Building a Vietnamese Dataset for Natural Language Inference Models

Chinh Trong Nguyen[1] and Dang Tuan Nguyen[2](✉)

[1] University of Information Technology, VNU-HCM, Ho Chi Minh City, Vietnam
chinhnt@uit.edu.vn
[2] Saigon University, Ho Chi Minh City, Vietnam
dangnt@sgu.edu.vn

Abstract. Natural language inference models are important resources for many natural language understanding applications. These models are possibly built by training or fine-tuning using deep neural network architectures for state-of-the-art results. This means high-quality annotated datasets are important for building state-of-the-art models. Therefore, we propose a method of building Vietnamese dataset for training Vietnamese inference models which work on native Vietnamese texts. Our method aims at two issues: removing cue marks and ensuring the writing-style of Vietnamese texts. If a dataset contains cue marks, the trained models will identify the relation between a premise and a hypothesis without semantic computation. For evaluation, we fine-tuned a BERT model on our dataset and compared it to a BERT model which was fine-tuned on XNLI dataset. The model which was fine-tuned on our dataset has the accuracy of 86.05% while the other has the accuracy of 64.04% when testing on our Vietnamese test set. This means our method is possibly used for building a high-quality Vietnamese natural language inference dataset.

Keywords: Natural language inference · Textual entailment · NLI dataset · Transfer learning

1 Introduction

Natural language inference (NLI) research aims at identifying whether a text ***p***, called the premise, implies a text ***h***, called the hypothesis, in natural language. NLI is an important problem in natural language understanding (NLU). It is possibly applied in question answering [1–3] and summarization systems[4, 5]. NLI was early introduced as RTE [6] (Recognizing Textual Entailment). The early researches in RTE were divided in two different approaches [6] similarity-based and proof-based. In similarity-based approach, the premise and the hypothesis are parsed into representation structures, such as syntactic dependency parses, then a similarity is computed on these representations. In general, the high similarity of the premise-hypothesis pair means there is an entailment relation. However, there are many cases that the similarity of the premise-hypothesis pair is high but there is no entailment relation. The similarity is possibly defined as a

T. K. Dang et al. (Eds.): FDSE 2021, CCIS 1500, pp. 185–199, 2021.
https://doi.org/10.1007/978-981-16-8062-5_12

handcraft heuristic function, or an edit-distance based measure. In proof-based approach, the premise and the hypothesis are translated into formal logic then the entailment relation is identified by a proving process. This approach has an obstacle of translating a sentence into formal logic which is a complex problem.

Recently, NLI problem has been studied on classification-based approach thus deep neural networks are effective for solving this problem. The release of BERT architecture [7] showed many impressive results of improving benchmarks in many NLP tasks including NLI. When using BERT architecture, we will save many efforts in creating lexicon semantic resources, parsing sentences into appropriate representation, and defining similarity measures or proving schemes. The only one problem when using BERT architecture is the high-quality training dataset for NLI. Therefore, many RTE or NLI datasets have been released for years. In 2014, SICK [8] was released with 10k English sentence pairs for RTE evaluation. SNLI [9] has the similar format of SICK with 570k pairs of text span in English. In SNLI dataset, the premises and the hypotheses may be sentences or groups of sentences. The training and testing results of many models on SNLI dataset was higher than on SICK dataset. Similarly, MultiNLI [10] with 433k English sentence pairs was created by annotating on multi-genre documents for increasing the difficulty of the dataset. For cross-lingual NLI evaluation, XNLI [11] was created by annotating different English documents from SNLI and MultiNLI.

For building Vietnamese NLI dataset, we may use machine translator for translating the above datasets into Vietnamese. Some Vietnamese NLI (RTE) models was created by training or fine-tuning on Vietnamese translated versions of English NLI dataset for experiments. The Vietnamese translated version of RTE-3 was used for evaluation of similarity-based RTE in Vietnamese [12]. When evaluating PhoBERT in NLI task [13], the Vietnamese translated version of MultiNLI was used for fine-tuning. Although we can use machine translator for automatically building Vietnamese NLI dataset, we should build our Vietnamese NLI datasets for two reasons. The first reason is that some existing NLI datasets contain cue marks which was used for entailment relation identification without considering the premises [14]. The second reason is that the translated texts may not ensure the Vietnamese writing style or may return weird sentences.

In this paper, we would like to propose our method of building a Vietnamese NLI dataset which is annotated from Vietnamese news for ensuring writing style and contains more "*contradiction*" samples for removing cue marks. When proposing our method, we would like to reduce the annotation cost by using entailment sentence pairs existing in news webpages. We present this paper in five sections. Section 1 introduces the demand of building Vietnamese NLI dataset for building Vietnamese NLI models. Section 2 presents our proposed method of building Vietnamese NLI dataset. Section 3 presents the process of building Vietnamese NLI dataset and some experiments. Section 4 presents some experiments on our dataset in Vietnamese NLI. Then, some conclusions and our future works are presented in Sect. 5.

