Journal of Computer Science and Cybernetics, V.41, N.3 (2025), 305–321
DOI no. 10.15625/1813-9663/20618

PRE-TRAINED LANGUAGE MODELS FINE-TUNED WITH SVM FOR LEGAL TEXTUAL ENTAILMENT RECOGNITION

QUAN VAN NGUYEN[1,3], ANH TRONG NGUYEN[2,3], HUY QUANG PHAM[1,3], KIET VAN NGUYEN[1,3,*]

[1]*Faculty of Information Science and Engineering, University of Information Technology, Quarter 6, Linh Trung Ward, Thu Duc City, Ho Chi Minh City, Viet Nam*
[2]*Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, 268 Ly Thuong Kiet Street, Dien Hong Ward, Ho Chi Minh City, Viet Nam*
[3]*Vietnam National University, Quarter 34, Linh Xuan Ward, Ho Chi Minh City, Vietnam*

Abstract. The breakthroughs in Natural Language Processing (NLP) are not only a crucial step in technological evolution but also deliver significant benefits across various fields demanding high intelligence and precision. One of the notable NLP applications is in the analysis and processing of legal texts. Capitalizing on this trend, the 10th Workshop on Vietnamese Language and Speech Processing (VLSP) 2023 hosted a new challenge: Legal Textual Entailment Recognition (RTE). The task involves determining whether a given statement is logically entailed by the relevant legal passage. Our proposed method leverages a novel layer based on Support Vector Machine (SVM) kernel formulations, effectively capturing nuanced relationships in the input data. Additionally, it capitalizes on the advantages of the Natural Language Inference (NLI) datasets which are very close to Textual Entailment Recognition (RTE) for enhancing performance and generalization. Our approach not only yielded accurate results but also demonstrated efficiency in the use of data resources, helping our A3N1 team achieve notable accuracy, with a score of 0.7194 on the test set, and ranking third on the leaderboard.

Keywords. Legal textual entailment recognition, support vector machine (SVM), transformer.

1. INTRODUCTION

Recognizing textual entailment (RTE) in English is attracting significant attention. Since the early years of the 21st century, RTE Challenge [1] has been held seven times with remarkable results. However, it seems that little research has focused on RTE for Vietnamese. Recently, datasets and research directions on natural language inference for Vietnamese [2, 3, 4] have appeared, opening up new opportunities in the RTE field.

RTE is a fundamental task in natural language understanding (NLU). The task is to recognize whether the meaning of one sentence could be inferred from the meaning of another given text [1]. The goal is to assess whether the meaning of the hypothesis can be logically inferred (entailed) from the information presented in the given passage. In simpler terms,

*Corresponding author.
E-mail addresses: 21521333@gm.uit.edu.vn (Q.V. Nguyen); anh.nguyence0912@hcmut.edu.vn (A.T. Nguyen); 21522163@gm.uit.edu.vn (H.Q. Pham); kietnv@uit.edu.vn (K.V. Nguyen).

RTE evaluates whether the content of one text can be concluded or deduced from another text. Vietnamese has multiple meanings, leading to ambiguity in spoken or written text, especially in the legal system; this can lead to uncertainty about the meaning and scope of regulations. This ambiguity can pose challenges in tasks such as answering questions, summarizing text, or extracting information. These are the issues that NLU is currently facing. Improving RTE is an important way to address the challenges of NLU.

Since the introduction of the Transformer architecture by Vaswani et al. (2017) [5], several models have leveraged its encoder component to learn contextual word representations. Typical examples include BERT [6], XLM-R [7], DeBERTa [8], T5 [9], and BART [10], all of which have demonstrated superior performance in solving Recognizing Textual Entailment (RTE) tasks compared to earlier approaches. Furthermore, notable works such as PhoBERT [11], BARTpho [12], and ViBERT [13] have also achieved promising results for Vietnamese.

Despite progress in the field, previous research on Vietnamese Recognizing Textual Entailment (RTE) still faces several limitations. One major issue is the limited availability of large-scale, domain-specific datasets-particularly in the legal domain-which creates a gap in the applicability of RTE models to real-world tasks such as legal document processing. In addition, the inherent complexity of the Vietnamese language, characterized by its multiple meanings and ambiguities-especially in legal contexts-poses significant challenges for traditional RTE models, which often struggle to effectively capture such nuances.

Our main contribution is the development of a robust method tailored for the legal RTE task, along with a training strategy designed to adapt effectively to small-scale Vietnamese datasets. The evaluation results, as demonstrated in the VLSP 2023 Legal Textual Entailment Recognition task [14], confirm the effectiveness of our proposed approach.

In this article, we present our approach to the Vietnamese Legal Textual Entailment Recognition[1] [14] task at the 10th Workshop on Vietnamese Language and Speech Processing (VLSP 2023). The structure of this paper is organized as follows: Section 2 reviews related work on RTE. Section 3 analyzes the challenges posed by the dataset. Section 4 describes our proposed method. Section 5 presents the experimental results and analysis. Section 6 presents discussion on how to deal with extremely small datasets. Finally, Section 7 concludes the paper and outlines potential directions for future research.

2. BACKGROUND AND RELATED WORK

Recognizing Textual Entailment (RTE) is a primary challenge in natural language understanding (NLU), where the main goal is to determine whether the meaning conveyed in text can be appropriately inferred. This task is important for various applications, including sentiment analysis, information retrieval, and question-answering systems, as it requires a deep understanding of the semantic relationships between words. The complexity of RTE comes from the nuanced nature of language, where meaning is often implicit, dependent on context, and subject to multiple interpretations. Successful RTE systems must solve these complex problems, relying on advanced computational and language models to discern not only the obvious connections, but also the subtle nuances of embedded reasoning in text.

