SN Computer Science (2022) 3: 395
https://doi.org/10.1007/s42979-022-01267-x

ORIGINAL RESEARCH

Building a Vietnamese Dataset for Natural Language Inference Models

Chinh Trong Nguyen[1] · **Dang Tuan Nguyen**[2]

Received: 20 April 2022 / Accepted: 22 June 2022 / Published online: 25 July 2022

Abstract

Natural language inference models are essential 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. That means high-quality annotated datasets are essential for building state-of-the-art models. Therefore, we propose a method to build a Vietnamese dataset for training Vietnamese inference models which work on native Vietnamese texts. Our approach 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 relationship between a premise and a hypothesis without semantic computation. For evaluation, we fine-tuned a BERT model, viNLI, on our dataset and compared it to a BERT model, viXNLI, which was fine-tuned on XNLI dataset. The viNLI model has an accuracy of 94.79%, while the viXNLI model has an accuracy of 64.04% when testing on our Vietnamese test set. In addition, we also conducted an answer selection experiment with these two models in which the P@1 of viNLI and of viXNLI are 0.4949 and 0.4044, respectively. That means our method can be used to build a high-quality Vietnamese natural language inference dataset.

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

Introduction

Natural language inference (NLI) research aims at identifying whether a text $\boldsymbol{p}$, called the premise, implies a text $\boldsymbol{h}$, 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 RTE researches were divided into two approaches [6], similarity-based and proof-based. In a similarity-based approach, the premise and the hypothesis are parsed into representation structures, such as syntactic dependency parses, and then the 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 where the similarity of the premise-hypothesis pair is high, but there is no entailment relation. The similarity is possibly defined as a handcraft heuristic function or an edit-distance based measure. In a 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, the NLI problem has been studied on a classification-based approach; thus, deep neural networks effectively solve this problem. The release of BERT architecture [7] showed many impressive results in improving NLP tasks' benchmarks, including NLI. Using BERT architecture will save many efforts in creating lexicon semantic resources,

This article is part of the topical collection "Future Data and Security Engineering 2021" guest edited by Tran Khanh Dang.

Biographical Notes: This paper is a revised and expanded version of our paper entitled "Building a Vietnamese Dataset for Natural Language Inference Models" presented at The 8th International Conference on Future Data and Security Engineering: Big Data, Security and Privacy, Smart City and Industry 4.0 Applications, FDSE 2021, Virtual Event, November 24–26, 2021. Communications in Computer and Information Science 1500, Springer 2021.

✉ Dang Tuan Nguyen
dangnt@sgu.edu.vn

Chinh Trong Nguyen
chinhnt@uit.edu.vn

[1] University of Information Technology, VNU-HCM, Ho Chi Minh City, Vietnam

[2] Saigon University, Ho Chi Minh City, Vietnam

parsing sentences into appropriate representation, and defining similarity measures or proving schemes. The only 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 10 k English sentence pairs for RTE evaluation. SNLI [9] has a similar SICK format with 570 k 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 433 k English sentence pairs was created by annotating on multi-genre documents to increase the dataset's difficulty. For cross-lingual NLI evaluation, XNLI [11] was created by annotating different English documents from SNLI and MultiNLI.

For building the Vietnamese NLI dataset, we may use a machine translator to translate 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 to evaluate 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 a machine translator to automatically build 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, which is the extended version of our paper [15], we propose our method of building a Vietnamese NLI dataset that is annotated from Vietnamese news to ensure 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 on news webpages. Our contributions are:

(1) To propose Vietnamese NLI dataset creation guidelines based on simple logic rules to ensure that there are no cue marks to determine the relation of a premise-hypothesis pair without semantic computation.
(2) To propose a method to create Vietnamese NLI samples with lower annotation cost by utilizing the title and the introductory sentence of every news from many news websites. In this method, the introductory sentence and the news title are the premise and the hypothesis of a sample, respectively. An annotator is required to check if a premise-hypothesis pair is an entailment sample and provide the contrary sentences from given sentences using our simple guidelines.