2 The Constructing Method

Our approach in building Vietnamese NLI dataset is generating samples from existing entailment pairs. These entailment pairs will be crawled from Vietnamese news websites for saving annotation cost, ensuring writing style and multi-genre.

2.1 NLI Sample Generation

The first requirement about our NLI dataset is that it does not contain cue marks. If a dataset contains these marks, the model trained on this dataset will identify "*contradiction*" and "*entailment*" relations without considering the premises or hypotheses [14]. Therefore, we will generate samples in which the premise and the hypothesis have many common words while their relation varies. We used some logic implication rules for this generation task. Given A and B are propositions, we will have the relations of eight premise-hypothesis types as shown in Table 1.

We used premise-hypothesis types 1 to 4 for removing the cues marks. When training a model, the model will learn from samples of types 1 to 4 the ability of recognizing the same sentences and contradiction sentences. We also used types 5 and 6 for training the ability of recognizing the summarization and paraphrase cases. Type 6 is added in the attempt of removing special marks which can occur when creating type 5 samples. We also added types 7 and 8 for recognizing the contradiction in paraphrase and summarization cases in which the proposition B is the paraphrase or the summary of the proposition A, respectively. Types 7 and 8 are valid only if B is the paraphrase or the summary of A.

Table 1. The relations of premise-hypothesis types used for building supplement dataset.

Type	Condition	P	H	Relation
1		A	A	entailment
2		¬A	¬A	entailment
3		A	¬A	contradiction
4		¬A	A	contradiction
5	A⇒B	A	B	entailment
6	A⇒B	¬B	¬A	entailment
7	A⇒B	A	¬B	contradiction*
8	A⇒B	¬A	B	contradiction*

In general, the types 7 and 8 cannot be applied in cases where the proposition A implies the proposition B by using presuppositions. For example, assuming A is the proposition "*we are hungry*", B is the proposition "*we will have lunch*" and A⇒B is the valid proposition "*if we are hungry then we will have lunch*" because we have two presuppositions that we should eat when we are hungry and we eat when we have lunch. We see that ¬B, which is the proposition "*we will not have lunch*", is not the contradiction of the proposition A.

2.2 Entailment Pair Collection

Entailment pairs exist in text documents, but it is difficult to extract them from the text documents. Therefore, after considering many news posts on many Vietnamese news

websites such as, VnExpress[1], we found that the title is usually the paraphrase or the summary of the introductory sentence in a news post. We can divide the news posts into four types. In type 1, the title is the paraphrase of the introductory sentence in the news post. In the example shown in Fig. 1, the title "*Nhiều tài xế dừng xe đậy nắp cống suốt 10 ngày*" (in English: "*many drivers was stopping to close the drain cover in 10 days*") is a paraphrase of the introductory sentence "*Nhiều tài xế dừng ôtô giữa ngã tư để đậy lại miệng cống hở do chiếc nắp cong vênh và câu chuyện diễn ra suốt 10 ngày ở Volgograd*" (in English: "*Many drivers was stopping the cars at the crossroad to close the slightly opened drain cover because the drain cover was bent*").

Fig. 1. An example of type-1 news post from vnexpress.net website

In type 2, the title is the summary of the introductory sentence in the news post. In the example shown in Fig. 2, the title "*Gạo chữa nhiều bệnh*" (in English: "*rice used for curing many diseases*") is the summary of the introductory sentence "Gạo nếp và gạo tẻ đều có vị thơm ngon, mềm dẻo, vừa cung cấp dinh dưỡng, vừa chữa nhiều bệnh như nôn mửa, rối loạn tiêu hóa, sốt cao" (in English: "*Glutinous rice and plain rice, which are delicious and soft when cooked, provide nutrition and are used for curing many diseases such as vomiting, digestive disorders, high fever*").

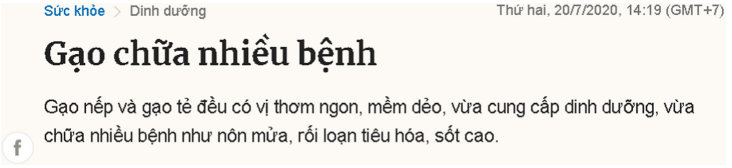

Fig. 2. An example of type-2 news post from vnexpress.net website

In type 3, the title is possibly inferred from the introductory sentence in the news post. Some pre-suppositions are possibly used in this inference. In the example shown in Fig. 3, the title "*Xuất khẩu rau quả tăng mạnh*" (in English: "*Vegetable export increases significantly*") can be inferred from the introductory sentence "*Bốn tháng đầu năm nay, giá trị xuất khẩu rau quả đạt 1,35 tỷ USD, tăng 9,5% so với cùng kỳ năm ngoái.*" (In English: "*in the first four months this year, vegetable export reaches 1.35 billion USD, increases 9.5% in comparison with the same period in last year*"). In this inference, we have used a pre-supposition which defines that increasing 9.5% means increasing significantly in export.

