Multi-task learning fact checking with supervised reading comprehension-based evidence extraction

Manh Trong Nguyen[a,b], Tri Thien Nguyen[a,b], Kiet Van Nguyen[a,b,*]

[a] *Faculty of Information Science and Engineering, University of Information Technology, Ho Chi Minh City, Vietnam*
[b] *Vietnam National University, Ho Chi Minh City, Vietnam*

Abstract

In the current digital era, fact-checking has become increasingly critical due to the widespread dissemination of fake news and inaccurate information. A major challenge in this task is the ability to provide relevant evidence to verify claims effectively. Existing approaches to evidence selection often rely on unsupervised methods like semantic similarity, which uses cosine similarity to find relevant evidence. However, these fail to accurately capture the nuanced contextual relevance necessary to distinguish between supportive and refuting evidence. These limitations can result in less reliable fact-checking outcomes. Our study proposes a supervised reading comprehension (SRC) model for evidence extraction (EE) tasks to address this issue. By leveraging supervised learning, our model aims to extract and assess evidence more accurately, addressing the challenges posed by unsupervised semantic similarity methods. The experimental results demonstrate the superior performance of the SRC model, particularly when utilizing the XLM-R and InfoXLM model, on the ViWikiFC and ISE-DSC01 datasets. Furthermore, we propose the SRC-FC model, which jointly applies multi-task learning (MTL) to train evidence extraction and verdict prediction tasks. For the evidence extraction task, the SRC-FC model proves particularly effective for low-resource languages, such as Vietnamese. The SRC-FC model delivered outstanding performance with 93.59% (F1) on the ViWikiFC dataset and 92.24% (F1) on the ISE-DSC01 dataset, when using the pre-trained XLM-R model as the embedding model. Our results demonstrate the advantages of integrating SRC and MTL approaches for more reliable and accurate fact-checking, especially in challenging linguistic environments.

Keywords: Fact-checking, Supervised reading comprehension, Multi-task learning, Low-resource NLP

*Corresponding author.
Email addresses: 21520343@gm.uit.edu.vn (Manh Trong Nguyen), 21522707@gm.uit.edu.vn (Tri Thien Nguyen), kietnv@uit.edu.vn (Kiet Van Nguyen)

1. Introduction

In the era of rapid development of information technology and the Internet, fact-checking has become critical in combating misinformation. The proliferation of fake news and inaccurate information has become widespread. As a result, verifying the truth of a statement, article, or piece of information is now a significant challenge for organizations and individuals. Among these challenges, the challenging task for researchers in fact-checking is to find appropriate evidence to verify claims. The fact-checking task involves assessing the authenticity of a claim through evidence gathered from reputable information sources. Traditionally, claims are manually verified, which requires considerable time and effort from fact-checkers [1]. However, with technological advancements, automated fact-checking systems have been developed to assist experts in making more efficient and accurate decisions. Systems like ClaimBuster [2] utilize machine learning techniques, such as support vector machines (SVM), and linguistic features to classify and prioritize claims. In addition, comprehensive surveys, such as the one conducted by Guo et al. (2022) [3], highlight the integration of natural language processing (NLP) and machine learning (ML) in developing robust automated fact-checking systems, emphasizing their role in addressing the growing challenge of misinformation in the digital age.

According to Vladika et al. (2023) [4], the standard framework for the fact-checking task comprises three components: document retrieval (DR), evidence selection (ES), and verdict prediction (VP). Among these three tasks, document retrieval and evidence selection are the two sub-tasks of the evidence retrieval task. To typically address the task of evidence retrieval, previous studies use word embedding methods to calculate the cosine similarity between claims and documents such as TF-IDF [5] or use BM25 to retrieve the top k documents, then employ the T5 model to re-rank and select a new top k documents [6, 7], then use models like Sentence-BERT [8, 9], or Longformer (binary head) [6, 7] to select evidence based on cosine similarity from the top k candidate documents. Most of these methods only measure the semantic similarity between the claim and the candidate evidence, which is not truly optimal for evidence that refutes the claim [7]. Similarity-based methods, using unsupervised learning models, may identify evidence that is contextually related but not always effective in refuting or supporting the claim. This approach also struggles with distinguishing between evidence that supports a claim and evidence that directly refutes it. As a result, similar pieces of evidence might be misclassified, potentially leading to incorrect conclusions. This motivated us to apply the supervised reading comprehension model, a different approach for the evidence retrieval task. The SRC model can perform deeper semantic analysis, allowing it not only to rely on keywords but also to understand the contextual meaning of passages. This is particularly important in accurately extracting evidence from large datasets, where traditional techniques like cosine similarity often fall short.

In addition to single-task training methods for evidence selection, there are several other effective systems. These systems often integrate evidence selection and verdict prediction tasks by sharing hidden layers, rather than building separate models for each task and using them in a pipeline. Presently, several combined models have been developed, such as ParagraphJoint [10], ARSJoint [11], and MultiVerS [6]. These combined models use the multi-task learning (MTL) method to train both the evidence selection

and verdict prediction tasks simultaneously. The models integrate shared representations of the claim and abstract by concatenating the claim with the entire abstract of the candidate document and then converting them into dense representations. This addresses the issue of contextual deficiency in predicting labels for claims. Given the advantages of the MTL approach, we decided to apply it to the SRC model to improve the performance of the evidence extraction task.

In this paper, we present three main contributions to enhancing the performance of the evidence retrieval task, specifically in extracting evidence relevant to the claim:

- First, we introduce an evidence extraction task immediately following the evidence selection task. The objective is to accurately identify the evidence that supports claim verification from the top k evidence selected during the evidence selection task. In the evidence extraction task, we pinpoint the exact position of the evidence within the context, which is compiled from the top k candidate evidence, and this top k evidence is extracted from candidate documents. We propose using the supervised machine reading (SRC) model to extract the evidence with input from k candidate evidence. In our approach, we use the claim sentence as the question, the needed evidence sentence as the answer, and k candidate evidence sentences as the passage and predict the position of the evidence within the context similarly to how a question answer is predicted in the answer extraction task. The SRC model can read and understand a text passage's content, allowing it to answer questions based on information from the text passage.

- Secondly, we propose the SRC-FC model, which utilizes the multi-task learning (MTL) approach by training both evidence extraction and verdict prediction tasks to enhance the performance of the evidence extraction task. According to ARSjoint [11], the MTL approach helps minimize information loss during the information transmission process between tasks in the pipeline. This is especially beneficial for low-resource languages like Vietnamese, as it addresses the issue of insufficient data for model training. Our experiments have demonstrated that the SRC-FC model performed better than the standalone SRC model.

- Finally, we conduct a feature analysis of the two benchmark datasets to identify the challenges in solving the evidence retrieval task. The features we explored between the claim and original evidence sentences include lexical overlap, new word rate, semantic similarity, and word embeddings. Through these features, we aim to uncover aspects that can be improved in the future.

The structure of this paper is organized as follows. Section 1 and Section 2 introduce our new approach and the proposed method, along with our surveys of related tasks and current approaches for evidence retrieval in fact-checking. Section 3 details the ideas and processes of our proposed method. Next, Section 4 includes the experimental setup, the results we obtained, and our analysis based on these results. Finally, Section 5 describes conclusions and future directions for the proposed method.

2. Related Works

In this section, we first review notable works related to evidence retrieval tasks in fact-checking. Next, we delve into studies that have used supervised machine reading (SRC) models for sentence-level evidence retrieval. Finally, we discuss studies that have used the multi-task learning approach to improve performance on sub-tasks within the fact-checking.

2.1. Evidence Retrieval Task

The evidence retrieval task often consists of two subtasks: document retrieval and evidence selection. Document retrieval involves retrieving documents that may contain evidence relevant to the claim. For SCIFACT [12], the document retrieval task focuses on retrieving abstract passages from a corpus containing around 5000 abstracts on scientific topics. The pipeline model VeriSci [12] uses the TF-IDF metric to retrieve the k most relevant abstracts to the claim. The VerT5erini model [13] later employed a new approach: first retrieving the top k relevant abstracts using BM25, then using the T5 model re-ranker [14], further filtering out more relevant abstracts to use as input for the next step, which is the evidence selection task. Evidence selection is the task of choosing the relevant evidence sentences from the previously retrieved documents to use as evidence for predicting the claim's label. The baseline model of PUB-HEALTH [8] uses the Sentence-BERT model [15] to extract the top 5 sentences most relevant to the claim, and then combines them with the claim to predict the label. However, this study only focuses on generating an explanation for the prediction result without selecting an explicit evidence sentence from these 5 sentences.

Recent advances, such as the GERE [16] framework, integrate data recovery and evidence selection into a unified approach. GERE employs a bidirectional transformer encoder for mapping claims into vectors and a sequence generation approach for document titles and evidence sentences. This method enhances the accuracy of retrieved documents and evidence by considering their dependencies. Another innovative method is a multi-hop evidence retrieval approach, Mr.Cod [17]. It utilizes a graph-based algorithm to identify evidence paths across multiple documents. This technique addresses the complexity of retrieving interconnected pieces of evidence scattered across a large corpus. By ranking these paths using dense retrievers, Mr.Cod ensures that the most relevant evidence is selected, providing a robust foundation for downstream tasks like relation extraction.

2.2. Supervised Reading Comprehension for Sentence-level Answers

In our approach, extracted evidences will be whole sentences for fact checking. Most current datasets for the SRC task, however, typically provide answers as sentence parts rather than a whole sentence [18]. Thus, if our proposed method yields positive results, it could represent a discovery regarding the ability of SRC models to extract sentence-level answers.

The CMRC 2019 dataset [19] is specifically related to the machine reading comprehension task and focuses on evaluating sentence-level inference abilities. The authors introduced a new task called Sentence Cloze-style machine reading comprehension (SC-MRC) to further test a machine's understanding in the context of long-range

reasoning. This task involves selecting the correct candidate sentence from a paragraph with multiple gaps. To increase difficulty, they also included fake candidates that closely resemble the correct ones, requiring the machine to judge their accuracy within the context. Baseline models were developed using popular pre-trained language models. The proposed method [19] uses BERT and its variants to fill in blanks in a passage by first replacing the blanks with special tokens and concatenating the passage with each answer option to form input sequences. BERT processes these sequences to generate hidden representations, which are then used to compute logics and predict probabilities for each blank. The model is trained to minimize the cross-entropy between these predicted probabilities and the actual positions of the blanks. During decoding, the answer option that yields the highest probability for a blank is selected as the prediction, allowing the same answer option to be used for multiple blanks if it provides the best fit.

WIKIQA [20] is another sentence-level dataset, consisting of publicly available question and sentence pairs that have been collected and annotated for open-domain question-answering research. Constructed through a more natural process, WIKIQA is significantly larger than previous datasets. It also includes questions that do not have correct sentences, which allows researchers to explore answer triggering—a crucial component in any QA system. Gharagozlou et al. (2022) [21] introduces a novel approach named RLAS-BIABC for Answer Selection (AS) for the WIKIQA dataset. This method combines an attention mechanism-based LSTM model with BERT word embeddings and incorporates an improved Artificial Bee Colony (ABC) algorithm for pretraining, alongside reinforcement learning for training the BP (Backpropagation) algorithm. The model aims to classify pairs of questions and answers as either positive (real answers) or negative (fake answers). To avoid local optima in the model's training, the policy weights are initialized using the improved ABC algorithm. The paper also introduces a mutual learning technique that selects the better candidate based on fitness from two individuals, guided by a mutual learning factor. Do et al. (2021) [22] illustrates MRC-based approaches archived better performances than other approaches (including ranking-based and classification-based) on sentence extraction-based machine reading comprehension.