Our paper has six sections. The previous section introduces the demand for building the Vietnamese NLI dataset for building Vietnamese NLI models. The following section reviews related works on creating NLI datasets. "The Constructing Method" presents our proposed method of building the Vietnamese NLI dataset. In "Building Vietnamese NLI Dataset", we present the process of building the Vietnamese NLI dataset and some experiments and the subsequent section presents some experiments on our dataset in Vietnamese NLI. Then, some conclusions and our future works are presented in the next section.

Related Works

The early NLI datasets were created for RTE shared tasks. These datasets was manually annotated thus they are good but not large datasets. In 2014, the SICK dataset [8] was released in SemEval 2014. This dataset was created with a three-step process, including sentence normalization, sentence expansion and sentence pair generation. In this process, the sentence expansion step was to automatically create entailment and contradiction sentences by applying syntactic and lexical transformations. In 2015, The SNLI dataset [9] was released to address small datasets' problems and ungrammatical generated sentences. The SNLI dataset was totally annotated by about 2.500 workers [9]. In SNLI creating process, a group of workers had to provide the entailment, contradiction and neutral sentences for every given sentence to ensure the quality of the samples. After that, every five workers had to specify if the relation of a premise-hypothesis pair is entailment, contradiction or neutral. Finally, the relation of each sample was identified as the highest voted relation of the sample. In 2017, MultiNLI dataset was released [10] to provide multi-genre NLI dataset. The MultiNLI dataset was created using the same process of SNLI; however, its data were collected from both written and spoken speech in ten genres.

The Constructing Method

According to the information about SICK, SNLI and MultiNLI datasets, the processes of creation of those datasets required these three steps:

(1) The first step was sentence selection. The conformed sentences are selected as the premises in NLI examples.
(2) The second step was sentence generation. In this step, the contradiction, entailment and neutral sentences of

a given sentence were generated manually or automatically. This step affected the quality of the dataset.

(3) The third step was sample generation. This step had two options to generate samples. In the first option, the workers provided their judgement about given premise-hypothesis pairs for voting the final relations of those pairs. The premise-hypothesis pairs were generated from selected sentences and their entailment, contradiction sentences in the second option.

Our approach to building the Vietnamese NLI dataset is generating samples from existing entailment pairs. These entailment pairs will be crawled from Vietnamese news websites to reduce entailment annotation costs and ensure writing style and multi-genre. We have to annotate contradiction sentences to create our dataset only manually.

NLI Sample Generation

The first requirement of 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 logical implication rules for this generation task. For example, 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 to recognize the same sentences and contradiction sentences. We also used types 5 and 6 for training the ability to identify the summarization and paraphrase cases. Type 6 is added in the attempt to remove 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 proposition B is the paraphrase or the summary of proposition A, respectively. Types 7 and 8 are valid only if B is the paraphrase or A's summary.

In general, the types 7 and 8 cannot be applied in cases where proposition A implies proposition B by using pre-suppositions. 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 pre-suppositions 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 a contradiction of proposition 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*

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 Vietnamese news websites such as VnExpress, we found that the title usually paraphrases or summarizes the introductory sentence in a news post. Therefore, we can divide these 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*").

In type 2, the title summarizes 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

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

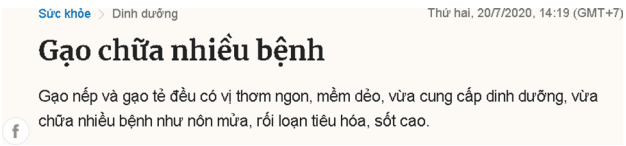

Fig. 2 An example of type-2 news post from vnexpress.net website

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

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

"*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*").

In type 3, the title is possibly inferred from the introductory sentence in the news post. Some pre-suppositions are perhaps 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 significantly growing exports.

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*").

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.