[1] https://vnexpress.net.

Fig. 3. An example of type-3 news post from vnexpress.net website

In type 4, the title is a question which cannot have an entailment relation to the introductory sentence in the news post. In the example shown in Fig. 4, the title, which is a question "*Vì sao giá dầu lao dốc chỉ trong 6 tuần?*" (In English: "*why does the oil price dramatically decreases in 6 weeks only*"), cannot have an entailment relation with the introductory sentence "*Chỉ mới cách đây hơn một tháng, giới buôn dầu còn lo ngại thiếu cung có thể đẩy dầu thô lên 100 USD một thùng.*" (In English: "*just more than one month ago, oil traders still worried that the insufficient supply could increase the oil price by 100 USD per barrel*").

Fig. 4. An example of type-4 news post from vnexpress.net website

We collected only title-introductory sentence pairs of type 1 and type 2 to make entailment pair collection because the pairs of type 3 and 4 cannot be applied 8 relation types when generating NLI samples. The type of a sentence pair is identified manually for high quality. In every pair in our collection, its title is the hypothesis, and its introductory sentence is the premise.

3 Building Vietnamese NLI Dataset

We built our NLI dataset with a three-step process. In the first step, we extracted title-introductory pairs from Vietnamese news websites. In the second step, we manually selected entailment pair and made the contradiction sentences from titles and introductory sentences for high quality. In the third step, we generate NLI samples from entailment pairs automatically and their contradiction sentences by applying 8 relation types shown in **Table 1.**

3.1 Contradiction Creation Guidelines

We made the contraction of a sentence manually for high-quality result. We proposed three types of making the contradiction. These are simple ways to make the contradiction of a sentence using syntactic transformation and lexicon semantic. In the type 1, a given

sentence will be transformed from affirmative to negative or vice versa by adding or removing the negative adverb. If the given sentence is an affirmative sentence, we will add a negative adverb to modifier the main verb of the sentence. If the given sentence is a negative sentence, we will remove the negative adverb which is modifying the main verb of the sentence. The negative adverbs used in our work are "*không"*, "*chưa"* and "*chẳng*" (in English: they mean "*not*" or "*not…yet"*). We used one of these adverbs according to the sentence for ensuring the Vietnamese writing-style. We have four cases of making contradiction with this type.

Case 1 of type 1, making contradiction from an affirmative sentence containing one verb. We will add one negative adverb to modify the verb. For example, making the contradiction of the sentence "*Đài Loan bầu lãnh đạo"* (in English: "*Taiwan voted for a Leader*"), we will add negative adverb "*không*" ("*not*") to modify the main verb "*bầu"* ("*voted*") for making the contradiction "*Đài Loan không bầu lãnh đạo"* (in English: "*Taiwan did not vote for a Leader"*).

Case 2 of type 1, making contradiction from an affirmative sentence containing a main verb and other verbs. We will add one negative adverb to modify the main verb only. For example, making the contradiction of the sentence "*Báo Mỹ đánh giá Việt Nam chống Covid-19 tốt nhất thế giới"* (in English: "*US news reported that Vietnam was the World's best nation in Covid-19 prevention"*), we will only add negative adverb "*không"* to modify the main verb "*đánh giá"* ("*reported"*) for making the contradiction "*Báo Mỹ không đánh giá Việt Nam chống Covid-19 tốt nhất thế giới*" (in English: "*US news did not report that Vietnam was the World's best nation in Covid-19 prevention"*).

Case 3 of type 1, making contradiction from an affirmative sentence containing two or more main verbs. We will add negative adverbs to modify all main verbs. For example, making the contradiction of the sentence "*Bão Irma mang theo mưa lớn và gió mạnh đổ bộ Cuba cuối tuần trước, biến thủ đô Havana như một 'bể bơi khổng lồ*"' (in English: "*Storm Irma brought heavy rain and winds to Cuba last week, making the Capital Havana a 'giant swimming pool"'*), we will add two negative adverbs "*không"* to modify two main verbs "*mang"* and "*biến"* for making the contradiction "*Bão Irma không mang theo mưa lớn và gió mạnh đổ bộ Cuba cuối tuần trước, không biến thủ đô Havana như một" "bể bơi khổng lồ"* (in English: "*Storm Irma did not bring heavy rain and winds to Cuba last week, not making the Capital Havana a 'giant swimming pool"'*).

Case 4 of type 1, making contradiction from a negative sentence containing negative adverbs. We will remove all negative adverbs in the sentence. In our data, we did not see any sentence of this case; however, we put this case in our guidelines for further use.