[1] https://vlsp.org.vn/vlsp2023/eval/lter

2.1. Recognizing textual entailment challenges

Recognizing Textual Entailment (RTE) was first introduced in 2005 [1], with the task framed as a binary classification problem: positive (entailment) and negative (non-entailment). Building on the success of the first challenge, RTE-2 [15] saw increased participation, with 23 groups worldwide (compared to 17 in RTE-1). RTE-3 [16] introduced extended sentence pairs, encouraging participants to focus on discourse-level inferences. The RTE-3 dataset consists of 1,600 sentence pairs manually extracted from various sources, serving different NLP applications such as question answering, information extraction, information retrieval, and summarization.

RTE-4 [17] expanded the task from a two-dimensional to a three-dimensional evaluation model. RTE-5 [18] and RTE-6 [19] further broadened the scope by incorporating datasets where a given hypothesis might be entailed in multiple contextual settings. RTE-7 [18] focused on recognizing entailment in realistic summarization scenarios using corpus-based approaches.

In 2021, the 8th Workshop on Vietnamese Language and Speech Processing (VLSP 2021) officially introduced the first Vietnamese dataset for the RTE task [20], marking a significant milestone in this field. One of the main challenges posed by this dataset is its linguistic diversity, especially the integration of both Vietnamese and English. This bilingual nature introduces a rich and multidimensional landscape that reflects real-world complexities and offers exciting opportunities for research in multilingual and cross-lingual natural language understanding. The fusion of these two languages not only presents technical challenges but also enriches Vietnamese NLP by fostering deeper linguistic insights and pushing the boundaries of existing models.

2.2. Automated legal question answering competition (ALQAC)

In 2021, the Automated Legal Question Answering Competition (ALQAC 2021) [21] introduced a series of compelling challenges, with Task 1, titled "Legal Text Information Retrieval," standing out as a key highlight. This task focused on the automatic answering of legal questions by leveraging legal datasets from Vietnam and Thailand. Additionally, Task 2, known as "Legal Text Entailment Prediction," challenged participants to identify logical relationships between legal documents. Task 3, "Legal Text Question Answering," required the development of systems capable of providing accurate answers to legal queries. The overarching goal of these tasks was to build a system capable of automatically assessing the legality of given statements.

The third edition, ALQAC 2023 [22], aimed to advance the state of the art in processing legal texts in low-resource languages, particularly Vietnamese. It addressed these challenges through two primary tasks: "Legal Document Retrieval" (Task 1) and "Legal Question Answering" (Task 2). Task 1 required participants to develop systems that could retrieve relevant legal articles in response to specific legal questions, while Task 2 focused on directly answering legal questions. These tasks were designed to push the boundaries of automated legal systems by targeting the unique challenges presented by Vietnamese legal texts, which remain underrepresented in natural language processing research.

The results of the ALQAC competitions over the years have provided valuable insights into the strengths and limitations of current methods for processing legal texts in under-

resourced languages. Moreover, ALQAC has demonstrated that while substantial progress has been made in automated legal systems, significant work remains to achieve high accuracy and generalizability in real-world legal applications.

2.3. Recognizing textual entailment methods

A study by Putra et al. [23] outlined key stages for addressing the Recognizing Textual Entailment (RTE) task, namely: Premise (P) and Hypothesis (H) Feature Extraction, and Textual Entailment Classification. The first stage involves four distinct approaches to feature extraction: lexical-based, syntax-based, semantic-based, and hybrid-based. By explicitly defining P as Premise and H as Hypothesis, the authors aim to enhance reader comprehension, particularly for those unfamiliar with the terminology. Meanwhile, the classification stage offers flexibility, allowing the use of both traditional machine learning and modern deep learning techniques.

Most approaches in this domain primarily rely on lexical-based strategies—especially word-overlapping methods-combined with machine learning algorithms to recognize entailment relations [1]. In a notable alternative, Malakasiotis and Androutsopoulos [24] proposed a method that integrates lexical string matching and shallow syntactic features with Support Vector Machines (SVM) [25]. Similarly, Castillo and Alemany [26] introduced an SVM-based approach using four distance-based features, including edit distance, WordNet distance, and the longest common substring between texts.

In recent years, the field has benefited significantly from the emergence of large-scale datasets focused on Natural Language Inference (NLI), and more specifically, RTE. These datasets, rich with inference scenarios, provide a robust foundation for applying deep learning techniques to model the relationships between premises and hypotheses. As a result, various deep learning-based approaches-such as Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), BERT [6], and transfer learning-have been employed to improve performance in RTE tasks. These models often leverage lexical or semantic features to better capture the complex nuances of textual entailment.

3. DATASET

We encountered a challenge with the dataset launched by Tran et al. [14], which is that the number of data samples is small, to be exact 76 samples in the training set and 139 samples in the test set. This dataset is divided into two main parts: legal passages and statements, located in both the training set and the test set. Each legal passage contains several sentences, while each statement contains only one or two sentences. The task is to assign a Yes/No label to each statement, based on the information available in the legal passages.

3.1. Legal passages

The dataset encompasses 18 laws, spanning a diverse range of legal subjects, with a collective total of 2,215 applicable legal provisions (see Figure 1. Notably, the “Civil Code 2015” stands out with the highest number of passages, reaching 689, indicating its extensive coverage and complexity. On the contrary, the “Law on Access to Information 2016” contains