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 the entailment pair and made the contradiction sentences from titles and introductory sentences. Finally, in the third step, we automatically generate NLI samples from entailment pairs and their contradiction sentences by applying eight relation types shown in Table 1. In Table 1, the relations of type 1 and type 2 are apparent thus, we created a different version of our dataset in which there have no samples of type 1 and type 2 to show if these samples are meaningful.

Contradiction Creation Guidelines

We made the contraction of a sentence manually for a high-quality result. In our approach, the contradiction sentences are generated in two ways. The first way is to transform them from affirmative structure to negative structure and vice versa. The second way is to use antonyms. We proposed three types of making the contradiction in which type 1 and type 2 are to use structure transformations, and type 3 is to use antonyms. These are simple ways to make the contradiction of a sentence using syntactic transformation and lexicon semantic.

In 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 affirmative, we will add a negative adverb to modifier the sentence's main verb. If the given sentence is negative, we will remove the negative adverb, which is modifying the sentence's main verb. 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 to ensure the Vietnamese writing style. We have four cases of making contradictions 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 the 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 the 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 the 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 semantics. 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 to make the contradiction, we still recommend this type because the samples generated with this type may help the fine-tuned models 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 antonyms, but the main verb does not have any antonyms because many verbs 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*").

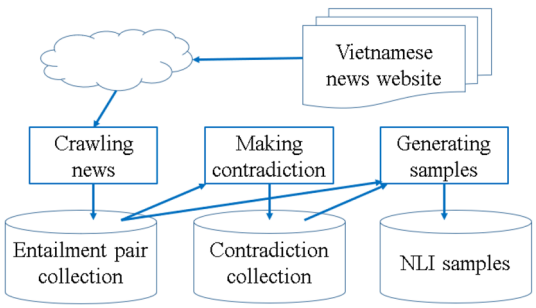

Fig. 5 Our three-step process of building Vietnamese NLI dataset

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*".

Building Steps

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

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 the 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 the types 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. The introductory sentences are the premises in the entailment pairs, and the titles are the hypotheses. As a result, we have a collection of pairs of sentences ¬A and ¬B stored in a 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 for 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 our 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. The computer program generates "neutral" samples to combine 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()* is used for creating entailment, neutral and contradiction samples from a premise and a hypothesis, respectively. For data balancing, we added some duplicated entailment samples in Algorithm 1.

Algorithm 1. Generating NLI samples.