In the type 2, a given sentence or phrase will be transformed using the structure "*không có …*" (in English: "*there is/are no"*) or "*không … nào …*" (in English: "*no …*"). We have two cases of making contradiction with this type.

Case 1 of type 2, making contradiction from an affirmative sentence by using structure "*không có …*". We use this case when the given sentence has a quantity adjective or a cardinal number modifying the subject of the sentence and it is non-native if we add a negative adverb to modifying the main verb of the sentence. The quantity adjective or cardinal number will be replaced by the phrase "*không có"*. For example, making the contradiction of the sentence "*120 người Việt nhiễm nCoV ở châu Phi sắp về nước"* (in

English: "*120 Vietnamese nCoV-infested people in Africa are going to return home"*), we will replace "120" by "*không có*" because if we add negative adverb "*không*" to modify the main verb "về" ("*return*"), the sentence "*120 người Việt nhiễm nCoV ở châu Phi sắp không về nước*" (in English: "*120 Vietnamese nCoV-infested people in Africa are not going to return home*") sounds non-native. Therefore, the contradiction should be "*không có người Việt nhiễm nCoV ở châu Phi sắp về nước*" (in English: "*no Vietnamese nCoV-infested people in Africa is going to return home*"). Case 1 of type 2 will be used when we are given a phrase instead of a sentence. For example, making the contradiction of the phrase "*trường đào tạo quản gia cho giới siêu giàu Trung Quốc*" (in English: "*the butler training school for Chinese super-rich class*"), we will add the phrase "*không có*" at the beginning of the phrase to make the contradiction "*không có trường đào tạo quản gia cho giới siêu giàu Trung Quốc*" (in English: "*there is no butler training school for Chinese super-rich class*").

Case 2 of type 2, making contradiction from an affirmative sentence by using structure "*không …nào …*". We will use this structure when we have case 1 of type 2 but the generated result of that case is not native. For example, making the contradiction of the sentence "*gần ba triệu ngôi nhà tại Mỹ mất điện vì bão Irma*" (in English: "*nearly three million houses in U.S. were without power because of Irma storm*"), if we replace "*gần ba triệu*" (in English: "*nearly three million*") by "*không có*", we will have a non-native sentence "*không có ngôi nhà tại Mỹ mất điện vì bão Irma*" therefore we should use the structure "*không … nào …*" to make the contradiction "*không ngôi nhà nào tại Mỹ mất điện vì bão Irma*" (in English: "*There are no houses in U.S. were without power because of Irma storm*").

In type 3, a contradiction sentence is generated using lexicon semantic. A word of the given sentence will be replaced by its antonym. This way will make the contradiction of the given sentence. Although we can use all cases of type 1 and type 2 for making the contradiction, we still recommend this type because the samples generated with this type may help the fine-tuned models to learn more about antonymy. We have two cases of making contradiction with this type.

Case 1 of type 3, making contradiction from a sentence by replacing the main verb of the sentence with its antonym. For example, making the contradiction of the sentence "*Mỹ thêm gần 18.000 ca nCoV một ngày*" (in English: "*the number of nCoV cases in U.S. increases about 18.000 in one day*"), we can replace the main verb "*thêm*" ("*increase*") by its antonym "*giảm*" ("*decrease*") to make the contradiction "*Mỹ giảm gần 18.000 ca nCoV một ngày*" (in English: "*the number of nCoV cases in U.S. decreases about 18.000 in one day*").

Case 2 of type 3, making contradiction from a given sentence by replacing an adverb or a phrase modifying the main verb by the antonym or the contradiction of that adverb or that phrase, respectively. We use this case when we need to make the samples containing antonymy, but the main verb does not have any antonyms because there are many verbs which do not have their antonym. For example, making the contradiction of the sentence "*Mỹ viện trợ nhỏ giọt chống Covid-19*" (in English: "*the U.S. aided a little in Covid-19 prevention*"), we cannot replace the main verb "*viện trợ*" ("*aid*") with its antonym because it does not have an antonym. Therefore, we will replace "*nhỏ giọt*" ("*a little*") by "*ào ạt*" ("*a lot*") to make the contradiction "*Mỹ viện trợ ào ạt chống Covid-19*" (in

English: "*the U.S. aided a lot in Covid-19 prevention*"). In this example, "*nhỏ giọt*" and "*ào ạt*" have the opposite meanings; and the phrases "*nhỏ giọt*" and "*ào ạt*" have the adverb role in the sentence when modifying the main verb "*viện trợ*".

3.2 Building Steps

We built our Vietnamese NLI dataset follow the three-step process which is a semi-automatic process shown in **Fig. 5**.