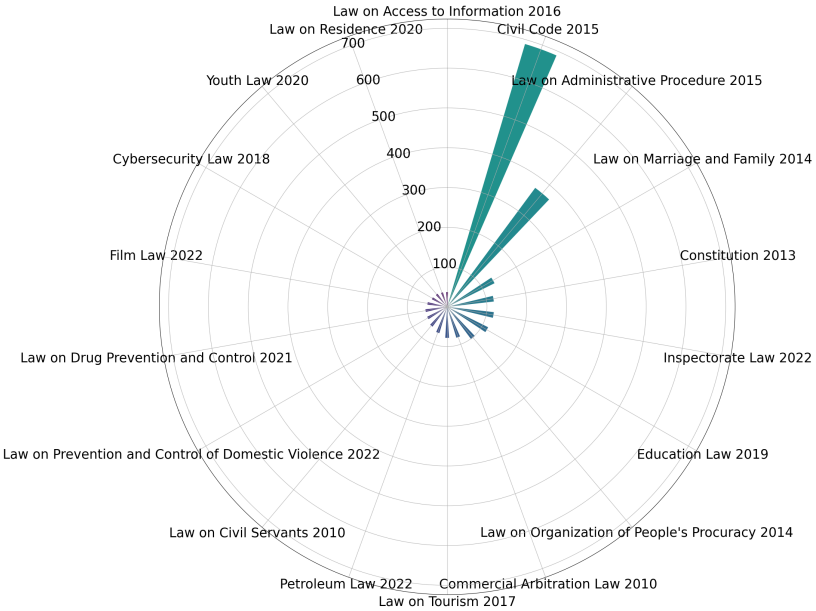

Figure 1: Number of passages in different laws

the fewest passages, including only 37. This considerable variation in the number of legal provisions among the laws underscores the breadth and depth of legal content within the dataset, reflecting the inherent intricacies and nuances in legal frameworks.

In this dataset, the passages exhibit a wide spectrum of sentence counts, spanning from as few as 2 to as many as 30 sentences per passage. This diversity in passage lengths is visually represented in Figure 3, accentuating the range of textual sizes present. While the majority of passages are concise and contain fewer than 300 words, there are notable exceptions. Specifically, a few passages extend to approximately 800 words, potentially indicative of detailed or comprehensive legal content within the respective law documents. This variance in passage lengths not only underscores the richness of the dataset but also suggests varying levels of intricacy and depth across different legal provisions.

3.2. Statements

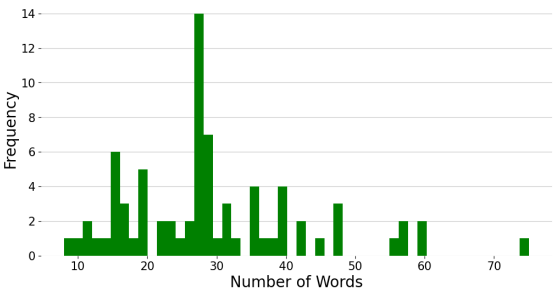

Figure 2: Distribution of statement length

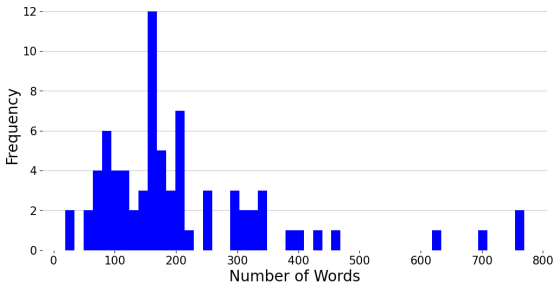

Figure 3: Distribution of passage length

In the given dataset, each statement is almost a single sentence. The variety in sentence length is illustrated in Figure 2, with the majority of sentences ranging from 20 to 40 words in length. However, it is worth noting that there are still some sentences with longer lengths, even up to 70 words, creating a significant diversity and flexibility in the linguistic structure of the data set. The variety of sentence lengths not only helps stimulate linguistic diversity but also provides flexibility to the overall linguistic structure of the data.

3.3. The relationship between RTE and NLI task

Recognizing Textual Entailment (RTE) and Natural Language Inference (NLI) are two concepts that have strong similarities in the field of natural language processing, although they have certain differences. Both aim to identify relationships between two passages, with the primary goal of determining whether a hypothesis can be inferred from a premise. However, while RTE focuses on identifying whether a premise implies a hypothesis (entailment), NLI covers a broader scope, with three main types of relationships: entailment, contradiction, and neutral. However, the methods and models used for RTE and NLI often overlap, with many deep learning models, such as BERT and other transformer models, being applicable to both tasks.

In this study, we also applied NLI datasets such as XNLI [27], and MNLI [28] to fine-tune the model before performing the RTE task, which helps to improve the ability to understand and classify the relationship between sentences. The detailed results are presented in Section 5.

4. METHODOLOGY

This section presents the methodology adopted in our study, providing a comprehensive overview of the preprocessing steps, data partitioning approach, and the innovative model architecture, as illustrated in Figure 4, which incorporates Finding k-sentences module and an SVM layer to significantly improve performance in the task of RTE.

4.1. Data preprocessing and finding K-sentence module

To begin, we segment the passage into individual sentences. This segmentation process is crucial for extracting key semantic components from each passage, as it allows us to retrieve sentences that are most relevant to the given statement. For this retrieval task, we used the Vietnamese SBERT [29], a highly effective sentence transformer pre-trained specifically for Vietnamese. Vietnamese SBERT enables us to identify and extract the top 1, 3, and 5 sentences that are most semantically related to the statement in the legal passage, as illustrated in Figure 4. This retrieval process reduces the input context size, as pre-trained language models typically perform effectively with fewer than 512 tokens. This optimizes computational efficiency by reducing memory load during model training, ultimately enhancing overall system performance.