2.3. Multi-Task Learning Approach in Fact-Checking

In fact-checking, joint models have significantly improved the accuracy and efficiency of verifying various claims. One of the pioneering contributions in this area is VerT5erini [13], which utilized the T5 model for abstract retrieval, evidence selection, and verdict prediction. These three tasks are essential auxiliary tasks for claim verification. The study demonstrated how leveraging the T5 model could enhance the overall performance in verifying claims by addressing each task effectively. Following this, the ParagraphJoint [10] introduced a multi-task learning model specifically designed for the SCIFACT task. The model directly computed context-aware sentence embeddings using the BERT model and simultaneously trained for evidence selection and label prediction. The approach aimed to improve the coherence and accuracy of the fact-checking process by integrating these tasks into a single training framework. Similarly, ARSJoint [11] developed a combined learning model that focused on abstract retrieval, evidence selection, and label prediction within the machine reading

comprehension (MRC) framework. This model incorporated additional claim information to enhance the learning process. Experimental results on the SCIFACT dataset demonstrated that ARSJoint outperformed both ParagraphJoint and VerT5erini at both the sentence and abstract levels, indicating its superior effectiveness in fact-checking tasks. MultiVerS [6] also proposed a multi-task learning model, but it focused on label prediction and evidence identification based on shared encoding of the claim and context. This shared encoding allows the model to efficiently leverage the interdependence between label prediction and evidence identification, resulting in improved accuracy in fact verification tasks. MultiVerS demonstrates that this approach not only streamlines the process of identifying relevant evidence but also enhances the model's overall performance in predicting the veracity of claims.

Additionally, Li et al. (2018) [23] introduces an end-to-end multi-task learning model with bi-directional attention (EMBA) for the FEVER task [5]. The model simultaneously extracts evidence and verifies claims by leveraging a bi-directional attention mechanism that processes claim-page pairs to enhance claim and page representations. Experimental results indicate that EMBA achieves performance comparable to the baseline, contributing to improved prediction accuracy by effectively utilizing comprehensive contextual information.

3. Methodology

In this section, we introduce the preliminaries in Section 3.1 and the proposed method for the evidence extraction using the supervised reading comprehension approach in Section 3.2. Additionally, we enhance the performance of the supervised reading comprehension model through the multi-task learning method in Section 3.3.

3.1. Preliminaries

Task Definitions. Our proposed method focuses on refining the approach to extracting evidence from the top k document. Instead of selecting evidence from the top candidate documents based solely on semantic similarity, as done in previous studies, we first use a semantic similarity-based method to identify the top candidate evidence. These pieces of evidence are then combined into a single passage. We subsequently apply a supervised reading comprehension model to accurately extract the specific location of the evidence that supports the claim verification. Therefore, our method involves two tasks: top k evidence selection and evidence extraction. The selection of top k candidate evidence aims to identify the most potential evidence to support the prediction of the claim's verdict. The input for selecting the top k evidence task will include claim (C), gold evidence (E), and corpus (D), which contains all sentences in the dataset. The output of this task will contain a set of top k candidate evidence $Tk = \{E_1, E_2, ..., E_k\}$ used as the context for the claim in the evidence extraction task. For the evidence extraction task, the goal is to accurately identify the evidence position within the top k candidate evidence, which can then improve the accuracy of the claim's verdict prediction. The SRC model will take the context (Ct) and the claim (C) as input, and the model's output is the start and end positions of the sentence within Ct that is considered the evidence for the claim.

Method Overview. Our method improves the fact-checking process by incorporating a supervised reading comprehension (SRC) model for evidence extraction. The approach is divided into two main tasks:

- Top k evidence selection: We first identify the top k relevant evidence sentences by calculating the cosine similarity between the claim and sentences in the dataset. These sentences are concatenated to form a single context for further analysis.
- Evidence extraction: The concatenated evidence is input into the SRC model, which predicts the exact start and end positions of the evidence within the context, improving claim verification accuracy.

To enhance the SRC model's performance, particularly for low-resource languages, we propose the SRC-FC model. This model uses a multi-task learning (MTL) approach, combining evidence extraction and verdict prediction tasks to improve accuracy. Sections detailing these tasks include Section 3.2 for the supervised reading comprehension approach and Section 3.3 for the multi-task learning enhancement.

3.2. The Supervised Reading Comprehension Approach for Evidence Extraction

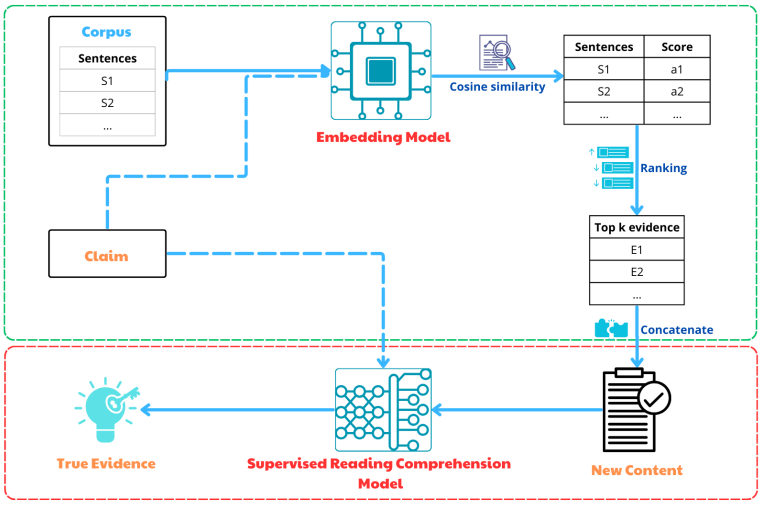

Figure 1: Overview of our proposed method. The green box represents the process of selecting the top k evidence candidates, while the red box represents the process of extracting evidence using the SRC model.

According to VerT5erini, in the evidence selection task, the fine-tuned T5 model on the MS MARCO dataset (the dataset used for the SRC task) was used to predict the degree of relationship between the claim and the abstract top k accessed by BM25 to retrieve the evidence sentence. We proposed considering the top k sentences after being retrieved as a context and the claim's original evidence as the answer for the SRC task based on the idea of using the MRC dataset for the evidence selection task. The

model's output is the starting and ending position of the evidence sentence, instead of outputting the similarity between sentences in context and claims like VerT5erini. Our proposed method execution process is illustrated in Figure 1.

Algorithm 1 Extracting top-k candidate evidence for the SRC model

Input: A claim C, gold evidence E and a corpus D containing all evidence sentences.
Output: The top k evidence is combined into a context for the SRC model.

procedure EXTRACTING TOP-K CANDIDATE EVIDENCE
  $S \leftarrow$ Establishing a score list for similarity between C and each sentence in D
  for each sentence S_i in D **do**
    $e_C \leftarrow$ Extracting embedding vector of C using Sentence Transformers, TF-IDF, or BM25
    $e_{S_i} \leftarrow$ Extracting embedding vector of S_i using the same method as e_C
    score$_i \leftarrow$ Calculating cosine similarity between e_C and e_{S_i}
    $S \leftarrow$ Add score$_i$ to the score list
  end for
  $S \leftarrow$ **Ranking** the score list S in descending order
  $L \leftarrow$ Selecting the top $k-1$ sentences from D based on the highest scores in S
  if E not in L **then**
    $L \leftarrow L \cup E$
  else
    $L \leftarrow$ Selecting the top k sentences from D based on the highest scores in S
  end if
  $L \leftarrow$ Randomly shuffling the order of sentences in L
  $Ct \leftarrow$ Combining sentences in L into a single context
  return Ct
end procedure

Training Data: Firstly, we aim to select the top k candidate evidence to serve as a context for the supervised reading comprehension model using Algorithm 1. To extract the top k candidate pieces of evidence, we measure the similarity between each claim sentence and all sentences in the corpus. We employ several models, including Sentence Transformers, TF-IDF, and BM25, to compute the embedding vectors and determine these similarities. Then we find the cosine similarity score between each sentence and the claim sentence. Next, we select the top k sentences with the highest cosine similarity scores to use as context input for the SRC model. Once we have the top k candidate evidence, we check to see if they contain the original evidence for the claim to ensure the accuracy of the data. If the original evidence is not included, we replace one of the candidate evidence with the original evidence. Since the NEI label in the ISE-DSC01 dataset does not include evidence, the step of checking for original evidence in the top k candidate sentences is skipped. To enhance the generalization of the SRC model, we randomly shuffle the order of these k sentences and combine the shuffled sentences into a single context for the model. After obtaining the new context,

we determine the start and end positions of the evidence within the new context and tokenize each sample using Algorithm 2.

Algorithm 2 Preprocessing data before training the SRC model.

Input: Dataset containing claims (C), context (Ct), evidence (E) and pretrained tokenizer (T).

Output: The tokenized input, the start and end positions of E within Ct

procedure PRE-PROCESSING DATA BEFORE TRAINING THE SRC MODEL
    $Inputs \leftarrow$ Tokenizing C and Ct using T.
    if E is not fully within Ct **then**
        $Start, End \leftarrow$ Setting start and end positions to (0, 0).
    else
        $Start, End \leftarrow$ Determining the start and end positions of E in $Inputs$.
    end if
    $Inputs \leftarrow$ Adding $Start$ and End to $Inputs$.
    return $Inputs$
end procedure

Fine-tuning Model: We fine-tune the SRC model using Algorithm 3. The SRC model uses input data consisting of the claim sentence and context processed by Algorithms 1 and 2. The model's output is the start and end positions of the evidence within the context. The SRC model is trained using the Trainer function from the Hugging Face [24]. In the training process, we use the training set for model training and the development set to fine-tune hyperparameters. The example input and output for the SRC model are shown in Figure 2.

Algorithm 3 Training the SRC model for evidence extraction.

Input: The training set (TS) and the development set (DS) are processed using Algorithms 1 and 2.