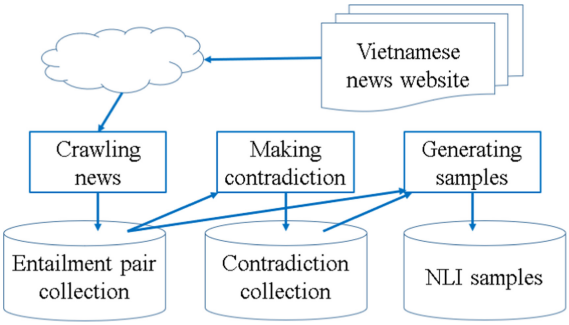

Fig. 5. Our three-step process of building Vietnamese NLI dataset

In the first step – crawling news, we used a crawler to fetch unique webpages from sections of international news, business, life, science, and education in website *vnexpress.net*. Then we extracted their titles and introductory sentences by a website-specific pattern defined with regular expression. The results are sentence pairs stored in an entailment pair collection with unique numbers. These pairs are not always type 1 or 2 therefore the entailment pairs will be manually selected right before making contradiction sentences.

In the second step – making contradiction, we firstly manually identified if each pair of the collection was type 1 or 2 for entailment pair selection. When an entailment pair was selected, we made the contradiction sentences for the title and the introductory sentence using the contradiction creation guidelines. In the entailment pairs, the introductory sentences are the premises, and the titles are the hypotheses. As the results, we have a collection of pairs of sentences ¬A and ¬B stored in contradiction collection in which each sentence pair ¬A and ¬B has a condition A⇒B. In this step, we have two people making contradiction sentences. These people are society science bachelors. Because the guidelines of making contradiction sentence are simple, there are no disagreements in the annotation results.

In the third step – generating samples, we used a computer program implemented from Algorithm 1 for combining the premises, hypotheses stored in entailment pair collection and their contradiction sentences stored in contradiction collection by their unique numbers. The combination rules follow Table 1 in generating NLI samples. For generating "*neutral*" samples, the computer program combined sentences from different premise-hypothesis pairs. In Algorithm 1, the function *getContradict()* return the contradiction sentence stored in contradiction collection. The three functions *ent()*, *neu()*, and

con() are used for creating entailment, neutral and contradiction sample from a premise and a hypothesis, respectively.

Algorithm 1 Generating NLI samples.

```
Input: E, a list of premise-hypothesis pairs.
Output: SD, the NLI sample data with SNLI format.

SD←∅
PL←∅  //premise list
HL←∅  //hypothesis list
cPL←∅ //premise contradiction list
nHL←∅ //hypothesis contradiction list
for i←1 to |E|
   prem ← E[i].premise
   hyp ← E[i].hypothesis
   nprem ← genContradict(prem)
   nhyp ← genContradict(hyp)
   if nprem = NULL and nhyp = NULL then
      continue
   end if
   PL←PL + {prem}
   HL←HL+{hyp}
   cPL←nPL+{nprem}
   cHL←nHL+{nhyp}
end for
PL ← PL+{PL[1]}, HL ← HL+{HL[1]}
cPL ← nPL+{nPL[1]}, cHL ← nHL+{nHL[1]}
for i←1 to |PL|-1
   SD ← SD+ent(PL[i],PL[i])+ent(HL[i],HL[i])
          +ent(PL[i],HL[i])+neu(PL[i],PL[i+1])
          +neu(PL[i+1],PL[i])+neu(HL[i],HL[i+1])
          +neu(HL[i+1],HL[i])+neu(PL[i],HL[i+1])
          +neu(PL[i+1],HL[i])+neu(HL[i],PL[i+1])
          +neu(HL[i+1],PL[i])
   if cHL[i] != NULL then
      SD ← SD+con(HL[i],cHL[i])+con(cHL[i],HL[i])
             +ent(cHL[i],cHL[i])
      if cHL[i+1] != NULL then
```

```
26            SD ← SD+neu(PL[i],cHL[i])+neu(cHL[i],cHL[i+1])
                  +neu(cHL[i+1],PL[i])+neu(cHL[i+1],cHL[i])
27        end if
28        SD ← SD+neu(PL[i+1],cHL[i])+neu(cHL[i],PL[i+1])
29    end if
30    if cPL[i] != NULL then
31        SD ← SD+con(PL[i],cPL[i])+con(cPL[i],PL[i])
32        SD ← SD+ent(cPL[i],cPL[i])
33        if cPL[i+1] != NULL then
34            SD ← SD+neu(HL[i],cPL[i+1])+neu(cPL[i],cPL[i+1])
                  +neu(cPL[i+1],PL[i])+neu(cPL[i+1],HL[i])
35        end if
36            SD ← SD +neu(HL[i+1],cPL[i])+neu(cPL[i],HL[i+1])
37    end if
38    if cPL[i]!=NULL && cHL[i]!=NULL then
39        SD ← SD+ent(cHL[i+1],cPL[i])
40        if cHL[i+1] != NULL then
41            SD ← SD+neu(cPL[i],cHL[i+1])
                  +neu(cHL[i+1],cPL[i])
42        end if
43        if cPL[i+1] != NULL then
44            SD ← SD+neu(cHL[i],cPL[i+1])+neu(cPL[i+1],cHL[i])
45        end if
46    end if
47 end for
48 return SD
```

