandings, reduce the seriousness and credibility of the document, and even affect administrative decisions and management processes.

In this article, we research and propose the application of large language models to detect spelling errors in drafting administrative documents. This opens up new perspectives in improving the efficiency and accuracy of the document drafting process, contributing to creating a more efficient and professional working environment in administrative agencies.
This article not only focuses on presenting research results and practical applications of large language models in detecting spelling errors but also suggests potential future development directions. The continuous development of technology and the capabilities of machine learning are reshaping the way traditional problems are approached and solved while opening new doors for sustainable development in the field of public administration. Ultimately, the contributions of the study include:

- Proposing a Bidirectional Encoder Representations from Transformers (BERT) large language model trained on administrative text data: a method for detecting and correcting spelling errors in Vietnamese.

- A publicly available dataset of spelling errors, this dataset is used to evaluate the proposed method based on different approaches.

2. Theoretical basis

Spelling error correction is an essential part of the field of natural language processing. In the context of the Vietnamese language, various techniques have been suggested for addressing this issue. Past methodologies can be broadly categorized into two groups: one relies on conventional statistical language models, while the other employs machine learning techniques.

In 2008, in Nguyen, Ngo, Phan, Dinh, and Huynh’s (2008) research, a method to fix spelling errors was proposed, including two stages: error identification and repair implementation. In the recognition phase, this method used dictionaries such as syllable classification dictionaries, proper name recording dictionaries, and abbreviation dictionaries to identify incorrect words. For context-related errors, this method combines consecutive syllable pairs (bi-gram) with syllable pairs and grammatical classification (POS Bi-gram), combined with several methods. predictions like concurrent probabilities, and unchecked word lists. During the correction phase, suggestions are generated using common rules about spelling errors along with sets of confusing words based on the Hoang Phe dictionary (Hoang, 2009). Then, the suggestions were ranked using algorithms such as Minimum Edit Distance, SoundEx, and some rules by measuring the word length and frequency of these words in the text.

In separate research (Nguyen, Dang, Nguyen, & Le, 2015), an N-gram language model was created to rectify Vietnamese spelling errors. A substantial unlabeled dataset was utilized to grasp the contextual use of syllables. To be precise, the N-gram score was computed for each syllable in the candidate set, taking into account their frequency of occurrence. The model generates a confusion mixture set for each syllable based on the edit distance and the chosen confusion mixture set. An error is identified if any syllable in the candidate set obtains a higher N-gram score than the present one. In another N-gram method for Vietnamese OCR post-processing, the authors Nguyen, Le, and Zelinka (2019) have developed a method to generate and score error correction candidates based on linguistic features language. Following the detection step, a candidate set is formed for each syllable error through the application of insertion, deletion, and substitution operators. These studies illustrated that, despite its age, the N-gram model remains a robust solution for addressing natural language processing challenges, delivering remarkable performance.

In a separate investigation (Nguyen, Nguyen, & Vaclav, 2015), a tool was developed for standardizing Vietnamese Twitter posts. This study introduced a two-step spelling error correction model, encompassing error identification and correction. The author constructed a Vietnamese word dictionary, designating a word as erroneous if it was not found in the dictionary. For each error, the author applied a set of syllable rules for normalization. Subsequently, the results were
subjected to the next phase, where the dice coefficient was used to calculate the similarity with words in the dictionary, aiding in the identification of words with the greatest resemblance. Furthermore, the 2-step model above represents practicality, as many systems want to maximize error detection and suggest possible cases without automatically correcting errors. Therefore, with the recent development of deep learning, pre-trained neural network-based language models can be applied to detect and correct spelling errors, yielding good results due to their ability to represent better language performance.

In a recent investigation (Tran, Dinh, Phan, & Nguyen, 2021), a Hierarchical Transformer model was introduced for the correction of spelling errors in Vietnamese. This model utilizes two Transformer encoders: one at the character level and the other at the word level, enabling the representation of each word from both character and word perspectives. The word representation is subsequently subjected to a detection classifier responsible for identifying misspelled words and, finally, a correction classifier designed to propose alternative replacements for the identified misspelled words. Each classifier is composed of a pair of fully connected layers, with a softmax activation function applied at the final layer. However, the seq2seq model has one biggest drawback: its performance is not as good as the word classification model. Furthermore, errors such as word sticking and word permutation errors cannot be handled by the model.