After that, we divided the data into a training set and a development set. Given the limited dataset of only 76 samples, we experimented with various training-to-evaluation ratios. Through empirical testing, we found that a 50/50 split yielded the most reliable and balanced results. This division allows the model to maintain accuracy and avoid overfitting during

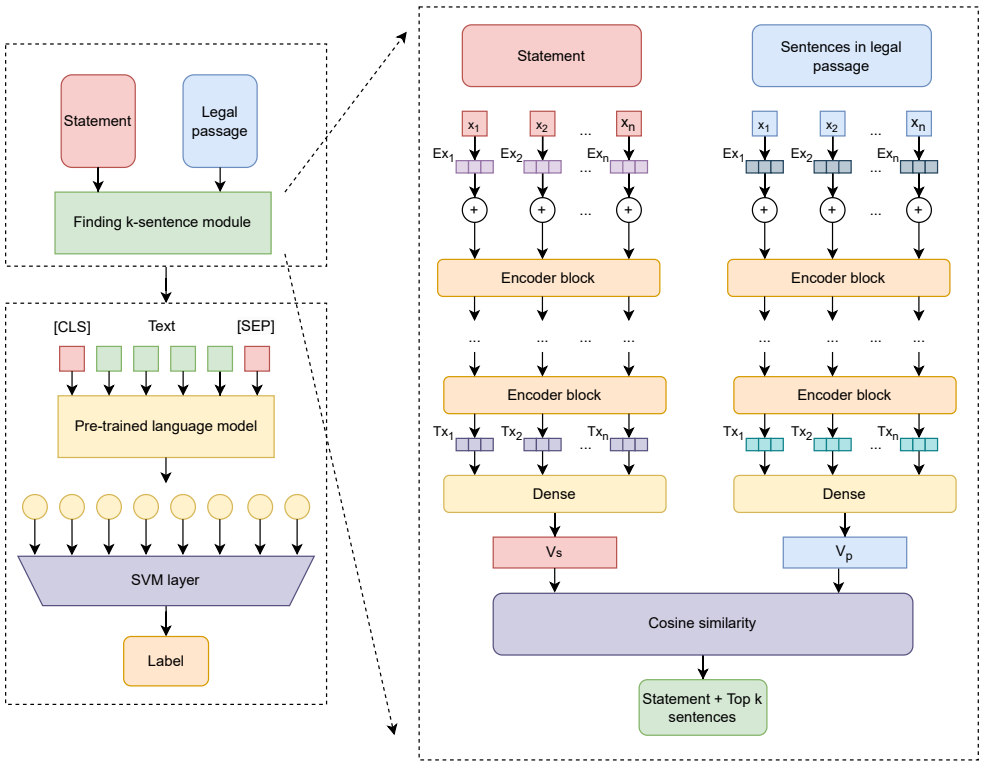

Figure 4: Overview of model structure

evaluation. Before model training, we conducted a series of preprocessing steps, including removing special characters, converting all text to lowercase, and excluding non-essential stopwords. These steps were implemented to streamline and enhance model performance by reducing noise and standardizing the input data.

4.2. Proposed model structure: SVM layer integration

Our proposed model incorporates a novel adaptation for fine-tuning by integrating an SVM kernel-based layer, designed to replace the traditional fully connected (FC) layer typically used for label probability prediction. This SVM-based approach, illustrated in Figure 4, enhances the model's ability to generalize across diverse data samples by leveraging advanced kernel functions. Unlike the conventional FC layer, our SVM kernel layer provides robust classification capabilities and superior adaptability in capturing complex patterns within legal text. The experimental results, presented in later sections, offer a comparative analysis between the traditional FC layer and our proposed SVM layer, highlighting the latter's advantages in terms of accuracy and generalization performance.

Polynomial Kernel Formulation: The polynomial kernel is a fundamental mathematical formulation often utilized in machine learning and kernel-based algorithms, particularly in Support Vector Machine (SVMs). This kernel is formally defined in Equation (1).

$$K(x, x') = (\gamma x^T x' + r)^d, \tag{1}$$

where γ represents a real constant, r is a positive constant, and d is a positive integer. This kernel function transforms data into a higher-dimensional space, calculating the power of the dot product between two feature vectors. The parameter d control the complexity of this mapping function, enabling the model to capture non-linear relationships within the data.

Gaussian Kernel (RBF Kernel) Formulation: Another kernel integrated into our SVM layer is the Gaussian or Radial Basis Function (RBF) kernel, given by Equation (2)

$$K(x, x') = \exp(-\gamma \|x - x'\|_2^2). \tag{2}$$

This kernel is widely utilized due to its ability to map data into a higher-dimensional space using an exponential function based on the Euclidean distance between two feature vectors. The Gaussian kernel's formulation allows it to capture nuanced variations within data points, making it particularly effective for classification tasks that require differentiation across subtle semantic distinctions.

Proposed SVM Layer: Our approach is inspired by both the polynomial and Gaussian kernels in a composite structure, referred to as the "SVM layer". This SVM layer, tightly coupled with the final layer of the pre-trained language model. The formula for our SVM layer is defined in Equation (3)

$$K(x, x_i') = \left(\gamma \|x - x_i'\|_2^2 + r\right)^d. \tag{3}$$

In Equation (3), γ, r, and d are hyperparameters adjustable for optimal performance. The vector $x \in \mathbb{R}^n$ denotes the input feature vector, where n is the input size. The term $x_i' \in \mathbb{R}^n$ represents the weight vector associated with class i, and $i = 1, 2, \ldots, C$ with C being the number of classes. These weight vectors are learnable parameters initialized randomly and optimized during training.

The Euclidean distance between the input feature vector and each class weight vector is calculated as Equation (4)

$$\text{dist}(x, x_i') = \|x - x_i'\|_2. \tag{4}$$

This distance measures the dissimilarity between the input and each class prototype in the feature space and then transforms controlled by the hyperparameters γ, r, and d. For each class i, the outputs are computed by adding a bias term $b \in \mathbb{R}^C$ to the kernel function results (see Equation (5))

$$\text{output}_i = \left(\gamma \left(\text{dist}(x, x_i')\right)^2 + r\right)^d + b_i. \tag{5}$$

To obtain the class probabilities, we apply the softmax function (see Equation (6)) to these outputs

$$P(y = i \mid x) = \frac{\exp(\text{output}_i)}{\sum_{j=1}^{C} \exp(\text{output}_j)}. \tag{6}$$

The softmax function converts the raw outputs into probabilities that sum to one across all classes. This allows the model to interpret the outputs as the likelihood of the input x belonging to each class i, facilitating refined classifications with enhanced accuracy.