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

SD←[]
PL←[]  //premise list
HL←[]  //hypothesis list
cPL←[] //premise contradiction list
cHL←[] //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←cPL+[nprem]
   cHL←cHL+[nhyp]
end for
PL←PL+[PL[1]], HL←HL+[HL[1]]
cPL←cPL+[cPL[1]], cHL←cHL∪[cHL[1]]
for i←2 to len(PL)
   SD←SD+[ent(PL[i],HL[i]), neu(PL[i],PL[i-1])]
   SD←SD+[ent(PL[i],HL[i]), neu(HL[i],HL[i-1])]
   SD←SD+[ent(PL[i],PL[i]), ent(HL[i],HL[i])]
   SD←SD+[neu(HL[i],PL[i-1]), neu(PL[i],HL[i-1])]
   if cPL[i]!=NULL and cHL[i]!=NULL then
      SD←SD+[ent(cHL[i],cPL[i]), neu(cHL[i],HL[i-1])]
      SD←SD+[ent(cHL[i],cPL[i]), neu(cPL[i],PL[i-1])]
   end if
   if cPL[i]!=NULL then
      SD←SD+[con(PL[i],cPL[i]), con(cPL[i],PL[i])]
      SD←SD+[con(PL[i],cPL[i]), con(cPL[i],PL[i])]
      SD←SD+[ent(cPL[i],cPL[i]), neu(PL[i-1],cPL[i])]
   end if
   if cHL[i]!=NULL then
      SD←SD+[con(HL[i],cHL[i]), con(cHL[i],HL[i])]
      SD←SD+[con(HL[i],cHL[i]), con(cHL[i],HL[i])]
      SD←SD+[ent(cHL[i],cHL[i]), neu(HL[i-1],cHL[i])]
   end if
return SD
```

Given a list of entailment samples E, Algorithm 1 firstly select from E a list of entailment samples in which the premise and the hypothesis of the ith sample are *PL[i]* and *HL[i]*. The ith sample is only selected if its premise *PL[i]* or hypothesis *HL[i]* has the contradiction premise *cPL[i]* or *cHL[i]*, respectively. Then, entailment and contradiction pairs are generated using the rules in Table 1. For example, a type 1 sample is generated as *ent(PL[i],PL[i])*, a type 3 sample is generated as *con(PL[i], cPL[i])* if the premise *PL[i]* has its contradiction *cPL[i]*, a type 5 sample is generated as *ent(PL[i], HL[i])*. The neutral samples are generated by pairing the premise, hypothesis, premise contradiction or hypothesis contradiction of the ith sample and the premise, hypothesis, premise contradiction or hypothesis contradiction of the i–1th sample as in building SICK dataset [8].

To show the necessity of the type 1 and type 2 relation in Table 1, we also used a different version of our Algorithm 1 to generate samples. In this version, which is presented in Algorithm 2, the samples of type 1 and type 2 are not generated when creating the dataset.

Algorithm 2. Generating NLI samples without type 1 and type 2.

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

SD←[]
PL←[]  //premise list
HL←[]  //hypothesis list
cPL←[] //premise contradiction list
cHL←[] //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←cPL+[nprem]
   cHL←cHL+[nhyp]
end for
PL←PL+[PL[1]], HL←HL+[HL[1]]
cPL←cPL+[cPL[1]], cHL←cHL∪[cHL[1]]
for i←2 to len(PL)
   SD←SD+[ent(PL[i],HL[i]), neu(PL[i],PL[i-1])]
   SD←SD+[ent(PL[i],HL[i]), neu(HL[i],HL[i-1])]
   if cPL[i]!=NULL and cHL[i]!=NULL then
      SD←SD+[ent(cHL[i],cPL[i]), neu(cHL[i],HL[i-1])]
      SD←SD+[ent(cHL[i],cPL[i]), neu(cPL[i],PL[i-1])]
   end if
   if cPL[i]!=NULL then
      SD←SD+[con(PL[i],cPL[i]), con(cPL[i],PL[i])]
   if cHL[i]!=NULL then
      SD←SD+[con(HL[i],cHL[i]), con(cHL[i],HL[i])]
return SD
```

Building Results

In our updated NLI dataset, VnNewsNLI, the rates of making contradiction sentences by applying type 1, type 2 and type 3 are 60.16%, 19.01% and 20.83%, respectively. We also created the $\text{VnNewsNLI}_\text{R}$, the types 1 and 2 sample removal version of VnNewsNLI using Algorithm 2. The rates of entailment, neutral and contradiction samples in our VnNewsNLI dataset are shown in Table 2. In Table 2, the rates of NLI relation categories are approximately 33.3%.

The statistics of the VnNewsNLI dataset by syllable are shown in Table 3. Table 3 and the distribution of the sentence length (in a syllable) on entailment, neutral and contradiction are shown in Table 4. We used syllables as text length units in Tables 3 and 4 because many multi-lingual pretrained models were trained on unsegmented Vietnamese text datasets. According to Tables 3 and 4, the premises and hypotheses are often short (≤ 14 syllables) and quite long (≥ 20 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 in that the premises and hypotheses are almost sentences in the VnNewsNLI dataset. At the same time, they are groups of sentences in many cases in the SNLI dataset.

We also calculated the frequency distribution of words in our both development set and test set to view the most discussing topics of the samples briefly. The 40 highest frequency words, common nouns and verbs, are presented in Table 5. The frequency distribution of words shows that the politics, military and life topics are most discussed in VnNewsNLI samples.