3.3 Building Results

In our present NLI dataset, called VnNewsNLI, the rates of making contradiction sentences by applying type 1, type 2 and type 3 are 61.74%, 17.67% and 20.58%, respectively. The rates of entailment, neutral and contradiction samples in our VnNewsNLI dataset are shown in Table 2. In Table 2, the rates of sample types are approximate. Although the rate of neutral samples (30.70%) is lower than of others in development set, the differences in number between these samples are not much therefore the development set is still balanced.

The statistics of the VnNewsNLI dataset by syllable are shown in Table 3. We used syllable as text length unit in Table 3 because there are many multi-lingual pretrained model which were trained on unsegmented Vietnamese text datasets. According to Table 3, the premises and hypotheses are often short (9–14 syllables) and quite long (> 26

syllables) sentences therefore this dataset may provide the characteristic of short and long sentences. There is a difference between the VnNewsNLI dataset and the SNLI dataset that the premises and hypotheses are almost sentences in the VnNewsNLI dataset while they are almost groups of sentences in the SNLI dataset.

Table 2. The statistics of NLI samples in VnNewsNLI dataset

Criterion	Development set		Test set	
	n	%	n	%
Entailment	947	34.74%	4,140	33.42%
Contradiction	942	34.56%	4,128	33.33%
Neutral	837	30.70%	4,118	33.25%
Total	2,726	100.00%	12,386	100.00%

Table 3. The statistics of NLI samples by syllable in VnNewsNLI dataset. (ent. – entailment, neu. – neutral, con. – contradiction).

Length in syllable	Development set			Test set		
	ent	neu	con	ent	neu	con
Premises, ≤ 8	55	54	37	267	266	188
Premises, 9–14	334	332	227	1589	1575	1060
Premises, 15–20	86	85	54	217	214	134
Premises, 20–26	48	35	60	163	155	212
Premises, > 26	424	331	564	1904	1908	2534
All premises	**947**	**837**	**942**	**4140**	**4118**	**4128**
Hypotheses, ≤ 8	62	54	75	297	266	376
Hypotheses, 9–14	346	332	453	1615	1575	2126
Hypotheses, 15–20	70	85	102	167	214	250
Hypotheses, 20–26	45	36	30	155	155	106
Hypotheses, > 26	424	330	282	1906	1908	1270
All hypotheses	**947**	**837**	**942**	**4140**	**4118**	**4128**

4 Experiments

We did some experiments on our VnNewsNLI dataset and on Vietnamese XNLI dataset [11] then compared their results to find if our dataset is useful when building a Vietnamese NLI model. XNLI dataset was manually annotated from English texts then the annotated

results were translated into different languages using machine translators. Therefore, Vietnamese XNLI dataset is a Vietnamese translated NLI dataset. For experiments, we used BERT architecture for training Vietnamese NLI models as shown in Fig. 6.

According to the BERT architecture in Fig. 6, a premise and a hypothesis of a sample will be concatenated into an input. This input has the following order: the "*[CLS]*" token, then all premise's tokens, then the "*[SEP]*" token, then all hypothesis' tokens, and the "*[SEP]*" token at the end. Each input token will be converted to a tuple of word embedding, segment embedding and position embedding. These embeddings will go through BERT architecture to generate a context vector for each input token and a context vector for the whole input. The context vector of the whole input is returned at the "*[CLS]*" position. This vector will be used for identifying the relation between the premise and the hypothesis by a classifier. This classifier is a feed forward neural network fully connected to the context vector of the input. It will be trained in fine-tuning steps. We chose BERT architecture for experiment because it can compute the context vector with syntactic and semantic features of the input [15–17].