In summary, the proposed SVM-based layer is not a standalone SVM model but rather a layer inspired by the SVM formulation that closely resembles a fully connected layer. By

leveraging the nonlinear transformation capabilities of the composite kernel function, it plays a crucial role in capturing the semantic complexity of legal text and enhancing generalization. Our comparative analyses in subsequent sections will further demonstrate the advantages of this novel layer over traditional fully connected layers in the legal Recognizing Textual Entailment (RTE) task.

5. EXPERIMENT AND RESULTS

5.1. Evaluation metrics

In the competition, accuracy is used as the ranking evaluation metric. This metric assesses the performance of a classification model by measuring its overall correctness in all classes or categories. Equation 7 illustrates the computation of accuracy, defined as the ratio of correctly predicted instances to the total number of instances in the dataset. In other words, accuracy tells us the percentage of correctly classified instances out of the total instances

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}}. \tag{7}$$

5.2. Embedding models

To adapt Vietnamese and simultaneously improve system performance on the dataset, we chose pre-trained language models that not only reflect the richness of the language but also help to improve the ability to process and deeply understand text in the Vietnamese context as follows:

ViT5-base: The Vietnamese Text-to-Text Transformer (ViT5) [30] stands as a powerful in the realm of pre-trained models for the Vietnamese language. Crafted by VietAI, it harnesses the power of Transformer architecture, tailored specifically for the intricacies of Vietnamese. Its prowess stems from rigorous training on a vast and varied corpus of Vietnamese texts, employing a T5-style self-supervised pretraining approach. This allows ViT5 to generate text, translate languages, write different kinds of creative content, and answer your questions in an informative way.

PhoBERT-base: PhoBERT [11], a cutting-edge monolingual language model for Vietnamese, represents a significant advancement in NLP. Developed by VinAI Research, it leverages the RoBERTa [31] pre-training technique and is trained on a vast corpus of Vietnamese text data. PhoBERT surpasses prior monolingual and multilingual models, achieving state-of-the-art results across various Vietnamese NLP tasks, including Part-of-speech tagging (POS tagging), Dependency parsing, Named-entity recognition (NER), and Natural language inference (NLI).

VNLawBert: VNLawBERT [32] is a language model specifically designed for answering legal questions in Vietnamese. It is based on the BERT language model [6], which is a state-of-the-art language model that has been shown to be effective for a variety of natural language processing tasks. VNLawBERT has been trained on a large corpus of Vietnamese legal texts, including laws, judicial decisions, and legal articles. This training allows VNLawBERT to understand the nuances of Vietnamese legal and to provide accurate and relevant answers to legal questions.

BARTpho-base: BARTpho [12], a state-of-the-art pre-trained sequence-to-sequence model for Vietnamese, is a remarkable achievement from VinAI. Built upon the BART [10] architecture, known for its versatility in various NLP tasks, BARTpho demonstrates exceptional performance. BARTpho is trained on a massive corpus of Vietnamese text data using the BART pre-training scheme, which allows it to generate text, translate languages, write different kinds of creative content, and answer your questions in an informative way.

mDeBERTa-v3-base: mDeBERTa-v3-base, as introduced by He et al. (2021) [33], represents a state-of-rhe-art multilingual transfromer model crafted by Microsoft. It meticulously trained on a colossal 2.5T CC100 dataset utilizing the ELECTRA-Style pre-training methodology and Gradient Disentangled Embedding Sharing. This unique training approach equips mDeBERTa-v3-base with the adeptness to intricately comprehend diverse languages, thereby exhibiting commendable performance across a multitude of tasks, including Vietnamese.

mBERT-base-cased: mBERT-base-cased [6] is a pre-trained language model that can handle text in over 100 languages. It's based on the Bidirectional Encoder Representations from Transformers (BERT) [6] architecture, which learns relationships between words in a sentence by masking some of them and trying to predict them based on the context. This model is trained on a massive dataset of text from Wikipedia in multiple languages, allowing it to understand the nuances of different languages, including Vietnamese.

mDeBERTa-v3-base (finetuned NLI): We utilized the pre-trained mDeBERTa-v3-base model and further finetuned it on Natural Language Inference datasets, including XNLI [27] and MNLI [28], to enhance its performance for RTE task.

mBERT-base-cased (finetuned NLI): Similarly mDeBERTa-v3-base (finetuned NLI), we employed the pre-trained mBERT-base-cased model, which was subsequently finetuned using the XNLI [27] and MNLI [28] datasets.

5.3. Experimental configuration

All baseline models and our proposed methods were trained and finetuned using the Adam optimization [34]. We utilized a Tesla T4 setup with 16GB of memory to finetune models. The hyperparameters for the models are set as follows: learning rate = 3e-05, dropout = 0.2, batch size = 4, and early stopping patience = 5. For the SVM layer, we choose $\gamma = 0.1$, $c = 1$, $d = 2$.

5.4. Main results

Table 1: Results of our proposed methods and baselines on development and test sets

Model	1 sen SVM layer		5 sen SVM layer		full passage SVM layer		3 sen SVM layer		3 sen FC layer	
	dev	test	dev	test	dev	test	dev	test	dev	test
mDeBERTa-v3-base (finetuned NLI)	0.7632	-	0.7895	-	0.7632	-	**0.8158**	0.7194	0.7895	-
mBERT-base-cased (finetuned NLI)	0.7895	-	0.7368	-	0.7105	-	**0.8158**	-	0.7632	-
mDeBERTa-v3-base	0.5789	-	0.6316	-	0.5526	-	0.6053	-	0.6053	-
mBERT-base-cased	0.6316	-	0.6053	-	0.5789	-	0.6053	-	0.6053	-
ViT5-base	0.5789	-	0.6053	-	0.5263	-	0.6053	-	0.5789	-
PhoBERT-base	0.5789	-	0.6053	-	0.5000	-	0.5789	-	0.5526	-
VNBertLaw	0.6053	-	0.5789	-	0.5789	-	0.6053	-	0.6053	-
BARTpho-base	0.4737	-	0.5526	-	0.5000	-	0.5526	-	0.5263	-
Average	0.6250	-	0.6382	-	0.5822	-	0.6480	-	0.6283	-