Experiments

We did some experiments on our VnNewsNLI dataset and on the Vietnamese XNLI dataset [11] and then compared their results to determine 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 translation of XNLI dataset. We also conducted an experiment to show the application of our dataset in answer selection. In this experiment, we used the Vietnamese NLI model for selecting the sentence containing the answer in machine reading comprehension tests. We selected the sentence with highest entailment score as the retrieval result and evaluating with the precision at top 1 (P@1) score. We used UIT-ViQuAD 2.0 dataset [16], which was the expansion of UIT-ViQuAD 1.0 [17], after removing no-answer samples for our evaluation. In our 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 [18–20].

Experiment Settings

We built three 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 Vietnamese word segmentation. The third model, viNLI_R, was fine-tuned from PhoBERT pretrained-model on our $\text{VnNewsNLI}_\text{R}$ development set with Vietnamese word segmentation. We compared viNLI to viNLI_R for showing the effect of type 1 and type 2 samples in NLI datasets. We used Huggingface python library[21] for implementing the BERT architecture and fairseq python library[22] for tokenizing Vietnamese words into sub-words. We also used VnCoreNLP [23] for Vietnamese word segmentation before tokenization.

We fine-tuned these models in 2–8 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 tokens. In addition, the lengths of the premises and hypotheses are rarely greater than 100 syllables in our datasets. Other parameters were left with default settings. We chose the best models from checkpoints for testing.

Experiment Results

The results of the three models viXNLI, viNLI and viNLI_R on XNLI and VnNewsNLI test sets are shown in Table 6. We conducted this experiment to show the necessary of a Vietnamese native NLI training set for building Vietnamese NLI models. The results show that our Vietnamese native NLI

training set, VnNewsNLI, has improved the performance of our Vietnamese NLI model on Vietnamese native test set with the highest accuracy of 94.79% but it has not with the accuracy of 41.47% on Vietnamese translation of XNLI test set. Meanwhile, the Vietnamese translation of XNLI development set shows its role when viXNLI model has the accuracy of 68.64% but it does not when viXNLI model has the accuracy of 64.04% on VnNewsNLI test set. The reason of these results is that Vietnamese translation of XNLI did not preserve the writing style of Vietnamese texts and the premises and the hypotheses may be a group of sentences. In addition, this experiment also shows that the type 1 and type 2 samples have their important roles in building NLI models for recognizing the equivalent sentences through the accuracy of viNLI model (41.47% and 94.79%) in comparison to viNLI$_R$ model (37.62% and 74.54%) on the two test sets.

We evaluated the three models on a test set consisting of type 1 and type 2 samples of VnNewsNLI test set for more evident results. The results are shown in Table 7. The results of the viNLI model (accuracy of 95.67%) confirm that type 1 and type 2 samples are necessary in NLI datasets

Table 2 The statistics of NLI samples in VnNewsNLI and VnNewsNLI$_R$ dataset

Dataset	Samples	Entailment		Neutral		Contradiction	
	#*n*	#*n*	Rate	#*n*	Rate	#*n*	Rate
VnNewsNLI-dev	20,246	6756	33.37%	6754	33.36%	6736	33.27%
VnNewsNLI-test	11,878	3964	33.37%	3962	33.36%	3952	33.27%
VnNewsNLI$_R$-dev	10,115	3374	33.35%	3373	33.35%	3368	33.30%

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	1578	1808	1684	909	1079	994
Premises, 9–14	1786	1568	1672	1036	889	958
Premises, 15–20	601	598	572	299	285	260
Premises, 20–26	2232	2223	2216	1286	1276	1266
Premises, > 26	559	557	592	432	431	470
Hypotheses, ≤ 8	1814	1807	1684	1085	990	1077
Hypotheses, 9–14	1572	1569	1672	894	960	891
Hypotheses, 15–20	545	597	572	225	260	286
Hypotheses, 20–26	2198	2223	2216	1246	1268	1276
Hypotheses, > 26	627	558	592	512	470	430

Table 4 The distribution of the sentence length on entailment, neutral and contradiction. (ent. – entailment, neu. – neutral, con. – contradiction)