BERT (Devlin, Chang, Lee, & Tautanova, 2019), which stands for Bidirectional Encoder Representations from Transformers, represents a novel architectural approach within the Language Representation domain. Diverging from its predecessors, BERT is specifically engineered to generate vectors that encapsulate contextual information from both the left and right dimensions of the text. Consequently, the representative vectors produced by the BERT model, when fine-tuned with additional output layers, have given rise to significantly enhanced architectures for a multitude of natural language processing tasks. These improvements span areas like Question Answering and Language Inference, among others, while preserving many aspects of the pre-existing architectures. This class of language modeling representation problems has been demonstrated to be highly effective in enhancing various tasks within the field of natural language processing, encompassing sentence-level challenges such as Natural Language Inference and Paraphrasing, which involve predicting relationships between sentences by analyzing them holistically, as well as word-level tasks like Named Entity Recognition (NER) and Question Answering, which necessitates returning precise responses to questions in word format, and so forth.

Methods typically apply grammatical rules and laws to detect and correct spelling errors. This can lead to omissions or errors where appropriate grammatical rules cannot be applied in all situations. On the other hand, previous studies were often conducted on English data, but rarely on Vietnamese data. This article studies recent deep learning models, especially the pre-trained BERT model, that can work effectively for the Vietnamese language.

3. Methodology

3.1. State the problem

The problem of detecting spelling errors in Vietnamese can be stated as follows: Given a string of \( n \) syllables \( X = (x_1, x_2, \ldots, x_n) \) and a string \( n \) of labels \( Y = (y_1, y_2, \ldots, y_n) \), \( y_i \in \{0,1\} \), where \( y_i \) is the label of the syllable \( x_i \). Error detection can be viewed as a sequential labeling problem, taking as input a string of \( n \) syllables \( X = (x_1, x_2, \ldots, x_n) \) and assigning each word a label of 0 for correct and 1 for misspelled words.
3.2. Model

BERT, as elucidated in the research (Ashish et al., 2017), is a bidirectional encoder representation derived from Transformers, which acquires sentence representations through the concurrent learning of two distinct tasks: language modeling utilizing masks and predicting the next sentence. BERT models can undergo fine-tuning at both the sentence and word levels. BERT’s architecture is deeply layered, comprising 12 layers, each with hidden layers of size 768 and 12 attention mechanisms. The model initiates at the word embedding layer. Within the 12-layer structure, attention is computed based on the word representation from the preceding layer, resulting in the creation of a fresh intermediate representation. Consequently, a token will possess 12 intermediate representations, all of identical size.

\[ \text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_n)W^0 \]  
\[ \text{Head}_i = \text{Attention}(QW^i, KW^i, VW^i) \]  
\[ \text{FFN}(X) = \max(0, XW_1 + b_1)W_2 + b_2 \]

Within the context of the masked language modeling task, 15% of the tokens were arbitrarily selected to establish a two-dimensional pre-trained language model. To ensure alignment between pre-training and fine-tuning phases, within this 15%, tokens were substituted as follows: 80% of the time with [MASK], 10% of the time with a randomly selected token from the vocabulary, and the remaining 10% of the time, they remained unchanged.

In this article, we apply the BERT model to the sequential labeling problem to detect spelling errors in the Vietnamese language. The experimental process consists of two phases, including pre-training BERT on a dataset of unlabeled administrative documents and fine-tuning the spelling error detection problem on a set of labeled administrative documents. BERT pre-training is done by masking some tokens and replacing them with the word [MASK]. The main task of this process is to train the model to predict the hidden word based on the words that come before and after it. During the process of refining the BERT model, two special classification tokens ([CLS]) and a separation token ([SEP]) will be added to the input of the BERT model as follows: \( \text{Bert - Input(text)} = [CLS] \text{text}[SEP] \) where \text{text} is a piece of administrative text included for checking spell check.

Figure 1. BERT model for spelling error detection problem
The contextual representation of each word generated by the BERT encoder serves as the basis for determining whether a token is erroneous or not. Subsequently, the output representations are channeled into a classifier for the detection task. These classifiers are structured with a fully connected layer featuring an activation function. Following their journey through the detection classifier, each token within the sequence is categorized as either accurate or erroneous. The training process for the model employs the Cross-Entropy function as its objective function.

\[
L = -\frac{1}{N} \sum_{i=0}^{N-1} \sum_{j \in [t_i]} y_{ij} \log p_{ij}
\]

Where \( y_{ij} \), \( p_{ij} \) represents the true label and the corresponding predicted probability.

4. Experiments and results

4.1. Evaluation data

The VBPL (Văn Bán Pháp Luật) data set for the problem is built to include 100,000 sentences collected from administrative documents on the website https://vbpl.vn (National database of legal documents, n.d.). This data has no typographical errors. Data containing spelling errors is automatically generated based on some common errors encountered when editing Vietnamese documents.