In this section, we present details of the results computed based on accuracy by employing pre-trained language models in conjunction with our SVM layer for label classification, drawing comparisons with the traditional fully connected layer. Table 1 presents the results of our experiments with various pre-trained language models, employing both SVM and fully connected (FC) layers for label classification. In the case of 3 related sentences, the SVM layer consistently outperforms other configurations across various pre-trained models. The average accuracy across all models is highest with SVM layer and 3 sentences, emphasizing our proposed method which is effective in label classification tasks. mDeBERTa-v3-base (finetuned NLI) and mBERT-base-cased (finetuned NLI) demonstrate superior performance.

After analyzing the results, we have selected mDeBERTa-v3-base (finetuned NLI) which trained on 3 most related sentences with SVM layer as our submission system, achieving an accuracy score of 0.7194 on the test set. The scoreboard of this challenge can be seen in Table 2.

Table 2: Scoreboard of challenge

Team	Score	Rank
CAN NOT STOP	**0.7698**	**1**
NOWJ	**0.7626**	**2**
A3N1(Ours)	**0.7194**	**3**
Angels	0.5468	4
HNO3	0.5324	5

5.5. Effects of finetuning on NLI dataset

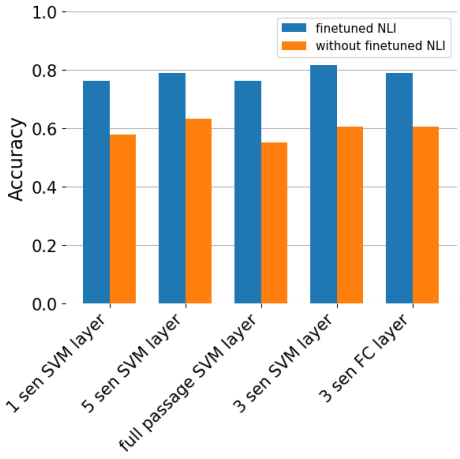

Figure 5: Results on dev set of mDeberta-v3-base with finetuned NLI and without finetuned NLI

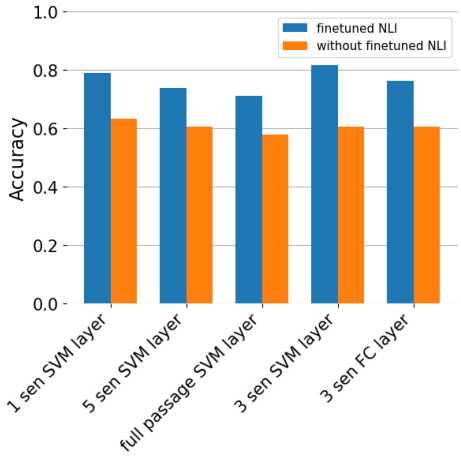

Figure 6: Results on dev set of mBERT-base-cased with finetuned NLI and without finetuned NLI

It can be clearly seen in Figure 5 and Figure 6 that the general trend is when finetuned NLI, performance increases significantly. mDeBERTa-v3-base (finetuned NLI) achieves a

remarkable accuracy of 0.8158 with SVM layer in 3 most related sentences on the development set, outperforming other configurations. Meanwhile, mBERT-base-cased (finetuned NLI) also displays competitive performance across SVM and FC layers. The SVM layer with 3 sentences yields the highest accuracy of 0.8158, demonstrating its effectiveness. What explains the excellence of the two pre-trained models is not just the SVM layer but also the fact that they have undergone training on a large amount of natural language inference data. The training process on large amounts of natural language inference data helps these two pre-trained models deeply capture linguistic structure and context. In this way, they are able to understand and represent complex semantic relationships, thereby enhancing their ability to predict and reason on new data.

In contrast, Table 1 shows that pre-trained models that are not trained on natural language inference data have significantly poor performance. The multilingual pre-trained group showed slightly positive performance as they had higher accuracy than the monolingual pre-trained group in Vietnamese, which showed that the multilingual pre-trained group was better able to capture the context. Despite being pre-trained in legal data, the performance of VNBertLaw is not better than the two pre-trained multilingual models we conducted. Meanwhile, monolingual pre-trained models such as ViT5-base have moderate performance with SVM layers, where the 3 and 5 sentences with SVM layer lead to the highest accuracy at 0.6053. PhoBERT-base performs similarly to ViT5-base, with 5 most related sentences and SVM layer achieved the highest accuracy at 0.6053. BARTpho-base combined with the SVM layer is the lowest-performing model, which has the best accuracy of 0.5526 when using 3 sentences.

5.6. Effect of passage length

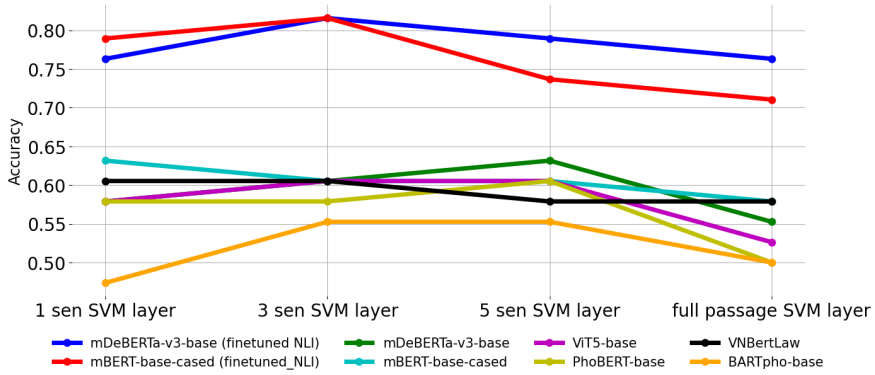

Figure 7: Results of models on development set based on passage length

In the challenge, the given dataset consisted of legal passages ranging from 50 to 800 words, with an average of about 200 words per passage (see Figure 3). The results presented in Figure 7 demonstrate that shortening a passage appropriately yields better results compared to using full passage. The data table indicates that when combining models with the SVM layer using the most relevant sentence, the average accuracy of 0.6250 is lower compared to using the 3 or 5 most related sentences with average accuracy of 0.6480 and 0.6382, respectively. However, using a single sentence yields higher results than using full passage

with average accuracy achieved at 0.5822. This suggests that a single sentence sentence is not enough information for the model, while 3 to 5 sentences provide an optimal balance for the model to perform at its best. On the other hand, using a passage that is too long may reduce the ability of the model to capture the semantic context.