Output: Fine-tuned SRC model for evidence extraction.

procedure TRAINING THE SRC MODEL FOR EVIDENCE EXTRACTION
    $args \leftarrow$ The set of hyperparameters for training.
    $PM \leftarrow$ Loading pretrained Question-Answering model.
    $M \leftarrow$ Initializing $Trainer(args, TS, DS, PM)$ and train it.
    return M
end procedure

3.3. SRC-FC: Strengthen supervised reading comprehension Model Using Multi-Task Learning Approach

After establishing the effectiveness of the SRC model in evidence extraction, we introduce the SRC-FC model, which integrates multi-task learning to further enhance performance by joint training on evidence extraction and verdict prediction. With the benefit shown in [11], multi-task learning helps minimize information loss between

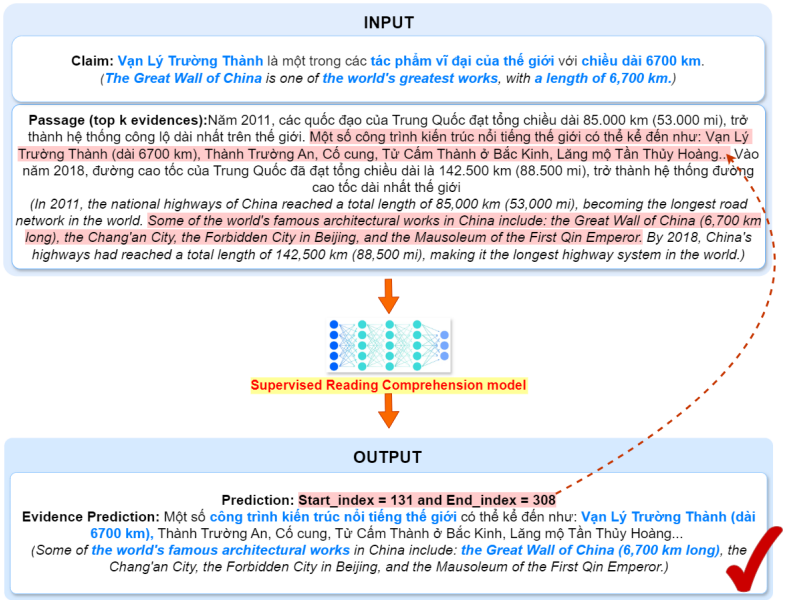

Figure 2: Illustrates the input data and the predicted output of the SRC model. The model's output is the start and end positions of the predicted evidence within the passage, highlighted in **red**. The phrases highlighted in **bold blue** match the information between the claim and the extracted evidence.

tasks in the pipeline when tasks are highly related to each other. We implement the SRC-FC model according to the architecture shown in Figure 4. For better visualization, Figure 3 provides an illustrative example of the input and output of the SRC-FC model.

3.3.1. The SRC-FC Model Architecture

Input Data. The input data of the SRC-FC model $D = \{Ct_i, C_i, A_i, L_i\}_{i=1}^{n}$ with Ct_i as context (with the context generated according to the method we mentioned in Section 3.2), C_i is the claim, A_i is evidence of the claim, L_i is the label for the claim based on evidence, and n is denoted as the number of samples. In the evidence extraction task, A_i will contain the starting and ending positions of the evidence within the context. There is no evidence for the NEI (Not Enough Information) label in the ISE-DSC01 dataset, so the starting and ending positions will default to 0. For each data sample $\{Ct_i, C_i, A_i, L_i\}$, before being embedded by the embedding model, will be concatenated together with special tokens `[CLS]` and `[SEP]`, which will obtain the following string: S_i = "`[CLS]` Ct_i `[SEP]` C_i `[SEP]`". This helps the model clearly distinguish the structure and relationship between the claim and the context, thereby improving the semantic encoding process of both the claim and the context. It also ensures that the information of the claim and context is coherent and not disordered when fed into the model.

Embeddings model. In the the SRC-FC model, we chose to use multilingual models instead of those specifically designed for Vietnamese. A key reason for this deci-

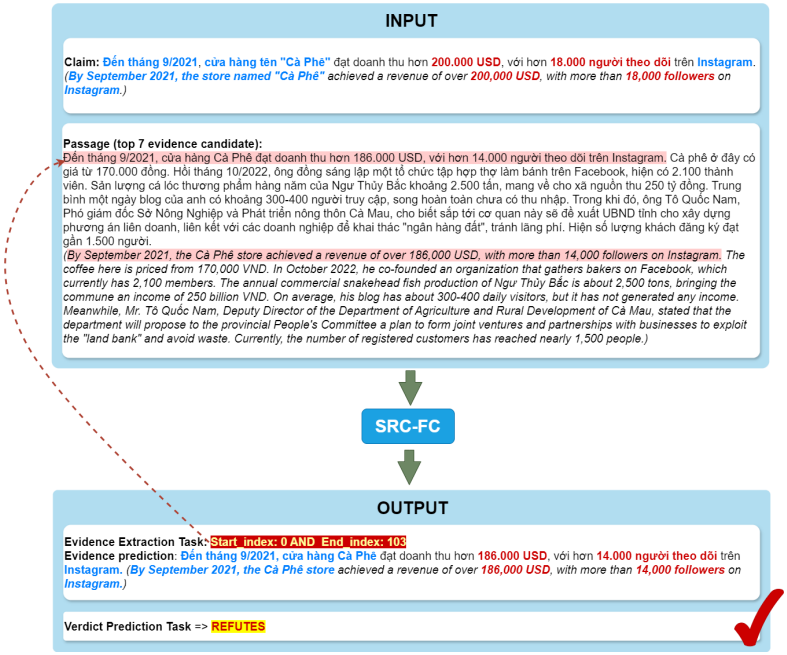

Figure 3: Illustrates the input data and the predicted output of the SRC-FC model. The model's output is the start and end positions of the predicted evidence within the passage for the evidence extraction task, highlighted in **red**, and the predicted labels for the verdict prediction task. The phrases highlighted in **bold blue** are the matching information between the claim and the extracted evidence. The phrases highlighted in **bold red** are information that refutes the claim.

sion is the generalization capability and multilingual processing ability of models like XLM-R or InfoXLM. These models have been trained in various languages, including Vietnamese, allowing them to capture the semantic and syntactic features of multiple languages effectively. This is particularly important when we deal with tasks that require contextual alignment between claims and evidence. Additionally, using multilingual models expands our model's potential application to other languages without redesigning the model or retraining it from scratch. Finally, although specialized models for Vietnamese like ViT5 or ViDeBERTa have shown high performance on certain tasks, their generalization capability to other languages may be limited. Meanwhile, multilingual models offer a more flexible and efficient solution for our problem, especially as we aim for a system that can scale beyond the scope of the Vietnamese language.

Context Alignment Module. We used the multilingual pretrained model for contextual embedding, and this model only provides sub-word features. This sub-word representation increases the input context size for the model, making training time and resource usage more costly. Therefore, converting sub-word features into word features is essential, especially for Vietnamese, where word-level characteristics can significantly improve model performance. To combine sub-word features, we need to extract the list of the number of sub-words divided from words for each data sample. For each

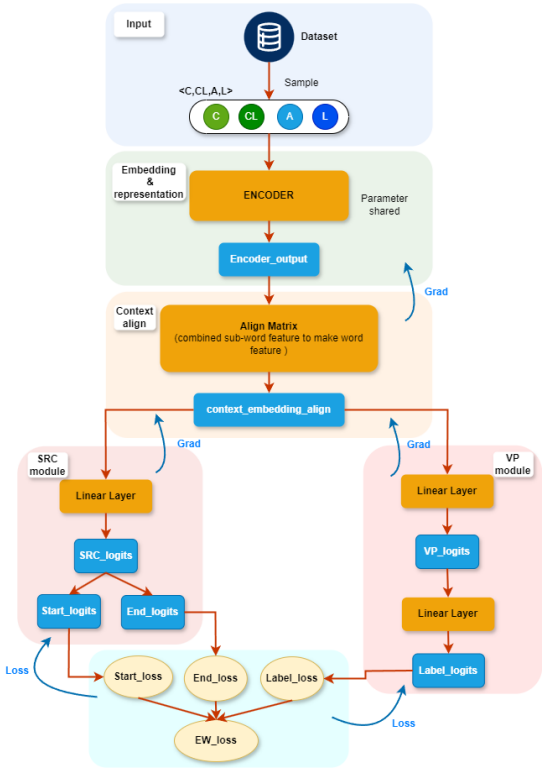

Figure 4: Architecture of the SRC-FC model by module. The dark yellow boxes represent the processing components of the module, while the blue boxes represent the outputs obtained after processing.

sample S_i, we split it into a list of words and encode each word. Then, we count the dimensions of each word after encoding, resulting in a list of sub-word counts W_i.

$$H_i = E(S_i) \tag{1}$$

$$S'_i = Split(S_i) \tag{2}$$

$$W_i = [|E(w)| \mid w \in S'_i] \tag{3}$$

where S_i denotes the i-th sample, where $i \in [1, n]$ and n is the number of samples. $H_i \in \mathbb{R}^{m \times d}$ represents the model's output of hidden vectors. The function *Split* separates S_i into individual words, transforming S_i into a list of words $S'_i \in \mathbb{R}^{m'}$. $W_i \in \mathbb{R}^{m'}$ represents the number of sub-words for each word. We initialize the alignment matrix (M). This matrix helps determine the positions of the sub-words of each word, and when multiplied with the matrix H_i, it will aggregate the separated word features into

sub-word features. The alignment matrix M_i is defined by the following formula:

$$M_i[j,k] = \begin{cases} 1 & \textit{if } W_{\mathrm{i}}[j] > 0 \text{ and k} \in [start_i, start_i + W_{\mathrm{i}}[j]] \\ 0 & \textit{otherwise} \end{cases} \tag{4}$$

With:

$$start_i = \sum_{k=0}^{j-1} \mathbf{W}_{\mathrm{i}}[k] \tag{5}$$

where $start_i$ denotes the starting position of the vector representing the j-th word in H_i, where $j \in [0, m']$ and $k \in [0, m]$. $M_i \in \mathbb{R}^{m' \times m}$ represents the alignment matrix marking the positions of sub-word features. After obtaining the matrix M_i, the matrix representing the word features (H'_i) is created by performing a matrix multiplication between M_i and H_i. The sub-word feature representations will be aggregated into word feature representations through matrix multiplication. Simply put, the word feature vector is generated by summing the sub-word feature vectors.

$$H'_{\mathrm{i}} = M_{\mathrm{i}} \cdot H_{\mathrm{i}} \tag{6}$$

where $H'_i \in \mathbb{R}^{m' \times d}$ represents the new feature representation of S_i.

Task-Specific Layers. After obtaining the new feature representation H'_i, to train and make predictions for the SRC and VP task. H'_i will be fed into separate Multi-Layer Perceptron (MLP) layers for each task. Additionally, after obtaining the probability distribution of the prediction results for both tasks, these will be concatenated once more and passed through an additional MLP layer to produce the final probability distribution. This step is intended to share feature information between the two tasks before making the final prediction. The model performs SRC and VP task computation as shown below.

$$h_{\mathrm{i}}^m = \textit{MLP}_1(H'_i) \tag{7}$$

$$h_{\mathrm{i}}^v = \textit{MLP}_2(H'_i) \tag{8}$$

$$h_{\mathrm{i}}^s, h_{\mathrm{i}}^e = \textit{Split}(h_{\mathrm{i}}^m) \tag{9}$$

$$h_{\mathrm{i}}^v = \textit{MLP}_3(\textit{Concat}(h_{\mathrm{i}}^m, h_{\mathrm{i}}^v)) \tag{10}$$

where $h_{\mathrm{i}}^m \in \mathbb{R}^{n \times d}$ denotes the feature vector for the SRC task, $h_{\mathrm{i}}^v \in \mathbb{R}^{n \times d}$ denotes the feature vector for the VP task. $h_{\mathrm{i}}^s \in \mathbb{R}^{n \times d}$, $h_{\mathrm{i}}^e \in \mathbb{R}^{n \times d}$ denote the hidden vectors of the model output start position and end position sequences.