Length in syllable	Development set			Test set		
	ent. (%)	neu. (%)	con. (%)	ent. (%)	neu. (%)	con. (%)
Premises, ≤ 8	23.4	26.8	25.0	22.9	27.2	25.2
Premises, 9–14	26.4	23.2	24.8	26.1	22.4	24.3
Premises, 15–20	8.9	8.9	8.5	7.5	7.2	6.6
Premises, 20–26	33.0	32.9	32.9	32.5	32.2	32.1
Premises, > 26	8.3	8.2	8.8	10.9	10.9	11.9
Total	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**
Hypotheses, ≤ 8	26.9	26.8	25.0	27.4	25.1	27.2
Hypotheses, 9–14	23.3	23.2	24.8	22.6	24.3	22.5
Hypotheses, 15–20	8.1	8.8	8.5	5.7	6.6	7.2
Hypotheses, 20–26	32.5	32.9	32.9	31.4	32.1	32.2
Hypotheses, > 26	9.3	8.3	8.8	12.9	11.9	10.9
Total	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**	**100.0**

The highest values are in bold

Table 5 The 40 highest frequency words which are common nouns and verbs in VnNewsNLI dataset

Ord	Word	Ord	Word	Ord	Word	Ord	Word
1	Tổng thống *(President)*	11	An ninh *(Security)*	21	Chỉ trích *(Criticize)*	31	Thủ tướng *(Prime Minister)*
2	Vắc xin *(Vaccine)*	12	Quốc hội *(Congress)*	22	Tranh cử *(Run for Election)*	32	Trở thành *(Become)*
3	Bang *(State)*	13	Điều tra *(Investigate)*	23	Cáo buộc *(Allegate)*	33	Vượt *(Excess)*
4	Bầu cử *(Vote)*	14	Súng *(Gun)*	24	Nhậm chức *(Take office)*	34	Dịch *(Disease)*
5	Biểu tình *(Protest)*	15	Tấn công *(Attack)*	25	Công bố *(Publish)*	35	Luật *(Law)*
6	Ủng hộ *(Support)*	16	Nhằm *(Aim)*	26	Thành phố *(City)*	36	Ứng viên *(Candidate)*
7	Chống *(Against)*	17	Cảnh báo *(Warn)*	27	Yêu cầu *(Require)*	37	Người dân *(Citizen)*
8	Tuyên bố *(Declare)*	18	Bạo loạn *(Violence)*	28	Y tế *(Medical)*	38	Hoạt động *(Activity)*
9	Kêu gọi *(Call)*	10	Phiếu *(Vote)*	29	Tuổi *(Age)*	39	Mạng *(Life)*
10	Cảnh sát *(Police)*	20	Tên lửa *(Rocket)*	30	Quốc gia *(Nation)*	40	Xe *(Vehicle)*

to recognise the equivalent sentences that are special cases of entailment samples.

To show the usefulness of our Vietnamese NLI dataset, we also conducted an answer selection experiment on has-answer samples of UIT-viQuAD 2.0. The results of this experiment are shown in Table 8. In Table 8, the viNLI model has the highest P@1 score of 0.4949 indicating the ability to choose the most appropriate sentence in a short paragraph with a given sentence. This result is higher than the results of two baselines TF-IDF with P@1 score of 0.4056 and BM25 with P@1 score of 0.3833, showing that viNLI model is applicable in Vietnamese answer selection.

In our experiments, we fine-tuned the viXNLI model on a small development set with about 2500 samples and tested it on two larger test sets with about 5000 and 12,000 samples. The results show that BERT pre-train models are possibly fine-tuned on small datasets to build effective models [7].