6. DISCUSSION ON HOW TO DEAL WITH EXTREMELY SMALL DATASET

Our proposed method is a good example of using a data-centric approach to improve model performance and deal with an extremely small dataset (see Figure 5, Figure 6, and Table 1). By using two extra large-scale datasets in nearly the same domain with RTE, such as XNLI [27] and MNLI [28], to pre-train the model, we provided the model with much of the linguistic knowledge needed to understand and reason about natural language. This helps the model learn basic language features such as grammar, syntax and meaning of each sentence.

After pre-training, we perform fine-tuning on the challenge dataset with a very limited number of samples. Although an extremely small training set can cause overfitting, thanks to pre-training the model on large datasets, the model has learned general concepts about the language and can apply to a smaller dataset in a new domain without overfitting.

This approach emphasizes the use of rich and diverse data to improve model performance. Instead of focusing on complex architecture or hyperparameters, we focus on giving the model more data to learn from. This results in a model with better generalization and better performance on new tasks and new data spaces.

7. CONCLUSION AND FUTURE WORK

In this paper, we presented a new approach by combining pre-trained language models with a proposed SVM layer, to address the challenge in the Legal Textual Entailment Recognition task. We reviewed and performed detailed tests to ensure the flexibility and effectiveness of the method. By combining detailed linguistic information from pre-trained models and the classification capabilities of the SVM layer as well as extra data in nearly the same domain, we developed an effective method capable of accurately recognizing legal textual entailment. Our method achieved an impressive performance, with an accuracy of 0.7194 on the test set. This result is a testament to the effectiveness of our method in the legal context, especially in the face of data and computation resource shortages.

For future work, we aim to tackle this task efficiently with less data, focusing on improving classification capabilities. Specifically, our plan includes performing experiments involving Few-shot Learning [35], aimed at enhancing the model's ability to learn from small amounts of data. Additionally, among the approaches that we are particularly interested in is the use of techniques related to "prompt tuning" such as [36, 37]. We believe we can leverage the power of large language models by adjusting prompts to achieve the best results. This takes advantage of the linguistic knowledge the model has learned from large amounts of pre-training data, while optimizing performance in a data-limited context.

ACKNOWLEDGEMENTS

This research is funded by Vietnam National University HoChiMinh City (VNU-HCM) under grant number DS2025-26-01.

REFERENCES

[1] I. Dagan, O. Glickman, and B. Magnini, "The PASCAL recognising textual entailment challenge," in *Machine Learning Challenges Workshop*. Springer, 2005, pp. 177–190.

[2] V. H. Tin, V. N. Kiet, and N. N. Luu-Thuy, "ViNLI: a Vietnamese corpus for studies on open-domain natural language inference," in *Proceedings of the 29th International Conference on Computational Linguistics*, 2022, pp. 3858–3872.

[3] C. T. Nguyen and D. T. Nguyen, "Building a Vietnamese dataset for natural language inference models," *SN Computer Science*, vol. 3, no. 5, p. 395, 2022.

[4] H. V. Tin, N. V. Kiet, and N. L.-T. Ngan, "ViANLI: Adversarial Natural Language Inference for Vietnamese," *arXiv preprint arXiv:2406.17716*, 2024.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, pp. 6000–6010, 2017.

[6] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, Jun. 2019, pp. 4171–4186.

[7] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, "Unsupervised cross-lingual representation learning at scale," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Jul. 2020, pp. 8440–8451.

[8] P. He, X. Liu, J. Gao, and W. Chen, "Deberta: Decoding-enhanced bert with disentangled attention," 2021. [Online]. Available: https://arxiv.org/abs/2006.03654

[9] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.

[10] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. Online: Association for Computational Linguistics, Jul. 2020, pp. 7871–7880.

[11] D. Q. Nguyen and A. Tuan Nguyen, "PhoBERT: Pre-trained language models for Vietnamese," in *Findings of the Association for Computational Linguistics: EMNLP 2020*, Nov. 2020, pp. 1037–1042.

[12] N. L. Tran, D. M. Le, and D. Q. Nguyen, "BARTpho: Pre-trained sequence-to-sequence models for Vietnamese," in *Proceedings of the 23rd Annual Conference of the International Speech Communication Association*, 2022, pp. 1751–1755.

[13] T. O. Tran, P. Le Hong *et al.*, "Improving sequence tagging for Vietnamese text using transformer-based neural models," in *Proceedings of the 34th Pacific Asia Conference on Language, Information and Computation*, 2020, pp. 13–20.

[14] V. Tran, H.-T. Nguyen, T. Vo, T.-S. Luu, H.-A. Dang, N.-C. Le, T.-T. Le, M.-T. Nguyen, T.-S. Nguyen, and L.-M. Nguyen, "Vlsp 2023 - lter: A summary of the challenge on legal textual entailment recognition," in *Proceedings of the 10th International Workshop on Vietnamese Language and Speech Processing*, 2023.