In this article, we propose a spelling error detection model with the following 03 types of errors:

**MISSING_SPACE error**: This error occurs when two consecutive words are stuck together.

For example: văn bán – văn bán. Another example of this error is the spelling error – văn bán – vawnbanr, this error occurs very often when typing.

**EDITS error**: This error occurs when a character in a word is changed, inserted, deleted, replaced and transposed, where unit is defined as either a vowel (excluding accents), such as a, e, i, o, u, y, or a consonant (encompassing consonants with 02 or 03 letters), like b, c, ngh, gi, and so on.

For example, trùng lặp – chùng lặp, văn bán – văn bán, văn bán – vawn banr

**MISSING_ACCENT error**: This type of error occurs when unaccenting a vowel.

For example: trùng lặp – trùng lặp, văn bán – văn bán.

From the initial benchmark dataset, we introduced errors through the following methods:

The **MISSING_SPACE error** is induced by eliminating spaces between randomly adjacent words and transforming one or more of them into Telex form.

**EDITS error** is generated by inserting, deleting, or replacing a character or transposing/permuting two adjacent characters.

The **MISSING_ACCENT error** is generated by removing all vowel accents.

In this method, the annotation data is divided into three separate parts: training, development, and testing (Table 1).
Table 1
Statistics of the VBPL dataset

<table>
<thead>
<tr>
<th>Data</th>
<th>Number of sentences</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>70,000</td>
</tr>
<tr>
<td>Develop</td>
<td>12,000</td>
</tr>
<tr>
<td>Test</td>
<td>18,000</td>
</tr>
<tr>
<td>Average length (syllable)</td>
<td>32</td>
</tr>
<tr>
<td>Vocabulary size (syllables)</td>
<td>24,200</td>
</tr>
</tbody>
</table>

Also, use a large corpus for BERT pre-training (Table 2).

Table 2
Statistics on unlabeled corpus collected from National database of legal documents (n.d.)

<table>
<thead>
<tr>
<th>Number of administrative documents</th>
<th>Data size (syllable)</th>
<th>Vocabulary size (syllables)</th>
</tr>
</thead>
<tbody>
<tr>
<td>14,079</td>
<td>70.6 million</td>
<td>489,960</td>
</tr>
</tbody>
</table>

4.2. Evaluation methods

We use word-level F1 measures to evaluate the system’s ability to detect errors. We use F1 instead of Accuracy because the number of spelling errors in a text is very small, so the F1 measure achieves a balance between two classes: words with errors and correct words.

Specifically, we used the following calculation:

\[
P = \frac{\text{Number of True Positives}}{\text{Number of True Positives} + \text{Number of False Positives}}
\]  

\[
R = \frac{\text{Number of True Positives}}{\text{Number of True Positives} + \text{Number of False Negatives}}
\]  

\[
F1 = \frac{2 \times P \times R}{P + R}
\]

4.3. Experimental results

BERT model installations are done from the NlpHUST/bert-base-vn model. First, we pre-trained BERT on unlabeled Vietnamese administrative text data (Table 2) and used the NlpHUST/bert-base-vn model to initialize the weights for the model at the beginning of training.

We collected 15,400 administrative documents from the website https://vbpl.vn (National database of legal documents, n.d.). Then set parameters such as maximum length of 512, batch size of 16 and learning rate of 2e-5 with 40,000 steps. The error function on the dev set decreased from 0.7 to 0.45 (Figure 2). The proposed model is called the BERT4Legal model.
After pre-training, our model was fine-tuned for spelling error detection using the VBPL dataset. Throughout the training phase, the method under consideration utilizes the Adam optimizer and the Cross-Entropy error function to train the neural network model incorporating the BERT architecture. Experiments with the BERT architecture with hyperparameters are shown in Table 3. The hyperparameters were adjusted to V on the development kit. The average training time for BERT models is 02 hours with 70 thousand training samples. We trained all BERT fine-tuning models with 05 iterations. The best model is evaluated based on calculating the F1 score on the dev set.

