Building a Semantic Role Labelling System for Vietnamese

Conference Paper · October 2015

DOI: 10.1109/ICDIM.2015.7381877

3 authors:

Hoang Thai Pham
FPT University
2 PUBLICATIONS **0** CITATIONS
SEE PROFILE

Khoai Xuan Pham
FPT University
2 PUBLICATIONS **0** CITATIONS
SEE PROFILE

Le-Hong Phuong
Vietnam National University, Hanoi
31 PUBLICATIONS **147** CITATIONS
SEE PROFILE

Building a Semantic Role Labelling System for Vietnamese

Thai-Hoang Pham
FPT University
hoangptse02637@fpt.edu.vn

Xuan-Khoai Pham
FPT University
khoaipxse02933@fpt.edu.vn

Phuong Le-Hong
Hanoi University of Science
phuonglh@vnu.edu.vn

***Abstract*—Semantic role labelling (SRL) is a task in natural language processing which detects and classifies the semantic arguments associated with the predicates of a sentence. It is an important step towards understanding the meaning of a natural language. There exists SRL systems for well-studied languages like English, Chinese or Japanese but there is not any such system for the Vietnamese language. In this paper, we present the first SRL system for Vietnamese with encouraging accuracy. We first demonstrate that a simple application of SRL techniques developed for English could not give a good accuracy for Vietnamese. We then introduce a new algorithm for extracting candidate syntactic constituents, which is much more accurate than the common node-mapping algorithm usually used in the identification step. Finally, in the classification step, in addition to the common linguistic features, we propose novel and useful features for use in SRL. Our SRL system achieves an F_1 score of 73.53% on the Vietnamese PropBank corpus. This system, including software and corpus, is available as an open source project and we believe that it is a good baseline for the development of future Vietnamese SRL systems.**

I. Introduction

SRL is the task of identifying semantic roles of predicates in the sentence. In particular, it answers a question *Who did What to Whom, When, Where, Why?*. A simple Vietnamese sentence *Nam giúp Huy học bài vào hôm qua* (Nam helped Huy to do homework yesterday) is given in Figure 1.

$$\underbrace{\text{Nam}}_{Who}\ giúp\ \underbrace{\text{Huy}}_{Whom}\ \underbrace{\text{học bài}}_{What}\ \underbrace{\text{vào hôm qua}}_{When}$$

Figure 1: An example sentence

To assign semantic roles for the sentence above, we must analyse and label the propositions concerning the predicate *giúp* (helped) of the sentence. Figure 2 shows a result of the SRL for this example, where meaning of the labels will be described in detail in Section IV.

$$\underbrace{\text{Nam}}_{Arg0}\ giúp\ \underbrace{\text{Huy}}_{Arg1}\ \underbrace{\text{học bài}}_{Arg2}\ \underbrace{\text{vào hôm qua}}_{ArgM-TMP}$$

Figure 2: Semantic roles for the example sentence

SRL has been used in many natural language processing (NLP) applications such as question answering [1], machine translation [2], document summarization [3] and information extraction [4]. Therefore, SRL is an important task in NLP.

The first SRL system was developed by Gildea and Jurafsky [5]. This system was performed on the FrameNet corpus and was used for English. After that, SRL task has been widely researched by the NLP community. In particular, there have been two shared-tasks, CoNLL-2004 [6] and CoNLL-2005 [7], focusing on SRL task for English. Most of the systems participating in these share-tasks treated this problem as a classification problem and applied some supervised machine learning techniques. In addition, there were some systems developed for other languages such as Chinese [8] or Japanese [9].

In this paper, we present the first SRL system for Vietnamese with encouraging accuracy. We first demonstrate that a simple application of SRL techniques developed for English or other languages could not give a good accuracy for Vietnamese. In particular, in the constituent identification step, the widely used 1-1 node-mapping algorithm for extracting argument candidates performs poorly on the Vietnamese dataset, having F_1 score of 35.84%. We thus introduce a new algorithm for extracting candidates, which is much more accurate, achieving an F_1 score of 83.63%.