Loss Computation. For calculating the loss of the SRC-FC model, we compute the loss for each task using the cross-entropy function. We then take the average of the losses from the two tasks and consider this as the value to be optimized by the model.

$$\mathcal{L}_{\text{SRC}} = \frac{\mathcal{L}_{CE}(y^s, h^s) + \mathcal{L}_{CE}(y^e, h^e)}{2} \tag{11}$$

$$\mathcal{L}_{\text{vp}} = \mathcal{L}_{CE}(y^v, h^v) \tag{12}$$

$$\mathcal{L}_{\text{total}} = \frac{\mathcal{L}_{\text{SRC}} + \mathcal{L}_{\text{vp}}}{2} \tag{13}$$

where $\mathcal{L}_{CE}$ represents the cross-entropy loss, which is a commonly used loss function in classification tasks. The terms y^s, y^e, and y^v denote the true labels for the EE and VP tasks, respectively. The terms h^s, h^e and h^v represent the model's output for the SRC and VP tasks. The loss $\mathcal{L}_{\text{SRC}}$ is the average cross-entropy loss over the SRC task, while $\mathcal{L}_{\text{vp}}$ is the cross-entropy loss for the VP task. Finally, $\mathcal{L}_{\text{total}}$ is the overall loss, calculated as the average of $\mathcal{L}_{\text{SRC}}$ and $\mathcal{L}_{\text{vp}}$.

3.3.2. Training the SRC-FC Model

Training Data. We first preprocess the data for the SRC-FC model similarly to the SRC model in Section 3.3.1, using Algorithms 1 and 2. However, we will add labels for the Verdict Prediction task into the $Inputs$ in Algorithm 2, as described in Algorithm 4.

Training the Model. We perform training on the SRC-FC model using Algorithm 5. The SRC-FC model uses input data consisting of the claim sentence and context processed by Algorithms 1 and 4. The model's output is the start and end positions of the evidence within the context, and label of the Verdict Prediction task. The SRC-FC model is also trained using the Trainer function from the Hugging Face [24] like the SRC model in Section 3.2. In the training process, we use the training set for model training and the development set to fine-tune hyperparameters.

Algorithm 4 Preprocessing data before training the SRC-FC model.

Input: Dataset containing claims (C), context (Ct), evidence (E), label (L) and pre-trained tokenizer (T).

Output: The tokenized input, label of VP task, the start and end positions of E within Ct

procedure PREPROCESS DATA BEFORE TRAINING THE SRC-FC MODEL
    $L \leftarrow$ encode L.
    $Inputs \leftarrow$ Tokenize C and Ct using T.
    if E is not fully within Ct **then**
        $Start, End \leftarrow$ Set start and end positions to (0, 0).
    else
        $Start, End \leftarrow$ Determine the start and end positions of E in $Inputs$.
    end if
    $Inputs \leftarrow$ Add $Start$, End and L to $Inputs$.
    return $Inputs$
end procedure

Algorithm 5 Training the SRC-FC model for evidence extraction and verdict prediction.

Input: The training set (TS) and the development set (DS) are processed using Algorithms 1 and 4.

Output: Trained SRC-FC model for evidence extraction and verdict prediction.

procedure TRAINING THE SRC-FC MODEL
    $args \leftarrow$ The set of hyperparameters for training.
    $M \leftarrow$ Define the SRC-FC model architecture and load it.
    $M \leftarrow$ Initialize $Trainer(args, TS, DS, M)$ and train it.
    return M
end procedure

Dataset	Claims	Domain	Evidence type	Has NEI	NEI has evidence
ViWikiFC	20,916	Wikipedia	Single	✓	✓
ISE-DSC01	48,000+	News	Single	✓	✗

Table 1: Summary of Vietnamese datasets used in experiments.

4. Experiments and Results

4.1. Datasets Used

We selected two Vietnamese benchmark datasets for our study: ViWikiFC [25] and ISE-DSC01[1]. Both datasets are designed for fact-checking tasks and are publicly available for research purposes and described in Table 1. The ViWikiFC dataset, as shown in Figure 5, is well-balanced across its labels. In contrast, the ISE-DSC01 dataset exhibits a slight imbalance in the REFUTES label. For the ISE-DSC01 dataset, only the original evidence of each training set is public, so we re-divide the dataset for the training set. Besides, we performed some experiments on feature extraction of 2 datasets as shown in Table 7. With the purpose of better understanding the ISE-DSC01 dataset because the ISE-DSC01 dataset is the dataset of a competition that has not had any scientific publication on the analysis of that dataset. The next goal is to analyze the differences between our approach and similarity-based methods based on the extracted features.

4.2. Baseline Models

In our experiments, we use pretrained language models as baselines to ensure feasibility and to compare the results with advanced models. Specifically, we use both multilingual models and language models designed specifically for Vietnamese.

Vietnamese-bi-encoder: The Vietnamese Bi-encoder is a sentence-transformer model that maps text into a 768-dimensional vector space. It is trained on a merged dataset, including MS Macro, SQuAD v2 (both translated to Vietnamese), and 80% of

[1] https://codalab.lisn.upsaclay.fr/competitions/15497

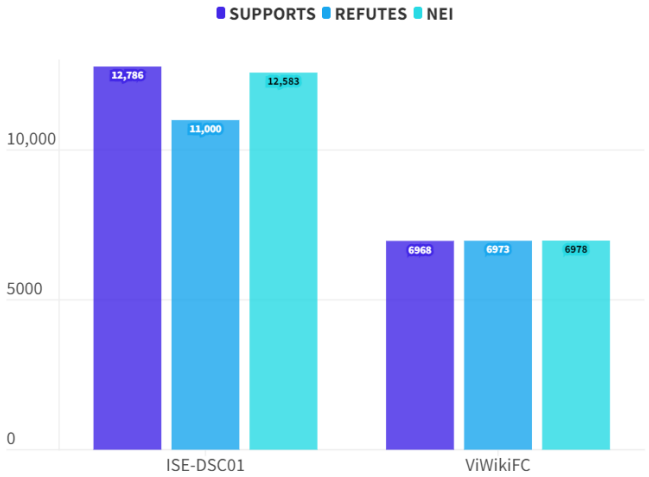

Figure 5: Label distribution of ViWikiFC and ISE-DSC01 datasets.

the Legal Text Retrieval Zalo 2021 challenge data, using $\text{PhoBERT}_{\text{base-v2}}$ as the backbone. The model achieves strong results on the remaining 20% of the Zalo dataset, outperforming Vietnamese-SBERT and $\text{PhoBERT}_{\text{base-v2}}$ across metrics like Acc@1, Acc@10, and MRR@10.

Vietnamese-SBERT: Vietnamese-SBERT is a sentence-BERT model tailored for Vietnamese, developed using PhoBERT as the underlying transformer for Vietnamese token embeddings. It was trained on the Vietnamese Natural Language Inference (NLI) and STS Benchmark (STSb) datasets. The model excels in tasks like sentence paraphrase identification, achieving a high accuracy of 95.33% and an F1-score of 95.42% on the VnPara dataset, outperforming other recent models in Vietnamese NLP tasks.

SIMCSE: SimCSE (Simple Contrastive Learning of Sentence Embeddings) is a model that uses contrastive learning to create high-quality sentence embeddings. It has two modes: unsupervised, where different augmentations of the same sentence are treated as positive pairs, and supervised, using labeled pairs from NLI datasets. SimCSE is effective at capturing semantic similarity, achieving strong performance on benchmark tasks.

Multilingual-E5: Multilingual e5 is a model for generating sentence embeddings across multiple languages using contrastive learning. It is pre-trained on a diverse multilingual dataset, enabling it to handle various languages effectively. The model is particularly strong in cross-lingual tasks like semantic retrieval and classification, making it highly effective for multilingual NLP applications.

ViT5: ViT5 is a Transformer-based encoder-decoder model pre-trained specifically for Vietnamese using T5-style self-supervised learning on a large, diverse corpus. It has been benchmarked on tasks like Abstractive Text Summarization and Named Entity Recognition (NER), consistently outperforming existing models and achieving state-of-the-art results.

ViDeberta: This is a Vietnamese-specific pre-trained language model based on

the DeBERTa architecture, and is trained on a large, high-quality Vietnamese text corpus. ViDeBERTa is fine-tuned and evaluated on three key natural language processing tasks: part-of-speech tagging, named-entity recognition, and question answering. Results show that ViDeBERTa, with fewer parameters, surpasses previous state-of-the-art models in Vietnamese language tasks.

mDeberta-v3: DeBERTaV3 is an enhanced version of the DeBERTa model, replacing mask language modeling (MLM) with replaced token detection (RTD) for more efficient pre-training. It uses a gradient-disentangled embedding sharing method to boost training efficiency and quality. DeBERTaV3 excels in natural language understanding (NLU) tasks, achieving a 91.37% average score on the GLUE benchmark, outperforming DeBERTa and ELECTRA. The multilingual version, mDeBERTa, also shows significant improvements, with mDeBERTa Base achieving 79.8% zero-shot cross-lingual accuracy on XNLI, surpassing XLM-R Base by 3.6%.

XLM-R: XLM-R is a multilingual Transformer-based model built on the RoBERTa architecture, designed to handle 100 languages. It is pre-trained on a massive dataset, leveraging a masked language modeling (MLM) objective to learn contextual representations. XLM-R excels in cross-lingual tasks, consistently outperforming previous multilingual models like mBERT on various benchmarks, including GLUE and XNLI. Its robust performance across languages, including low-resource ones, makes it a powerful tool for multilingual natural language understanding (NLU) tasks.

InfoXLM: InfoXLM extends the XLM-R model by incorporating cross-lingual information extraction capabilities, making it more effective for tasks that require understanding and generating text in multiple languages simultaneously. One of the significant features of InfoXLM is its ability to transfer knowledge from one language to another. This means that if the model is trained on a task in one language, it can perform well on the same task in another language with minimal additional training.

4.3. Experimental Settings

Firstly, for the top k evidence selection task, we experimentally evaluated several models, including TF-IDF, BM25, and various sentence-transformer models such as Vietnamese-SBERT [26], Vietnamese-bi-encoder [27], SimCSE [28], and multilingual-e5 [29]. We then selected the best-performing model to synthesize the context by retrieving the top k sentences most relevant to the claim. After retrieving these sentences, we checked whether they included the original evidence. If the original evidence was missing, we randomly inserted it into the top k sentences and removed one of the retrieved sentences. Finally, the selected sentences were concatenated to form a complete context. We also conducted experiments to optimize the evidence selection process by determining the optimal value of k, considering precision and average context length as the key criteria.

Secondly, for the evidence extraction task, we use pre-trained models such as VIT5 [30], ViDeberta$_{\text{base}}$ [31], mDeberta-v3 [32], XLM-R [33], and InfoXLM [34] to evaluate the performance of the SRC model. We experiment and select common hyperparameters as follows: $batch - size$ = 16, $learning - rate$ = $2e - 5$, $epochs$ = 10, $weight - decay$ = 0.01.

Finally, we trained the SRC-FC model to improve the performance of the evidence extraction task. The SRC-FC model utilizes the XLM-R and InfoXLM model as the

contextual embedding model and has been described in Section 3.3. The following are the hyperparameters to train the SRC-FC model: $batch-size$ = 16, $learning-rate$ = $2e-5$, $epochs$ = 10. During the training of the models mentioned above, we stored and conducted performance evaluations of the model training at each epoch for each task of the SRC-FC model. For ease of comparison with the experimental results of the SRC-FC model, we also conducted the verdict prediction task experiment. The common hyperparameters for the models are as follows: $batch-size$ = 16, $learning-rate$ = $2e-5$, $epochs$ = 10.