<table>
<thead>
<tr>
<th>Model</th>
<th>Max-length</th>
<th>Learning rate</th>
<th>Batch size</th>
</tr>
</thead>
<tbody>
<tr>
<td>FPTAI/vibert-base-cased</td>
<td>512</td>
<td>2e-5</td>
<td>16</td>
</tr>
<tr>
<td>NlpHUST/bert-base-vn</td>
<td>512</td>
<td>2e-5</td>
<td>16</td>
</tr>
<tr>
<td>BERT4Legal</td>
<td>512</td>
<td>2e-5</td>
<td>16</td>
</tr>
</tbody>
</table>

We also compare our pre-trained model in the administrative text domain with pre-trained BERT models in other popular domains:

- FPTAI/vibert-base-cased: Trained on 10GB of Vietnamese news data using vocabulary edited from BERT-multilingual (a BERT model for multilingual published by Google).
- NlpHUST/bert-base-vn: Trained on 100GB of Vietnamese Oscar data is a data set collected from websites on the Internet covering many fields (news, entertainment, society, academics, technology, etc.), using 62,000 vocabulary words based on Vietnamese syllables.

The LSTM model we employ, as per (Rivera-Acosta et al., 2021), is structured as a binary classification framework comprising a pair of hidden Bi-LSTM layers. This particular model accepts a word as input and produces an output signifying whether the word is correctly spelled.
The LSTM’s recurrent connections enable it to analyze both preceding and succeeding segments of characters or words, enhancing its ability to identify spelling errors. The model takes the complete word as input, which is subsequently segmented into tokens using the BPE Tokenizer.

Our models were deployed using Torch and all testing was performed on an Nvidia RTX 3090 24GB GPU.

### Table 4

<table>
<thead>
<tr>
<th>Model</th>
<th>F1 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTM</td>
<td>88.61</td>
</tr>
<tr>
<td>FPTAI/vibert-base-cased</td>
<td>92.65</td>
</tr>
<tr>
<td>NlpHUST/bert-base-vn</td>
<td>96.83</td>
</tr>
<tr>
<td>BERT4Legal</td>
<td>97.98</td>
</tr>
</tbody>
</table>

Table 4 provides comprehensive experimental outcomes for the VBPL dataset, categorized into three sections: neural networks with LSTM encoders, BERT pre-trained on various corpora, and fundamental probability models. For all these models, the input unit utilized is syllables. Table 4 also shows that the deep learning method outperforms the baseline models, especially with BERT when pre-trained on domain-specific data. BERT4Legal achieved the highest performance (97.98% in F1). These experiments demonstrate that when the source domain used in the training model matches the target domain, it can significantly enhance the final results. Vocabulary in administrative documents includes many words used to refer to abbreviated decrees and bases such as 1268/QD-UBND (Ủy ban nhân dân tỉnh Bạc Liêu, 2023), 138/2020/ND-CP (Chính phủ, 2020), etc. Pre-training of the BERT model in the field administration before adjusting for a specific task gives the model better knowledge in the field.

Below is an example of the test models’ ability to detect spelling errors. Words are bolded to indicate that they are words that the model predicts are errors.

**Error sentence:** Lập đễ án phân loại và công nhận thị trình Mộ Đúc, huyê MơĐưêclà đỗ thị loay V vào năm 2018.

**Results of detecting spelling errors:**

**LSTM:** Lập đễ án phân loại và công nhânhtrên Mộ Đúc, huyê MơĐưêclà đỗ thị loay V vào năm 2018.

**FPTAI/vibert-base-cased:** Lập đễ án phân loại và công nhânhtrên Mộ Đúc, huyê MơĐưêclà đỗ thị loay V vào năm 2018.

**NlpHUST/bert-base-vn:** Lập đễ án phân loại và công nhânhtrên Mộ Đúc, huyê MơĐưêclà đỗ thị loay V vào năm 2018.

**BERT4Legal:** Lập đễ án phân loại và công nhânhtrên Mộ Đúc, huyê MơĐưêclà đỗ thị loay V vào năm 2018.

Through the above example, we can see that models using BERT are much better at detecting errors than models using LSTM because they are pre-trained on a large amount of data. With models FPTAI/vibert-base-cased and NlpHUST/bert-base-vn all identified “loa” and “V” as two misspelled words (the correct word is type V). With our proposed model BERT4Legal, due
to pre-training on the administrative document data set, the word “V” has appeared many times in similar contexts, so the model can identify the word “V” as the correct spelling.

5. Conclusions

In this article, we have proposed a method to detect spelling errors in the Vietnamese language using the BERT language model. The proposed method, called BERT4Legal, is experimentally comparing the performance with several related methods. Specifically, we performed a series of experiments between the LSTM method and BERT-based methods to detect spelling errors on a Vietnamese dataset of administrative documents. Experimental results show that the BERT4Legal method has the best performance compared to related methods. In particular, the administrative document dataset generated when training the BERT model can be useful for related research.

With this research, we hope to promote research directions related to the field of administrative document application in Vietnam. In the future, we will study automatic spelling correction based on large language models trained specifically for administrative text data.

References