In the classification step, in addition to the common linguistic features, we propose novel and useful features for use in SRL, including function tags and word clusters obtained by performing a Gaussian mixture analysis on the distributed representations of Vietnamese words. These features are employed in two statistical classification models, Maximum Entropy and Support Vector Machines, which are proved to be good at many classification problems.

Our SRL system achieves an F_1 score of 73.53% on the Vietnamese PropBank corpus. This system, including software and corpus, is available as an open source project and we believe that it is a good baseline for the development of future Vietnamese SRL systems.

The paper is structured as follows. Section II introduces briefly the SRL task and two well-known corpora for English. Section III describes the methodologies of some existing systems and of our system. Some difficulties of SRL for Vietnamese are also discussed. Section IV presents the evaluation results and discussion. Finally, Section V concludes the paper and suggests some directions for future work.

Nam *giúp* Huy học bài vào hôm qua

Figure 3: Example of identification task

II. Background

A. SRL Task Description

The SRL task is usually divided into two steps. The first step is argument identification. The goal of this step is to identify the syntactic constituents of a sentence which are the most likely to be semantic arguments of its predicates. This is a difficult problem since the number of constituent candidates is exponentially large, especially for long sentences.

The second step is argument classification which decides the exact semantic role for each constituent candidate identified in the first task. For example, the identification step of the sentence in the previous example *Nam giúp Huy học bài vào hôm qua* is described in Figure 3 and in the classification task, semantic roles are labelled as shown Figure 2.

B. Existing Corpora for SRL

1) FrameNet: The FrameNet project is a lexical database of English. It was built by annotating examples of how words are used in actual texts. It consists of more than 10,000 word senses, most of them with annotated examples that show the meaning and usage and more than 170,000 manually annotated sentences [10]. This is the most widely used dataset upon which SRL systems for English have been developed and tested.

FrameNet is based on the Frame Semantics theory [11]. The basic idea is that the meanings of most words can be best understood on the basis of a semantic frame: a description of a type of event, relation, or entity and the participants in it. All members in semantic frames are called frame elements. For example, a sentence in FrameNet is annotated in cooking concept as shown in Figure 4.

The boy *grills* their catches on an open fire
Cook *Food* *Heating−instrument*

Figure 4: An example sentence in the FrameNet corpus

2) PropBank: PropBank is a corpus that is annotated with verbal propositions and their arguments [12]. PropBank tries to supply a general purpose labelling of semantic roles for a large corpus to support the training of automatic semantic role labelling systems. However, defining such a universal set of semantic roles for all types of predicates is a difficult task; therefore, only Arg0 and Arg1 semantic roles can be generalized. In addition to the core roles, PropBank defines several adjunct roles that can apply to any verb. It is called Argument Modifier. The semantic roles covered by the PropBank are the following:

- **Core Arguments** (Arg0-Arg5, ArgA): Arguments define predicate specific roles. Their semantics depend on predicates in the sentence.

The boy *grills* their catches on an open fire
Arg0 *Arg1* *Arg2*

Figure 5: An example sentence in the PropBank corpus

- **Adjunct Arguments** (ArgM-): General arguments that can belong to any predicate. There are 13 types of adjuncts.
- **Reference Arguments** (R-): Arguments represent arguments realized in other parts of the sentence.
- **Predicate** (V): Participant realizing the verb of the proposition.

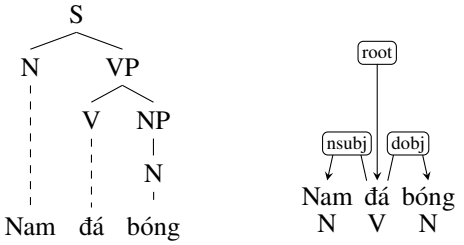
For example, the sentence of Figure 4 can be annotated in the PropBank role schema as shown in Figure 5.