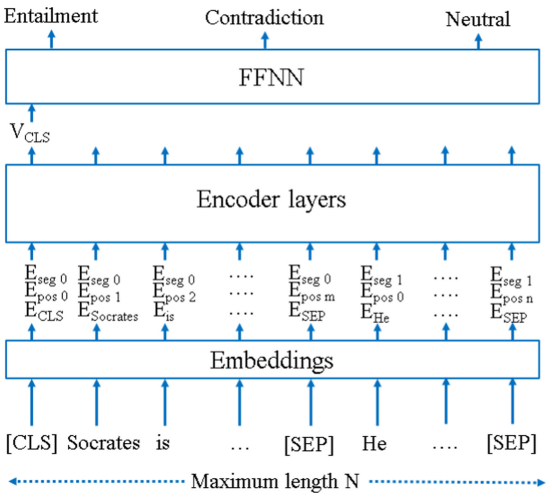

Fig. 6. The illustration of NLI BERT architecture[7]

4.1 Experiment Settings

We built two Vietnamese NLI models using BERT architecture as shown in **Fig. 6**. The first model, viXNLI, was fine-tuned from PhoBERT pretrained-model [13] on Vietnamese version of XNLI development set with word segmentation. The second model, viNLI, was fine-tuned from PhoBERT pretrain-model on our VnNewsNLI development set with word segmentation. We used a small Vietnamese development set of XNLI and an equally small development set of VnNewsNLI for showing the efficiency when using PhoBERT pre-trained model. We used Huggingface python library[18] for implementing

the BERT architecture and fairseq python library[19] for tokenizing Vietnamese words into sub-words. We also used VnCoreNLP [20] for word segmentation.

We fine-tuned these models in 2 to 8 epochs with learning rate of 3.10^{-5}, batch size of 16 and input maximum length of 200 because the $\text{PhoBERT}_{\text{base}}$ pretrained model has the limit input length of 258 and the lengths of the premises and hypotheses are rarely greater than 100 syllables. Other parameters were left with default settings. We chose the best models from checkpoints for testing.

4.2 Experiment Results

The experiment results are shown in Table 4. In Table 4, the accuracy of viNLI model (40.30%) is lower than of viXNLI model (68.64%). In our VnNewsNLI dataset, each premise or hypothesis is a sentence. In XNLI dataset, each premise or hypothesis is translated from English and is a group of sentences. Our viNLI model was fine-tuned on our VnNewsNLI dataset therefore it may not capture the semantic of multi-sentential premise-hypothesis pairs in XNLI test set effectively. In contrast, viXNLI was fine-tuned on XNLI dataset therefore it may capture the semantic of premise-hypothesis pairs effectively in both XNLI's samples and VnNewsNLI's samples. This is the reason why viXNLI's accuracy on XNLI (68.64%) approximates to viXNLI's accuracy on VnNewsNLI (64.04%) while there are big gaps between the viNLI's accuracies on XNLI (40.30%) and on VnNewsNLI (86.05%) and between the viXNLI's accuracy (64.04%) and viNLI's accuracy (86.05%) on the same VnNewsNLI test set.

Table 4. The accuracy of viXNLI and viNLI models on test datasets

Dataset	viXNLI (%)	viNLI (%)
XNLI test set	68.64	40.30
VnNewsNLI test set	64.04	86.05

The accuracy of viNLI model (86.05%) is higher than the accuracy of viXNLI model (64.04%) on VnNewNLI test set. This means our development set is more appropriate for fine-tuning a Vietnamese NLI model than the Vietnamese XNLI's development set. It also means our proposed method is possibly used for building Vietnamese NLI dataset with an attention in adding many multi-sentential.

In our experiment, we fine-tuned viXNLI and viNLI models on two small development sets with about 2,500 samples and test them on two larger test sets with about 5,000 samples and 12,000 samples. The results shows that BERT pre-train models are possibly fine-tuned on small datasets to build effective models as described in [7].

5 Conclusion and Future Works

In this paper, we proposed a method of building a Vietnamese NLI dataset for fine-tuning and testing Vietnamese NLI models. This method is aimed at two issues. The first issue

is the cue marks which are used by the trained model for identifying the relation between a premise and a hypothesis without considering the premise. We addressed this issue by generating samples using eight types of premise-hypothesis pair. The second issue is the Vietnamese writing style of samples. We addressed this issue by generating samples from titles and introductory sentences of Vietnamese news webpages. We used title-introductory pairs of appropriate webpages for reducing annotation cost. These samples were generated by applying a semi-automatic process. For evaluating our method, we built our VnNewsNLI dataset by extracting the title and the introductory sentence of many webpages in a Vietnamese news website VnExpress and applied our building process. When building our VnNewsNLI, we had two people manually annotated each sentence for generating contraction sentences.

We evaluated our proposed method by comparing the results of a NLI model, viXNLI, fine-tuned on Vietnamese XNLI dataset and of a NLI model, viNLI, fine-tuned on our VnNewsNLI dataset. We used the same deep neural network architecture BERT for building these NLI models. The results showed that viNLI model had a higher accuracy (86.05% vs. 64.04%) on our VnNewsNLI test set while it had a lower accuracy (40.30% vs. 68.64%) on Vietnamese XNLI test set when comparing to viXNLI. The VnNewsNLI's accuracy of 86.05% showed a promise of building high-quality Vietnamese NLI dataset from Vietnamese documents for ensuring writing-style.