4.4. Evaluation Metrics

SRC Model: According to the survey of [18], for the question-answer extraction task, we use the two most common metrics: F1-score and exact match (EM).

- *F1-score*: The harmonic mean of Precision and Recall, commonly used in SRC systems, evaluates the balance between these two metrics. Precision is the ratio of correctly predicted tokens to the total predicted tokens, while Recall is the ratio of correctly predicted tokens to the total ground truth tokens. The F1-score is then calculated as follows:

$$\textit{Precision} = \frac{TP}{TP + FP} \tag{14}$$

$$\textit{Recall} = \frac{TP}{TP + FN} \tag{15}$$

$$\textit{F1} = \frac{2 \times \textit{Precision} \times \textit{Recall}}{\textit{Precision} + \textit{Recall}} \tag{16}$$

  where TP (True Positive) represents the number of correctly predicted words, FP (False Positive) denotes the number of words that were predicted but do not appear in the reference answer and FN (False Negative) is the number of words that appear in the reference answer but were not predicted by the model.

- *Exact Match (EM)*: This is the percentage of answers that exactly match the correct answers. We directly compare the evidence sentence extracted from the SRC model with the ground truth evidence sentence; if they are completely identical (including whitespace, special characters, punctuation, and case), it is considered a match.

VP Model: The VP model is used to classify claim sentences into labels SUPPORTS (SUP), REFUTES (REF), and NEI (Not Enough Information). Therefore, we also use the two most common metrics: *F1-score* and *Accuracy*.

- *F1-score*: F1-score is the harmonic mean of Precision and Recall, commonly used for evaluating classification tasks. It considers both the correctness and completeness of the predictions compared to the ground truth across all classes.

- *Accuracy*: Accuracy measures the proportion of correct predictions out of the total number of predictions made by the model. Evaluating how well the model performs regarding correct classification is a straightforward metric.

Table 2: Performance of models based on claim-evidence similarity on ViWikiFC and ISE-DSC01 datasets (Top 5 candidate evidence) *Note: The "—" symbol indicates that the NEI label in the ISE-DSC01 dataset does not have evidence, so it cannot be evaluated.

Model	ViWiKiFC				ISE-DSC01			
	SUP	REF	NEI	Overall	SUP	REF	NEI	Overall
BM25	92.94	93.63	**72.38**	86.31	77.54	**86.90**	—	**81.87**
TF-IDF	83.67	81.46	54.58	73.24	76.86	85.86	—	81.02
SimCSE	95.06	76.63	66.77	79.49	78.01	67.60	—	73.19
Vietnamese-bi-encoder	96.33	**95.14**	72.08	**87.85**	**78.32**	83.98	—	80.94
Vietnamese-SBERT	94.92	90.08	70.61	85.20	78.17	81.43	—	79.68
Multilingual-e5	**96.47**	91.78	71.34	86.54	76.03	84.32	—	79.86

4.5. Experimental Results

4.5.1. Supervised Reading Comprehension Model for Evidence Extraction

Firstly, we conduct experiments to evaluate similarity-based methods in two benchmark datasets, ViWikiFC and ISE-DSC01, as shown in Table 2. The goal is to select the best model for the top k evidence selection task. We calculate the cosine similarity between the claim sentence and all sentences in the corpus and then rank all candidate sentences from highest to lowest. To assess the accuracy of this method, we check whether the original evidence sentence is among the top *k* candidate sentences. If the original evidence sentence is among the top *k* candidates, the result is 1; otherwise, it is 0. We calculate the accuracy by dividing the number of correctly retrieved evidence sentences by the total number of samples in the test set. The results in Table 2 show that the similarity-based method is still unreliable in accurately selecting evidence, as the original evidence sentences are scattered across larger tops. However, in the top 5 of the ViWikiFC dataset, the results for the SUPPORTS (SUP) and REFUTES (REF) labels are very high. This indicates that the method of calculating cosine similarity between the claim and high-ranking candidate sentences is effective. Thus, combined with the results from Table 2, we will use the Vietnamese-bi-encoder model for the top k evidence selection task, as it provides the highest and most consistent performance, especially on the ViWikiFC dataset.

The next target is to optimize the selection process for the most relevant top k evidence. Based on the results in Table 3 the priority result for k is k=7 in both datasets. We set 0.5% as the threshold to choose the optimal k value. If increasing k by 1 unit does not improve the accuracy by more than 0.5%, then we stop the loop. Because with k=7, we found that it nearly reached the model's threshold for evidence retrieval, as the accuracy in evidence extraction only increased by about 0.3% with k=8, 9, or 10. Additionally, the larger the k value, the larger the aggregated context, which will burden the model and may exceed the model's token limit.

After selecting the optimal pre-trained model and value of k for the evidence selection task, we create the training data as mentioned in Section 3.2. We trained the SRC model on the training set, which had been enriched with new context, and evaluated the SRC model on the test set, obtaining the results shown in Table 4. Since the ISE-DSC01 dataset does not contain evidence for claims labeled as NEI, the NEI label in this dataset is only evaluated using the EM score. Overall, the SRC model demonstrated good performance on the SUPPORTES and REFUTES labels of the ViWikiFC

Table 3: Results of the top *k* evidence selection task using the Vietnamese-bi-encoder model for k from 1 to 10 (By calculating the cosine similarity between the claim sentence and each sentence in the corpus)

k	ViWikiFC		ISE-DSC01	
	Avg. context lenght	**Acc (%)**	**Avg. context lenght**	**Acc (%)**
1	28.74	74.05	24.09	71.56
2	57.06	81.93	48.83	76.59
3	86.24	84.96	73.70	78.93
4	114.41	86.75	98.23	79.98
5	143.17	87.86	123.07	80.88
6	171.98	88.69	147.66	81.70
7	200.77	89.57	172.16	82.52
8	229.53	90.05	196.81	82.89
9	258.56	90.39	221.17	83.37
10	287.19	90.82	246.09	83.59

dataset and the NEI label of the ISE-DSC01 dataset, with some models achieving over 90% (EM). Notably, InfoXLM and XLM-R models produced the best results on both datasets regarding EM score. Despite being a multilingual model, InfoXLM and XLM-R outperformed models specifically trained for Vietnamese, such as ViT5 and ViDeberta on ViWikiFC dataset. When considering the results for each label, we found that the NEI label of the ViWikiFC dataset yielded significantly lower results than the other two labels, by about 10% or more on both F1 and EM metrics. This suggests that the SRC model struggles to identify evidence for claims with the NEI label accurately. It could be that these claims, during their creation in the ViWikiFC task, contain relatively little information related to the original evidence. In contrast, the NEI claims in the ISE-DSC01 dataset only need to determine whether there is evidence in the context. Therefore, the results for the NEI label are quite good and do not differ significantly from the other two labels.

We found that on the ViWikiFC dataset, the ViT5 model achieved an F1 score comparable to the best model, XLM-R. However, its performance in terms of Exact Match (EM) was significantly lower. This discrepancy is likely due to the ViWikiFC dataset containing numerous special characters that ViT5 struggles to encode effectively. ViT5 is primarily trained on Vietnamese data, whereas XLM-R is a multilingual model trained on over 2 terabytes of data from 100 languages, allowing it to better handle diverse textual elements. In contrast, the ISE-DSC01 dataset, derived from Vietnamese newspapers, lacks the same volume of special characters, leading to more consistent results across models.

4.5.2. SRC-FC Model Training Process

Before presenting the SRC-FC model results on the test sets for the ViWikiFC and ISE-DSC01 datasets, we provide the metrics obtained during the training process of the SRC-FC model. This includes the variation in loss values during training and the accuracy of the tasks on the validation set across each epoch. These are illustrated in the following charts.

Table 4: Fine-tuning results of the SRC models based on *F1* and *Exact Match (EM)* scores. *Note: The F1 score for the NEI label in the ISE-DSC01 dataset is marked as "—" since this label does not have corresponding evidence, making the F1 evaluation inapplicable.

Model	ViWikiFC								ISE-DSC01							
	F1 (%)				EM (%)				F1 (%)				EM (%)			
	SUP	REF	NEI	Overall	SUP	REF	NEI	Overall	SUP	REF	NEI	Overall	SUP	REF	NEI	Overall
$ViT5_{base}$	96.15	**96.24**	82.54	91.64	87.15	87.11	72.97	82.41	83.94	88.52	—	86.05	81.36	87.34	89.13	85.85
$Videberta_{base}$	93.77	94.72	76.60	88.36	90.40	90.65	72.08	84.38	78.02	86.54	—	81.96	75.09	84.39	76.38	78.34
$mDeberta_{base}$	83.39	84.23	68.79	78.80	73.73	76.63	60.12	70.16	82.91	88.72	—	85.60	81.00	87.28	85.84	84.57
$XLM\text{-}R_{base}$	96.27	95.79	81.84	91.30	**94.92**	**93.48**	78.14	**88.85**	76.04	81.73	—	78.67	73.94	80.21	**93.10**	82.47
$XLM\text{-}R_{large}$	**97.25**	95.68	**83.98**	**92.30**	94.63	91.93	**79.62**	88.72	85.49	**89.54**	—	87.36	83.76	**88.20**	90.33	87.37
$InfoXLM_{base}$	96.76	95.94	81.58	91.57	94.35	93.20	76.96	88.33	82.06	84.01	—	82.96	79.63	82.97	87.67	83.42
$InfoXLM_{large}$	96.02	94.99	81.16	90.86	92.66	91.36	74.00	86.18	**86.31**	88.73	—	**87.43**	**83.97**	87.34	90.96	**87.41**

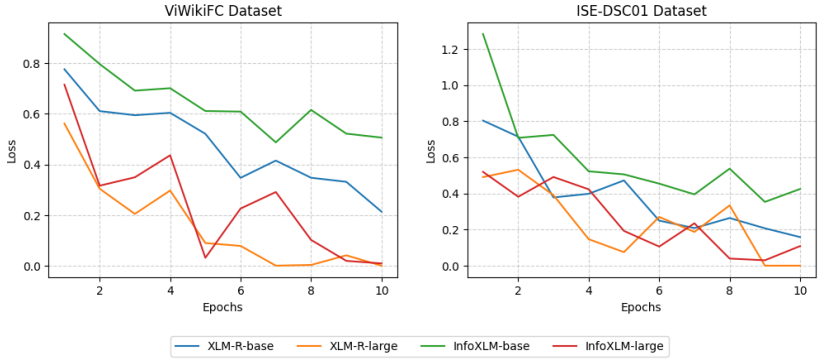

Figure 6: Loss value per epoch for the SRC-FC model on the ViWikiFC and ISE-DSC01 datasets

First, Figure 6 shows the loss values of the SRC-FC model based on pretrained models across epochs. Overall, the training process on both the ViWikiFC and ISE-DSC01 datasets shows a fairly stable decrease in loss, gradually converging with each epoch. This indicates that the model is learning and optimizing well during training. For ViWikiFC, a sharp decrease occurs right from the first few epochs (especially from epoch 1 to epoch 3). For ISE-DSC01, a similar sharp decrease is observed in the early epochs, though at a slightly slower pace compared to ViWikiFC. As for the individual models, both $XLM\text{-}R_{large}$ and $InfoXLM_{large}$ demonstrated superior performance on both datasets, with the loss decreasing rapidly and converging well after 8 to 10 epochs. This suggests that using the large versions of XLM-R and InfoXLM allows for deeper and better learning for multitask fact-checking tasks. As for the base versions of XLM-R and InfoXLM, they show a slower decrease in loss. This is understandable, as these models are smaller and cannot capture complex contextual features and data as effectively as the large versions. Additionally, we can observe differences in model performance between the two datasets. ViWikiFC and ISE-DSC01 differ in structure and size, and thus, the variation in loss values between the two graphs reflects the complexity of each dataset. For ISE-DSC01, all models achieved lower loss compared to ViWikiFC, suggesting that the ISE-DSC01 data may be easier to learn or better structured for tasks in the SRC-FC model. On ViWikiFC, although the loss tends to decrease with each epoch, the reduction is not uniform, with spikes occurring at certain epochs, particularly with the InfoXLM-large model. This could be due to the complexity of the

context or heterogeneity in the ViWikiFC dataset.