[15] R. B. Haim, I. Dagan, B. Dolan, L. Ferro, D. Giampiccolo, B. Magnini, and I. Szpektor, "The Second PASCAL Recognising Textual Entailment Challenge," in *Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment*, vol. 7, 2006, pp. 785–794.

[16] D. Giampiccolo, B. Magnini, I. Dagan, and W. B. Dolan, "The third Pascal recognizing textual entailment challenge," in *Proceedings of the ACL-PASCAL workshop on textual entailment and paraphrasing*, 2007, pp. 1–9.

[17] D. Giampiccolo, H. T. Dang, B. Magnini, I. Dagan, E. Cabrio, and B. Dolan, "The Fourth PASCAL Recognizing Textual Entailment Challenge," in *TAC*, 2008.

[18] B. Luisa, C. Peter, D. Ido, and G. Danilo, "The Seventh PASCAL Recognizing Textual Entailment Challenge," in *TAC*, 2011.

[19] L. Bentivogli, P. Clark, I. Dagan, and D. Giampiccolo, "The Sixth PASCAL Recognizing Textual Entailment Challenge," in *Text Analysis Conference*, 2009. [Online]. Available: https://api.semanticscholar.org/CorpusID:858065

[20] H. T. Anh, N. T. M. Huyen, N. Lien *et al.*, "VLSP 2021-vnNLI Challenge: Vietnamese and English-Vietnamese Textual Entailment," *VNU Journal of Science: Computer Science and Communication Engineering*, vol. 38, no. 2, 2022.

[21] N. H. Thanh, B. M. Quan, C. Nguyen, T. Le, N. M. Phuong, D. T. Binh, V. T. H. Yen, T. Racharak, N. Le Minh, T. D. Vu *et al.*, "A summary of the alqac 2021 competition," in *2021 13th International Conference on Knowledge and Systems Engineering (KSE)*. IEEE, 2021, pp. 1–5.

[22] C. Nguyen, S. T. Luu, T. Tran, A. Trieu, A. Dang, D. Nguyen, H. Nguyen, T. Pham, T. Pham, T.-T. Vo, D.-T. Dol, N.-K. Le, D.-H. Nguyen, N.-C. Le, T.-T. Le, Q. Bui, P. Nguyen, H.-T. Nguyen, V. Tran, and L.-M. Nguyen, "A summary of the alqac 2023 competition," in *2023 15th International Conference on Knowledge and Systems Engineering (KSE)*, 2023, pp. 1–6.

[23] I. M. S. Putra, D. Siahaan, and A. Saikhu, "Recognizing textual entailment: A review of resources, approaches, applications, and challenges," *ICT Express*, 2023.

[24] P. Malakasiotis and I. Androutsopoulos, "Learning textual entailment using SVMs and string similarity measures," in *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, Prague, Jun. 2007, pp. 42–47.

[25] C. Cortes and V. N. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, pp. 273–297, 1995.

[26] J. J. Castillo and L. A. Alemany, "An approach using named entities for recognizing textual entailment," *Theory and Applications of Categories*, 2008.

[27] A. Conneau, R. Rinott, G. Lample, A. Williams, S. Bowman, H. Schwenk, and V. Stoyanov, "XNLI: Evaluating cross-lingual sentence representations," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, Eds. Brussels, Belgium: Association for Computational Linguistics, Oct.-Nov. 2018, pp. 2475–2485. [Online]. Available: https://aclanthology.org/D18-1269

[28] A. Williams, N. Nangia, and S. Bowman, "A broad-coverage challenge corpus for sentence understanding through inference," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics, 2018, pp. 1112–1122.

[29] Q. L. Phan, T. H. P. Doan, N. H. Le, N. B. D. Tran, and T. N. Huynh, "Vietnamese sentence paraphrase identification using sentence-bert and phobert," in *Intelligence of Things: Technologies and Applications*, 2022, pp. 416–423.

[30] L. Phan, H. Tran, H. Nguyen, and T. H. Trinh, "ViT5: Pretrained text-to-text transformer for Vietnamese language generation," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Student Research Workshop*, D. Ippolito, L. H. Li, M. L. Pacheco, D. Chen, and N. Xue, Eds. Hybrid: Seattle, Washington + Online: Association for Computational Linguistics, Jul. 2022, pp. 136–142. [Online]. Available: https://aclanthology.org/2022.naacl-srw.18

[31] L. Zhuang, L. Wayne, S. Ya, and Z. Jun, "A robustly optimized BERT pre-training approach with post-training," in *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, S. Li, M. Sun, Y. Liu, H. Wu, K. Liu, W. Che, S. He, and G. Rao, Eds. Huhhot, China: Chinese Information Processing Society of China, Aug. 2021, pp. 1218–1227. [Online]. Available: https://aclanthology.org/2021.ccl-1.108

[32] C.-N. Chau, T.-S. Nguyen, and L.-M. Nguyen, "VNLawBERT: A Vietnamese Legal Answer Selection Approach Using BERT Language Model," in *2020 7th NAFOSTED Conference on Information and Computer Science (NICS)*, 2020, pp. 298–301.

[33] P. He, J. Gao, and W. Chen, "Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing," 2023. [Online]. Available: https://arxiv.org/abs/2111.09543

[34] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations (ICLR)*, 2015.

[35] L. Tunstall, N. Reimers, U. E. S. Jo, L. Bates, D. Korat, M. Wasserblat, and O. Pereg, "Efficient few-shot learning without prompts," *arXiv preprint arXiv:2209.11055*, 2022.

[36] W. Zhu and M. Tan, "Improving prompt tuning with learned prompting layers," *arXiv preprint arXiv:2310.20127*, 2023.

[37] C. Peng, X. Yang, K. E. Smith, Z. Yu, A. Chen, J. Bian, and Y. Wu, "Model tuning or prompt tuning? a study of large language models for clinical concept and relation extraction," *Journal of Biomedical Informatics*, p. 104630, 2024.

Received on April 19, 2024
Accepted on April 07, 2025