Currently, our VnNewsNLI dataset contains a quite small number of samples with about 15,000 samples. In future, we will apply our proposed process for building a large and high-quality multi-genre Vietnamese NLI dataset.

References

1. Punyakanok, V., Roth, D., Yih, W.-T.: Natural language inference via dependency tree mapping: an application to question answering. Comput. Linguist. **6**, 10 (2004)
2. Lan, W., Xu, W.: Neural network models for paraphrase identification, semantic textual similarity, natural language inference, and question answering. In: International Conference on Computational Linguistics, pp. 3890–3902. Association for Computational Linguistics, Santa Fe, New Mexico, USA (2018)
3. Minbyul, J., et al.: Transferability of natural language inference to biomedical question answering. In: Conference and Labs of the Evaluation Forum, Thessaloniki, Greece (2020)
4. Falke, T., Ribeiro, L.F.R., Utama, P.A., Dagan, I., Gurevych, I.: Ranking generated summaries by correctness: an interesting but challenging application for natural language inference. In: Annual Meeting of the Association for Computational Linguistics, pp. 2214–2220. Association for Computational Linguistics, Florence, Italy (2019)
5. Pasunuru, R., Guo, H., Bansal, M.: Towards improving abstractive summarization via entailment generation. In: Workshop on New Frontiers in Summarization, pp. 27–32. Association for Computational Linguistics, Copenhagen, Denmark (2017)
6. Dagan, I., Roth, D., Sammons, M., Zanzotto, F.M.: Recognizing Textual Entailment: Models and Applications. Morgan & Claypool Publishers, San Rafael (2013)
7. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: BERT: pre-training of deep bidirectional transformers for language understanding. In: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4171–4186. Association for Computational Linguistics (2019)

8. Marelli, M., Menini, S., Baroni, M., Bentivogli, L., Bernardi, R., Zamparelli, R.: A SICK cure for the evaluation of compositional distributional semantic models. In: International Conference on Language Resources and Evaluation, pp. 216–223. European Language Resources Association, Reykjavik, Iceland (2014)
9. Bowman, S.R., Angeli, G., Potts, C., Manning, C.D.: A large annotated corpus for learning natural language inference. In: Conference on Empirical Methods in Natural Language Processing, pp. 632–642. Association for Computational Linguistics, Lisbon (2015)
10. Williams, A., Nangia, N., Bowman, S.R.: A broad-coverage challenge corpus for sentence understanding through inference. In: Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, vol. 1, pp. 1112–1122. Association for Computational Linguistics, New Orleans (2017)
11. Conneau, A., et al.: XNLI: evaluating cross-lingual sentence representations. In: Conference on Empirical Methods in Natural Language Processing, pp. 2475–2485. Association for Computational Linguistics, Brussels (2018)
12. Nguyen, M.-T., Ha, Q.-T., Nguyen, T.-D., Nguyen, T.-T., Nguyen, L.-M.: Recognizing textual entailment in vietnamese text: an experimental study. In: International Conference on Knowledge and Systems Engineering, pp. 108–113. IEEE, Ho Chi Minh City (2015)
13. Nguyen, D.Q., Nguyen, A.T.: PhoBERT: pre-trained language models for Vietnamese. In: Conference on Empirical Methods in Natural Language, pp. 1037–1042 (2020)
14. Jiang, N., de Marneffe, M.-C.: Evaluating BERT for natural language inference: a case study on the CommitmentBank. In: Conference on Empirical Methods in Natural Language Processing, pp. 6086–6091. Association for Computational Linguistics, Hong Kong (2019)
15. Tenney, I., Das, D., Pavlick, E.: BERT rediscovers the classical NLP pipeline. Annual Meeting of the Association for Computational Linguistics, pp. 4593–4601. Association for Computational Linguistics, Florence (2019)
16. Rogers, A., Kovaleva, O., Rumshisky, A.: A primer in bertology: what we know about how bert works. Trans. Assoc. Comput. Linguist. **8**, 842–866 (2020)
17. Peters, M.E., Neumann, M., Zettlemoyer, L., Yih, W.-T.: Dissecting contextual word embeddings: architecture and representation. In: Conference on Empirical Methods in Natural Language Processing, pp. 1499–1509. Association for Computational Linguistics, Brussels (2018)
18. Wolf, T., et al.: Transformers: state-of-the-art natural language processing. In: Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp. 38–45. Association for Computational Linguistics (2020)
19. Ott, M., et al.: fairseq: a fast, extensible toolkit for sequence modeling. In: Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations), pp. 48–53. Association for Computational Linguistics, Minneapolis (2019)
20. Vu, T., Nguyen, D.Q., Nguyen, D.Q., Dras, M., Johnson, M.: VnCoreNLP: a Vietnamese natural language processing toolkit. In: Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations, pp. 56–60. Association for Computational Linguistics, New Orleans (2018)