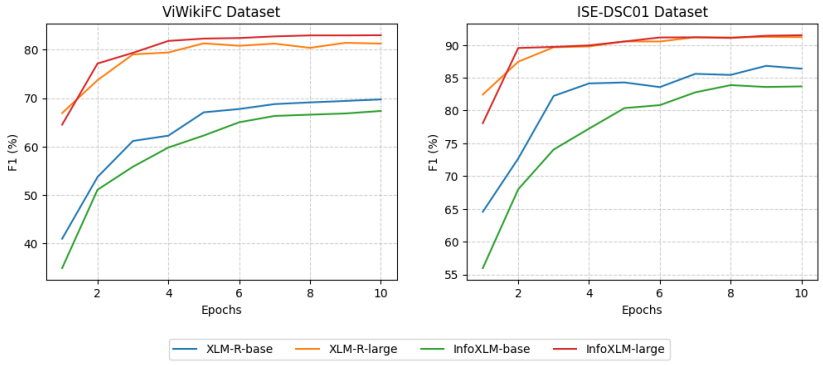

Figure 7: F1-score per epoch for Verdict Prediction of the SRC-FC Model on the ViWikiFC and ISE-DSC01 datasets

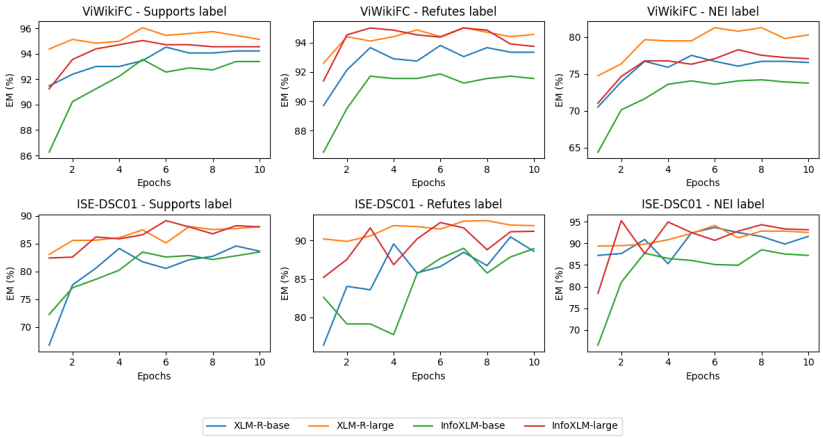

Figure 8: EM-score per epoch for the evidence extraction task of the SRC-FC model on the ViWikiFC and ISE-DSC01 datasets

Second, based on the F1-score chart across epochs for the Verdict Prediction task in Figure 7, we can draw some insightful analyses regarding the performance of the SRC-FC models on the two datasets, ViWikiFC and ISE-DSC01. Both datasets reveal that the model quickly reaches its performance threshold, with the model struggling to improve accuracy for the VP task after approximately 5 epochs. Subsequently, the F1-score tends to stabilize in the final epochs, indicating that the model has nearly reached the point of convergence. Additionally, the performance of the large versions of XLM-R and InfoXLM continued to demonstrate superiority over the base versions, with high F1-scores and fast convergence on both datasets. On ViWikiFC, the F1-scores of XLM-R_{large} and InfoXLM$_{large}$ reached nearly 80% by epoch 3 and stabilized above 80% after epoch 4. On ISE-DSC01, these models quickly achieved F1-scores above 90% after just

2 to 3 epochs and maintained this level until the end of training. Apart from the differences between models, the datasets also show significant distinctions. ISE-DSC01 has a higher F1-score compared to ViWikiFC across all models, suggesting that the ISE-DSC01 data may be less complex or easier to predict. The large models ($\text{InfoXLM}_{\text{large}}$ and $\text{XLM-R}_{\text{large}}$) achieved very high F1 scores, above 90%, after the initial epochs. This reflects the strong learning ability of the model on the ISE-DSC01 dataset.

Next, based on the EM-score (Exact Match) chart across epochs for the evidence extraction task of each label (SUPPORTS, REFUTES, NEI) in the two datasets, ViWikiFC and ISE-DSC01, as shown in Figure 8. For the SUPPORTS label, both the ViWikiFC and ISE-DSC01 datasets show consistently high EM-scores for the models after the initial epochs. On ViWikiFC, the $\text{InfoXLM}_{\text{large}}$ model achieves around 95% (EM) and maintains this throughout training. On ISE-DSC01, it holds steady at 86-88% (EM) after the fourth epoch. In contrast, while the REFUTES label on ISE-DSC01 also delivers strong results, the model struggles with consistent accuracy in pinpointing the evidence. The NEI label presents more challenges for evidence extraction on ViWikiFC, as claims often lack enough information, with the $\text{InfoXLM}_{\text{large}}$ model reaching just above 80% (EM). However, on ISE-DSC01, the NEI label only requires detecting if the context contains evidence, resulting in very high EM-scores of 90-95% (EM).

From the results and evaluations of each task during the training process of the SRC-FC model, multi-task learning approach has demonstrated its performance by simultaneously learning evidence extraction and verdict prediction. This joint learning helps the model leverage the correlation between tasks, resulting in better loss reduction and faster convergence, especially in larger models such as $\text{InfoXLM}_{\text{large}}$. The higher F1 and EM scores across datasets show that the model effectively utilizes multi-task learning to extract evidence and predict verdicts more accurately. Additionally, we can observe the impact of factors such as the data and the type of model used on the performance achieved across tasks.

4.5.3. Our Proposed Model

After training the SRC-FC model in Section 4.5.2, we evaluated the model on the test set and compared the results with the SRC model on the evidence extraction task and the VP model on the verdict prediction task, obtaining the results shown in Table 5. For the ViWikiFC dataset, we observed an improvement in performance for the evidence extraction task. Compared to the separately trained SRC model, the $\text{SRC-FC}_{\text{infoXLM-large}}$ model results in a 2.52% (F1) and 4.06% (EM) increase across all three labels. But for the VP task, the SRC-FC model yielded significantly lower results compared to the VP model trained separately, specifically 4.33% lower with $\text{XLM-R}_{\text{large}}$ and 4.31% with $\text{InfoXLM}_{\text{large}}$. However, this lower performance in the VP task is not a major concern for us, since our primary goal is to enhance the performance of the evidence extraction task. Overall, with the ViWikiFC dataset, the results for the evidence extraction task are not particularly outstanding, partly because the accuracy of evidence retrieval for claims in the ViWikiFC dataset may have already reached its limit, as the results for the SUPPORTS and REFUTES labels are very high, above 95%, when training the SRC model alone. Therefore, the SRC-FC model's performance improvement of 4.06% (EM) is still quite significant. For the ISE-DSC01 dataset, the results exceeded expectations for the evidence extraction task, with an increase in F1

Table 5: Results of the SRC-FC model on evidence extraction and verdict prediction task. *Note: The F1 score for the NEI label in the ISE-DSC01 dataset is marked as "—" since this label does not have corresponding evidence, making F1 evaluation inapplicable. Up (↑) and down (↓) arrows represent the increase or decrease in performance compared to the corresponding single-task models on each task.

Datasets	**Model**	**VP**		**SRC**							
		Acc (%)	**F1 (%)**	**F1 (%)**				**EM (%)**			
				SUP	REF	NEI	Overall	SUP	REF	NEI	Overall
ViWikiFC	SRC-FC$_{\text{xlmr-base}}$	67.48 (↓ 11.05)	67.74 (↓ 10.82)	97.74 (↑ 1.47)	96.63 (↑ 0.84)	82.86 (↑ 1.02)	92.41 (↑ 1.11)	94.51 (↑ 0.32)	93.80 (↑ 0.32)	76.71 (↓ 1.43)	88.34 (↓ 0.51)
	SRC-FC$_{\text{xlmr-large}}$	80.79 (↓ 4.89)	80.82 (↓ 4.33)	97.75 (↑ 0.5)	96.80 (↑ 1.12)	86.21 (↑ 2.23)	93.59 (↑ 1.29)	95.43 (↑ 0.80)	94.40 (↑ 2.47)	81.27 (↑ 1.65)	90.37 (↑ 1.65)
	SRC-FC$_{\text{infoxlm-base}}$	66.60 (↓ 12.59)	66.70 (↓ 12.53)	97.52 (↑ 0.82)	95.64 (↓ 0.30)	83.84 (↑ 2.26)	92.33 (↑ 0.76)	94.82 (↑ 0.47)	91.98 (↓ 1.22)	78.18 (↑ 1.22)	88.35 (↑ 0.02)
	SRC-FC$_{\text{infoxlm-large}}$	82.19 (↓ 4.31)	82.20 (↓ 4.31)	98.48 (↑ 2.46)	96.55 (↑ 1.56)	85.12 (↑ 3.96)	93.38 (↑ 2.52)	96.34 (↑ 3.68)	94.10 (↑ 2.74)	80.29 (↑ 6.29)	90.24 (↑ 4.06)
ISE-DSC01	SRC-FC$_{\text{xlmr-base}}$	86.64 (↑ 0.96)	86.65 (↑ 0.89)	86.40 (↑ 10.36)	91.10 (↑ 9.37)	–	88.57 (↑ 9.90)	83.66 (↑ 9.72)	89.42 (↑ 9.21)	89.27 (↓ 3.83)	87.34 (↑ 4.87)
	SRC-FC$_{\text{xlmr-large}}$	90.36 (↑ 1.31)	90.33 (↑ 1.22)	91.43 (↑ 5.94)	93.19 (↑ 3.65)	–	92.24 (↑ 4.88)	89.69 (↑ 5.93)	92.07 (↑ 3.87)	90.45 (↑ 0.12)	90.67 (↑ 3.30)
	SRC-FC$_{\text{infoxlm-base}}$	84.10 (↑ 0.29)	83.96 (↑ 0.80)	88.67 (↑ 6.61)	91.12 (↑ 7.11)	–	89.08 (↑ 6.84)	84.03 (↑ 4.40)	89.10 (↑ 6.13)	87.72 (↑ 0.05)	86.84 (↑ 3.42)
	SRC-FC$_{\text{infoxlm-large}}$	91.82 (↑ 4.82)	91.82 (↑ 4.33)	90.74 (↑ 4.43)	93.91 (↑ 5.18)	–	92.20 (↑ 4.77)	88.90 (↑ 4.93)	92.88 (↑ 5.54)	92.01 (↑ 1.05)	91.18 (↑ 3.77)

score by up to 10.36% for the SUPPORTS label and 9.37% for the REFUTES label when using SRC-FC with XLM-R$_{\text{base}}$. Additionally, SRC-FC with XLM-R$_{\text{large}}$ showed an increase of 5.94% and 3.65% in the F1 score for the SUPPORTS and REFUTES labels, respectively. However, for the NEI label, the results did not show significant differences compared to the standalone model. This may be due to the NEI label being relatively easy to predict based on the presence of evidence in the passage, which allowed the standalone SRC model to perform well, leading to minimal improvement when applying SRC-FC. Nevertheless, the SRC-FC model demonstrated superiority for the SUPPORTS and REFUTES labels, where the standalone SRC model struggled to accurately identify the evidence's location within the passage. This was likely due to the additional features learned from the VP task, enabling SRC-FC to more effectively pinpoint the correct evidence corresponding to the claim's label. This demonstrates the effectiveness of multi-task learning in enhancing certain aspects of fact-checking tasks.