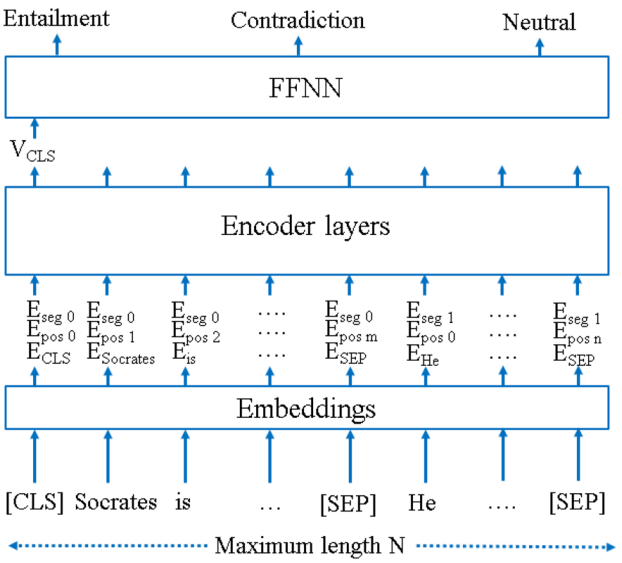

Fig. 6 The illustration of NLI BERT architecture [7]

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 aims at two issues. The first issue is the trained model's cue marks for identifying the relationship between a premise and a hypothesis without considering the premise. We addressed this issue by generating samples using eight types of premise-hypothesis pairs. 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 to reduce annotation costs. These samples were generated by applying a semi-automatic process. To evaluate our method, we built our VnNewsNLI dataset by extracting the title and the introductory sentence of many web pages in a Vietnamese news website VnExpress and applying our building process. When creating our VnNewsNLI, we had two people manually annotate each sentence to generate contraction sentences.

Table 6 The accuracy of viXNLI, viNLI and $viNLI_R$ models on test datasets

Model	Accuracy (%)	
	XNLI	VnNewsNLI
viXNLI	**68.64**	64.04
viNLI	41.47	**94.79**
$viNLI_R$	37.62	74.54

The highest values are in bold

Table 7 The accuracy of viXNLI, viNLI and $viNLI_R$ models on type 1 and type 2 samples of VnNewsNLI test set

Model	Accuracy (%) of type 1 and type 2 entailment
viXNLI	82.23
viNLI	**95.67**
$viNLI_R$	0.00

The highest values are in bold

Table 8 The P@1 scores of viXNLI, viNLI and $viNLI_R$ models on answer selection with two baselines TF-IDF and BM25

Model	P@1
viXNLI	0.4044
viNLI	**0.4949**
$viNLI_R$	0.1733
TF-IDF	0.4056
BM25	0.3833

The highest values are in bold

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 (94.79% vs. 64.04%) on our VnNewsNLI test set while it had a lower accuracy (41.47% vs. 68.64%) on the Vietnamese XNLI test set when compared to viXNLI. To show the usefulness of our NLI dataset, we also conducted an answer selection experiment using viXNLI model, viNLI model and two baselines TF-IDF and BM25. The accuracy of 94.79% and the highest P@1 score of 0.4949 of viNLI model in the two experiments promised to build a high-quality Vietnamese NLI dataset from Vietnamese documents to ensure writing style.

Currently, our VnNewsNLI dataset contains a pretty small number of samples, with about 32,000 samples. In future, we will apply our proposed process for building a large and high-quality multi-genre Vietnamese NLI dataset. We will also train a Vietnamese NLI model to help develop our dataset by automatically suggesting the relation of a premise-hypothesis pair. This model might reduce our effort in building our dataset.

Declarations

Conflict of Interest The authors declare that they have no conflict of interest.