To further verify the effectiveness of the SRC and SRC-FC models in real-world evidence retrieval, we applied them to the pipeline proposed in Section 3.2, with the results presented in Table 6. The results in Table 6 were obtained by using the Vietnamese-bi-encoder model to select the top candidate evidence (k=7), then concatenating them and feeding them into the SRC or SRC-FC models to predict the start and end positions of the evidence. Overall, the results on the ViWikiFC dataset are quite high for both the SRC model and our proposed SRC-FC model. The best model, XLM-R$_{\text{large}}$ of SRC-FC, achieved 88.21% (F1 score) and 83.66% (EM). However, when using the same pre-trained model, there is not a significant difference between SRC and SRC-FC. This can be explained by the limitation of the top k evidence selection step, which only retrieves easily extractable evidence, while difficult-to-retrieve original evidence is missed and does not appear in the top k evidence. Additionally, when comparing the performance on the test set, SRC-FC only outperformed SRC by around 1-2%. Therefore, the SRC-FC model does not show superiority over the SRC model when using context that already contains the original evidence. As for the ISE-DSC01 dataset, the results for both SRC and SRC-FC models were also good. The best SRC-FC model

Table 6: Results of our proposed method on the evidence retrieval pipeline based on *F1* and *EM* scores. *Note: The F1 score for the NEI label in the ISE-DSC01 dataset is marked as "—" since this label does not have corresponding evidence, making the F1 evaluation inapplicable.

Model		ViWikiFC								ISE-DSC01							
		F1 (%)				EM (%)				F1 (%)				EM (%)			
		SUP	REF	NEI	Overall	SUP	REF	NEI	Overall	SUP	REF	NEI	Overall	SUP	REF	NEI	Overall
SRC-FC model	XLM-R_{base}	95.60	93.95	71.14	86.90	91.81	89.94	63.22	81.66	73.90	84.02	—	78.58	70.18	81.01	88.56	79.81
	XLM-R_{large}	**97.09**	**94.91**	**72.63**	**88.21**	**94.49**	**91.36**	65.13	**83.66**	76.73	**85.77**	—	80.91	72.95	**83.10**	91.64	**82.48**
	InfoXLM$_{base}$	95.12	92.59	69.64	85.78	91.38	88.95	61.60	80.64	74.51	83.34	—	78.59	70.29	80.09	86.73	78.94
	InfoXLM$_{large}$	95.94	92.78	70.42	86.38	93.08	89.53	62.19	81.60	**78.18**	85.64	—	**81.62**	**73.84**	82.73	90.60	82.32
SRC model	ViT5$_{base}$	95.01	93.23	70.81	86.35	84.46	83.99	60.86	76.44	74.20	82.21	—	77.90	70.34	79.90	89.08	79.71
	Videberta$_{base}$	91.75	91.74	68.06	83.85	88.56	88.10	60.71	79.12	72.40	80.89	—	76.33	68.62	78.18	75.97	74.05
	mDeberta$_{base}$	82.67	82.18	59.28	74.71	75.99	74.36	51.26	67.20	74.39	82.88	—	78.31	71.91	80.95	85.63	79.39
	XLM-R_{base}	95.53	92.32	70.24	86.03	93.64	90.51	65.13	83.10	66.64	75.90	—	70.92	63.92	73.57	**92.63**	76.77
	XLM-R_{large}	95.19	93.53	70.37	86.36	92.51	89.38	**65.14**	82.34	75.20	83.44	—	79.01	72.17	81.07	90.39	81.16
	InfoXLM$_{base}$	95.54	93.98	70.23	86.82	93.08	90.79	63.07	82.59	70.68	77.36	—	73.77	67.15	75.35	87.04	76.51
	InfoXLM$_{large}$	95.60	92.33	70.43	86.35	92.09	88.24	62.63	81.25	73.77	81.33	—	77.27	70.44	78.86	90.44	79.91

was InfoXLM$_{\text{large}}$ with 81.62% (F1) and XLM-R$_{\text{large}}$ with 82.48% (EM). In contrast to ViWikiFC, the performance of SRC-FC was significantly better than SRC when comparing the models using the same pre-trained model. The ISE-DSC01 dataset has many features that the SRC model did not fully use. However, the SRC-FC model showed the benefits of multitask learning on the test set by increasing F1 and EM scores by about 3-4%. Therefore, applying it to the extraction task in the pipeline would also yield much better performance compared to the SRC model.

In conclusion, our experiments demonstrate that the SRC-FC model shows significant potential, particularly in enhancing evidence extraction performance. For the ViWikiFC dataset, the model achieved notable improvements in both F1 and EM scores, despite the limitations of evidence retrieval in the top-k selection process. While the performance gains in verdict prediction were lower, this was expected, as our primary focus was on improving evidence extraction. On the ISE-DSC01 dataset, the results are even more promising, with the SRC-FC model significantly outperforming the SRC model, particularly for the SUPPORTS and REFUTES labels. The observed improvements highlight the effectiveness of multitask learning, especially in tasks requiring complex evidence location. Overall, our findings underscore the potential of the SRC-FC model for fact-checking tasks, though further exploration of evidence retrieval strategies could yield even better results.

4.5.4. Evidence-Claim Analysis

From the SRC and SRC-FC model results, we observe differences in the performance for the SUPPORTS, REFUTES, and NEI labels across both datasets. In the ViWikiFC dataset, the result for the SUPPORTS and REFUTES labels is comparable and significantly higher than for the NEI label. However, in the ISE-DSC01 dataset, the NEI label shows the most favorable results, with considerable discrepancies in the performance across all three labels. Therefore, we conducted several experiments on the evidence-claim pairs for each label to elucidate the reasons for these differences. We analyzed the relevance between the evidence and claim sentences based on the New word rate measure and some measures presented by Abdalla et al. (2023) [35]. These methods have been shown by Abdalla et al. (2023) [35] to be highly correlated to the relevance between two sentences. The metrics we considered include:

Related Words - all: For this metric, we use a Word2Vec model to find embeddings for all tokens in the sentence. We then sum the embedding vectors and average

Table 7: Analyzing the Characteristics of Two Benchmark Datasets ViWikiFC and ISE-DSC01. *Note: The "—" symbol indicates that the NEI label in the ISE-DSC01 dataset does not have evidence, so it cannot be evaluated.

Measure	ViWikiFC			ISE-DSC01		
	SUP	REF	NEI	SUP	REF	NEI
Lexical Overlap	0.3825	0.3752	0.2507	0.4639	0.6020	–
New word rate	0.3330	0.3181	0.5142	0.3439	0.2348	–
Sentence Transformer	0.7385	0.7004	0.6281	0.7881	0.7901	–
Related Words - all	0.9767	0.9731	0.9599	0.9571	0.9800	–
Related Words - PROPN	0.6619	0.6548	0.5672	0.3610	0.3876	–
Related Words - NOUN group	0.8956	0.8805	0.8337	0.8654	0.9201	–

them to obtain the sentence embedding. Once we have the embeddings for both the evidence and claim sentences, we use cosine similarity to measure their similarity. The Word2Vec model used is PhoW2V [36] with an embedding size of 300. This Word2Vec model was pre-trained on a 20GB corpus of Vietnamese texts.

Related Words - PROPN: The process is similar to Related Words - all, but only tokens with the POS tag Np (Proper noun) are used to calculate the sentence embedding. We use VnCoreNLP [37, 38, 39] to find POS tags for the tokens in the sentence.

Related Words - NOUN group: The process is similar to Related Words - PROPN, but only tokens with POS tags within the NOUN group are used to calculate the sentence embedding. The NOUN group includes Np (Proper noun), Nc (Classifier noun), Nu (Unit noun), N (Noun), Ny (Abbreviated noun), Nb ((Foreign) borrowed noun).

Lexical Overlap: Lexical overlapping between two sentences refers to the extent to which they share common words. We use the Jaccard similarity to calculate the overlap ratio.

New Token Rate: We measure the proportion of unique tokens in the claim sentence that are not found in the evidence sentence. We first tokenize both sentences, convert them to lowercase, and count the tokens in the claim that do not exist in the evidence, finally returning this count as a fraction of the total number of tokens in the claim.

Sentence Transformer: For this metric, we compute the sentence embeddings for both the evidence and claim sentences using a Sentence Transformer model, then use cosine similarity to measure their similarity. The Sentence Transformer model used is the vietnamese-bi-encoder.

From Table 7, we observe a correlation between the evidence-claim analysis measure and the performance of the SRC and SRC-FC model, especially for the NEI label in the ViWikiFC dataset and the SUPPORTS label in the ISE-DSC01 dataset. Specifically, labels with lower performances also have lower evidence-claim analysis measures and vice versa; except for the New word rate measure, as a higher New word rate indicates more new words appearing in the claim sentence compared to the evidence sentence, making these samples more challenging than others, thus leading to a decrease in performance. Similarly, low lexical overlap and Sentence Transformer measures indicate challenging samples.

Moreover, for the NEI label in the ViWikiFC dataset, we notice that annotators use

more new nouns compared to the nouns present in the evidence sentence to create the claim sentence (these nouns have embeddings that are "further apart" from existing noun in the evidence sentence, leading to the difference between the NEI label and SUPPORTS-REFUTES labels in Related Words measures), similar to the SUPPORTS label and the REFUTES label in the ISE-DSC01 dataset.

5. Conclusion and Future Work

We introduced novel approaches to address the Vietnamese fact-checking task. By leveraging a supervised reading comprehension model, we achieved promising results in evidence retrieval, which enhanced the model's ability to predict claim labels accurately. Our analysis of the relationships between the evidence and the claim sentence in the datasets provided valuable insights into performance variations across different labels. The experimental results demonstrated the effectiveness of SRC-FC models once again, reinforcing their potential as a robust approach for tackling complex deep-learning tasks. Moving forward, we plan to expand our research to diverse datasets, particularly those involving claims that relate to multiple evidence sentences. This expansion aims to enhance the practicality and real-world applicability of our fact-checking model. Furthermore, we intend to explore and experiment with various architectures for multi-task learning models to optimize their performance. We hope our research findings will pave the way for new studies in the fact-checking task, particularly for the Vietnamese language.

Although we achieved promising results on the ViWikiFC and ISE-DSC01 datasets using the SRC-FC model, in reality, a claim may relate to multiple different evidence sentences, not only a single evidence sentence. Therefore, future work will focus on addressing the challenge of handling claims associated with multiple evidence sentences, aiming to enhance the robustness and generalizability of the proposed models. For claim sentences containing information combined from multiple evidence sentences, we intend to apply some algorithms to split the original claim into several smaller claims to facilitate processing.

Moreover, SRC-FC architecture is quite diverse and can be implemented in various ways, such as selecting the loss function, optimization function, and the strategy to combine vector embeddings. We believe that the performance of the models can be further improved by researching and choosing the most appropriate multi-task learning architecture for the fact-checking task.

Limitations

Through the results presented in Section 4.5, the positive aspects of our proposed approach using the SRC model for evidence extraction and enhancing the performance of the SRC model have been demonstrated. Alongside these positive outcomes, our research still has some limitations regarding experimental data, as we only used two benchmark datasets. This limitation means that certain other aspects of data have not yet been experimented on using our proposed method. Additionally, we omitted a component in our proposed ER pipeline—document retrieval. While this simplification

might make the experimental demonstration of applying SRC to evidence extraction easier, it reduces the generalizability of the method when applied in real-world scenarios. Furthermore, a critical component still needs to be improved in our Evidence Retrieval pipeline, which is the stage of using Sentence-BERT or BM25 to retrieve the top k sentences most relevant to the claim sentence. If this component fails to retrieve the evidence sentence within the top k evidence candidate, then using the SRC model or the SRC-FC model to extract the evidence sentence becomes meaningless, as the input to the model will not contain the evidence sentence from the start.

CRediT authorship contribution statement

Manh Trong Nguyen: Conceptualization; Formal analysis; Investigation; Methodology; Validation; Visualization; Writing - original draft. **Tri Thien Nguyen**: Conceptualization; Formal analysis; Investigation; Methodology; Validation; Visualization; Writing - original draft. **Kiet Van Nguyen**: Conceptualization; Formal analysis; Investigation; Methodology; Validation; Supervision; Writing - review & editing.

Declaration of Interest

The authors declare that they have no conflict of interest.

Acknowledgement

This research was supported by The VNUHCM-University of Information Technology's Scientific Research Support Fund.

References

[1] N. Hassan, B. Adair, J. T. Hamilton, C. Li, M. Tremayne, J. Yang, C. Yu, The quest to automate fact-checking, in: Proceedings of the 2015 computation+ journalism symposium, Citeseer.

[2] N. Hassan, G. Zhang, F. Arslan, J. Caraballo, D. Jimenez, S. Gawsane, S. Hasan, M. Joseph, A. Kulkarni, A. K. Nayak, et al., Claimbuster: The first-ever end-to-end fact-checking system, Proceedings of the VLDB Endowment 10 (2017) 1945–1948.

[3] Z. Guo, M. Schlichtkrull, A. Vlachos, A survey on automated fact-checking, Transactions of the Association for Computational Linguistics 10 (2022) 178–206.

[4] J. Vladika, F. Matthes, Scientific fact-checking: A survey of resources and approaches, in: Findings of the Association for Computational Linguistics: ACL 2023, pp. 6215–6230.

[5] J. Thorne, A. Vlachos, C. Christodoulopoulos, A. Mittal, FEVER: a large-scale dataset for fact extraction and VERification, in: M. Walker, H. Ji, A. Stent (Eds.), 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, New Orleans, Louisiana, 2018, pp. 809–819.

[6] D. Wadden, K. Lo, L. L. Wang, A. Cohan, I. Beltagy, H. Hajishirzi, Multivers: Improving scientific claim verification with weak supervision and full-document context, in: Findings of the Association for Computational Linguistics: NAACL 2022, pp. 61–76.

[7] A. Wührl, R. Klinger, Entity-based claim representation improves fact-checking of medical content in tweets, in: G. Lapesa, J. Schneider, Y. Jo, S. Saha (Eds.), Proceedings of the 9th Workshop on Argument Mining, International Conference on Computational Linguistics, Online and in Gyeongju, Republic of Korea, 2022, pp. 187–198.

[8] N. Kotonya, F. Toni, Explainable automated fact-checking for public health claims, in: B. Webber, T. Cohn, Y. He, Y. Liu (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, Online, 2020, pp. 7740–7754.

[9] A. Saakyan, T. Chakrabarty, S. Muresan, COVID-fact: Fact extraction and verification of real-world claims on COVID-19 pandemic, in: C. Zong, F. Xia, W. Li, R. Navigli (Eds.), Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Association for Computational Linguistics, Online, 2021, pp. 2116–2129.

[10] X. Li, G. A. Burns, N. Peng, A paragraph-level multi-task learning model for scientific fact-verification., in: SDU@ AAAI.

[11] Z. Zhang, J. Li, F. Fukumoto, Y. Ye, Abstract, rationale, stance: A joint model for scientific claim verification, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 3580–3586.

[12] D. Wadden, S. Lin, K. Lo, L. L. Wang, M. van Zuylen, A. Cohan, H. Hajishirzi, Fact or fiction: Verifying scientific claims, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 7534–7550.

[13] R. Pradeep, X. Ma, R. Nogueira, J. Lin, Scientific claim verification with vert5erini, in: Proceedings of the 12th International Workshop on Health Text Mining and Information Analysis, pp. 94–103.

[14] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, Journal of machine learning research 21 (2020) 1–67.

[15] N. Reimers, I. Gurevych, Sentence-bert: Sentence embeddings using siamese bert-networks, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 3982–3992.

[16] C. Samarinas, W. Hsu, M. L. Lee, Improving evidence retrieval for automated explainable fact-checking, in: A. Sil, X. V. Lin (Eds.), Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Demonstrations, Association for Computational Linguistics, Online, 2021, pp. 84–91.

[17] K. Lu, I.-H. Hsu, W. Zhou, M. D. Ma, M. Chen, Multi-hop evidence retrieval for cross-document relation extraction, in: A. Rogers, J. Boyd-Graber, N. Okazaki (Eds.), Findings of the Association for Computational Linguistics: ACL 2023, Association for Computational Linguistics, Toronto, Canada, 2023, pp. 10336–10351.

[18] R. Baradaran, R. Ghiasi, H. Amirkhani, A survey on machine reading comprehension systems, Natural Language Engineering 28 (2022) 683–732.

[19] Y. Cui, T. Liu, Z. Yang, Z. Chen, W. Ma, W. Che, S. Wang, G. Hu, A sentence cloze dataset for chinese machine reading comprehension, arXiv preprint arXiv:2004.03116 (2020).

[20] Y. Yang, W.-t. Yih, C. Meek, Wikiqa: A challenge dataset for open-domain question answering, in: Proceedings of the 2015 conference on empirical methods in natural language processing, pp. 2013–2018.

[21] H. Gharagozlou, J. Mohammadzadeh, A. Bastanfard, S. S. Ghidary, Rlas-biabc: A reinforcement learning-based answer selection using the bert model boosted by an improved abc algorithm, Computational Intelligence and Neuroscience 2022 (2022) 7839840.

[22] P. N.-T. Do, N. D. Nguyen, T. Van Huynh, K. Van Nguyen, A. G.-T. Nguyen, N. L.-T. Nguyen, Sentence extraction-based machine reading comprehension for vietnamese, in: Knowledge Science, Engineering and Management: 14th International Conference, KSEM 2021, Tokyo, Japan, August 14–16, 2021, Proceedings, Part II 14, Springer, pp. 511–523.

[23] S. Li, S. Zhao, B. Cheng, H. Yang, An end-to-end multi-task learning model for fact checking, in: J. Thorne, A. Vlachos, O. Cocarascu, C. Christodoulopoulos, A. Mittal (Eds.), Proceedings of the First Workshop on Fact Extraction and VERification (FEVER), Association for Computational Linguistics, Brussels, Belgium, 2018, pp. 138–144.

[24] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. Le Scao, S. Gugger, M. Drame, Q. Lhoest, A. Rush, Transformers: State-of-the-art natural language processing, in: Q. Liu,

D. Schlangen (Eds.), Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Online, 2020, pp. 38–45.

[25] H. T. Le, L. T. To, M. T. Nguyen, K. Van Nguyen, Viwikifc: Fact-checking for vietnamese wikipedia-based textual knowledge source, arXiv preprint arXiv:2405.07615 (2024).

[26] Q. L. Phan, T. H. P. Doan, N. H. Le, N. B. D. Tran, T. N. Huynh, Vietnamese sentence paraphrase identification using sentence-bert and phobert, in: N.-T. Nguyen, N.-N. Dao, Q.-D. Pham, H. A. Le (Eds.), Intelligence of Things: Technologies and Applications, Springer International Publishing, Cham, 2022, pp. 416–423.

[27] N. Q. Duc, L. H. Son, N. D. Nhan, N. D. N. Minh, L. T. Huong, D. V. Sang, Towards comprehensive vietnamese retrieval-augmented generation and large language models, arXiv preprint arXiv:2403.01616 (2024).

[28] T. Gao, X. Yao, D. Chen, Simcse: Simple contrastive learning of sentence embeddings, in: EMNLP 2021-2021 Conference on Empirical Methods in Natural Language Processing, Proceedings.

[29] L. Wang, N. Yang, X. Huang, L. Yang, R. Majumder, F. Wei, Multilingual e5 text embeddings: A technical report, arXiv preprint arXiv:2402.05672 (2024).

[30] L. Phan, H. Tran, H. Nguyen, 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, pp. 136–142.

[31] C. D. Tran, N. H. Pham, A. T. Nguyen, T. S. Hy, T. Vu, Videberta: A powerful pre-trained language model for vietnamese, in: Findings of the Association for Computational Linguistics: EACL 2023, pp. 1071–1078.

[32] P. He, J. Gao, W. Chen, Debertav3: Improving deberta using electra-style pretraining with gradient-disentangled embedding sharing, in: The Eleventh International Conference on Learning Representations.

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

[34] Z. Chi, L. Dong, F. Wei, N. Yang, S. Singhal, W. Wang, X. Song, X.-L. Mao, H. Huang, M. Zhou, InfoXLM: An information-theoretic framework for cross-lingual language model pre-training, in: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Association for Computational Linguistics, Online, 2021, pp. 3576–3588.

[35] M. Abdalla, K. Vishnubhotla, S. Mohammad, What makes sentences semantically related? a textual relatedness dataset and empirical study, in: Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics, pp. 782–796.

[36] A. T. Nguyen, M. H. Dao, D. Q. Nguyen, A pilot study of text-to-sql semantic parsing for vietnamese, in: Findings of the Association for Computational Linguistics: EMNLP 2020, pp. 4079–4085.

[37] T. Vu, D. Q. Nguyen, M. Dras, M. Johnson, et al., 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, pp. 56–60.

[38] D. Q. Nguyen, T. Vu, M. Dras, M. Johnson, et al., A fast and accurate vietnamese word segmenter, in: Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018).

[39] D. Q. Nguyen, T. Vu, M. Dras, M. Johnson, et al., From word segmentation to pos tagging for vietnamese, in: Proceedings of the Australasian Language Technology Association Workshop 2017, pp. 108–113.