References

1. Punyakanok V, Roth D, Yih W-T. Natural language inference via dependency tree mapping: an application to question answering. Comput Linguist. 2004;6:10.
2. Lan W, Xu W. Neural network models for paraphrase identification, semantic textual similarity, natural language inference, and question answering. International Conference on Computational Linguistics, pp. 3890--3902. Association for Computational Linguistics, Santa Fe, New Mexico, USA (2018)
3. Minbyul J, Mujeen S, Gangwoo K, Donghyeon K, Wonjin Y, Jaehyo Y, Jaewoo K. Transferability of natural language inference to biomedical question answering. Conference and Labs of the Evaluation Forum, Thessaloniki, Greece (2020)
4. Falke T, Ribeiro LFR, Utama PA, Dagan I, Gurevych I. Ranking generated summaries by correctness: An interesting but challenging application for natural language inference. 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. Workshop on New Frontiers in Summarization, pp. 27--32. Association for Computational Linguistics, Copenhagen, Denmark (2017)
6. Dagan I, Roth D, Sammons M, Zanzotto FM. Recognizing textual entailment: models and applications. Morgan & Claypool Publishers (2013)
7. Devlin J, Chang M-W, Lee K, Toutanova K. BERT: pre-training of deep bidirectional transformers for language understanding. 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. International Conference on Language Resources and Evaluation, pp. 216--223. European Language Resources Association, Reykjavik, Iceland (2014)
9. Bowman SR, Angeli G, Potts C, Manning CD. A large annotated corpus for learning natural language inference. Conference on Empirical Methods in Natural Language Processing, pp. 632--642. Association for Computational Linguistics, Lisbon, Portugal (2015)
10. Williams A, Nangia N, Bowman SR. A broad-coverage challenge corpus for sentence understanding through inference. 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, Louisiana (2017)
11. Conneau A, Rinott R, Lample G, Williams A, Bowman S, Schwenk H, Stoyanov V. XNLI: Evaluating cross-lingual sentence representations. Conference on Empirical Methods in Natural Language Processing, pp. 2475--2485. Association for Computational Linguistics, Brussels, Belgium (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. International Conference on Knowledge and Systems Engineering, pp. 108--113. IEEE, Ho Chi Minh City, Vietnam (2015)

13. Nguyen DQ, Nguyen AT. PhoBERT: pre-trained language models for Vietnamese. Conference on Empirical Methods in Natural Language; 2020. pp. 1037—1042.
14. Jiang N, de Marneffe M-C. Evaluating BERT for natural language inference: a case study on the CommitmentBank. Conference on Empirical Methods in Natural Language Processing, pp. 6086--6091. Association for Computational Linguistics, Hong Kong, China (2019).
15. Nguyen CT, Nguyen DT. Building a Vietnamese dataset for natural language inference models. Future Data and Security Engineering; 2021, pp. 185--199. Springer, Singapore.
16. Nguyen KV, Tran SQ, Nguyen LT, Huynh TV, Luu ST, Nguyen NL-T. VLSP 2021 Shared Task: Vietnamese Machine reading comprehension. Kiet Van Nguyen, The 8th International Workshop on Vietnamese Language and Speech Processing (VLSP 2021) (2021).
17. Nguyen KV, Nguyen V, Nguyen A, Nguyen N. A Vietnamese dataset for evaluating machine reading comprehension. International Conference on Computational Linguistics, pp. 2595--2605. International Committee on Computational Linguistics, Barcelona, Spain (Online) (2020)
18. 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, Italy (2019)
19. Rogers A, Kovaleva O, Rumshisky A. A primer in BERTology: What we know about how BERT works. Trans Assoc Comput Linguistics. 2020;8:842–66.
20. Peters ME, Neumann M, Zettlemoyer L, Yih W-T. Dissecting contextual word embeddings: architecture and representation. Conference on Empirical Methods in Natural Language Processing, pp. 1499--1509. Association for Computational Linguistics, Brussels, Belgium (2018)
21. Wolf T, Chaumond J, Debut L, Sanh V, Delangue C, Moi A, Cistac P, Funtowicz M, Davison J, Shleifer S, others. Transformers: state-of-the-art natural language processing. Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pp. 38--45. Association for Computational Linguistics (2020).
22. Ott M, Edunov S, Baevski A, Fan A, Gross S, Ng N, Grangier D, Auli M. fairseq: a fast, extensible toolkit for sequence modeling. Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations), pp. 48--53. Association for Computational Linguistics, Minneapolis, Minnesota (2019)
23. Vu T, Nguyen DQ, Nguyen DQ, Dras M, Johnson M. VnCoreNLP: a vietnamese natural language processing toolkit. Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations, pp. 56--60. Association for Computational Linguistics, New Orleans, Louisiana (2018).