III. Methodology

A. Existing Approaches

This section summarizes existing approaches used by typical SRL systems for well-studied languages. We describe these systems by investigating two aspects, namely data type that the systems use and their strategies for labelling semantic roles, including model types, labelling strategies and degrees of granularity.

1) Data Types: There are some kinds of data used in the training of SRL systems. Some systems use bracketed trees as the input data. A bracketed tree of a sentence is the tree of nested constituents representing its constituency structure. Some systems use dependency trees of a sentence, which represents dependencies between individual words of a sentence. The syntactic dependency represents the fact that the presence of a word is licensed by another word which is its governor. In a typed dependency analysis, grammatical labels are added to the dependencies to mark their grammatical relations, for example *nominal subject* (nsubj) or *direct object* (dobj). Figure 6 shows the bracketed tree and the dependency tree of an example sentence.



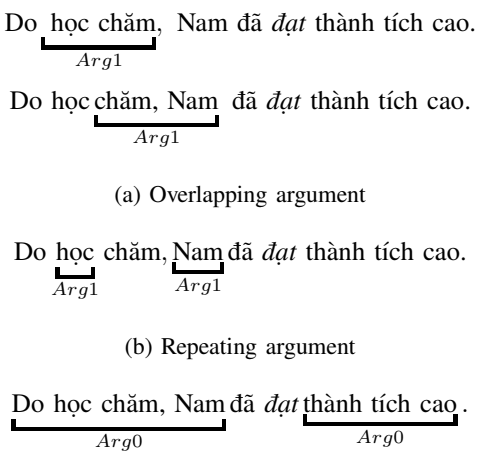
(a) The bracketed tree (b) The dependency tree

Figure 6: Bracketed and dependency trees for sentence *Nam đá bóng* (Nam plays football)

2) SRL Strategy:

a) Model Types: There are two types of classification models: Independent Model and Joint Model. While independent model decides the label of each argument's candidate independently of other candidates, joint model finds the best

overall labelling for all candidates in the sentence. Independent model runs fast but are prone to inconsistencies. For example, Figure 7 shows some typical inconsistencies, including overlapping arguments, repeating arguments and missing arguments of a sentence *Do học chăm, Nam đã đạt thành tích cao* (By studying hard, Nam got a high achievement).



(a) Overlapping argument

(b) Repeating argument

(c) Missing argument

Figure 7: An example of inconsistencies

b) Labelling Stategies: Strategies for labelling semantic roles are diverse, but we can summarize that there are three main strategies. Most of the systems use a two-step approach consisting of identification and classification [13], [14]. The first step identifies arguments from many candidates. It is essentially a binary classification problem. The second step classifies these arguments into particular semantic roles. Some systems use single classification step by adding a "null" label into semantic roles, denoting that this is not an argument [15]. Other systems consider SRL as a sequence tagging [16], [17].

c) Granularity: Existing SRL systems use different degrees of granularity when considering constituents. Some systems use individual words as their input and perform sequence tagging to identify arguments. This method is called Word-by-Word (W-by-W) approach. Other systems directly take syntactic phrases as input constituents. This method is called Constituent-by-Constituent (C-by-C) approach.

Compared to the W-by-W approach, C-by-C approach has several advantages. First, phrase boundaries are usually consistent with argument boundaries. Second, C-by-C approach allows us to work with larger contexts due to a smaller number of candidates in comparison to the W-by-W approach.

B. Our Approach

The previous subsection has reviewed existing techniques for SRL which have been published so far for well-studied languages. In this section, we first show that these techniques per se cannot give a good result for Vietnamese SRL, due to some inherent difficulties, both in terms of language characteristics and of the available corpus. We then develop a new algorithm for extracting candidate constituents for use in the identification step.

Some difficulties of Vietnamese SRL are related to its SRL corpus. We use the Vietnamese PropBank [18] in the development of our SRL system.[1] This SRL corpus has 5,000 annotated sentences, which is much smaller than SRL corpora of other languages. For example, the English PropBank contains about 50,000 sentences, which is ten times larger. While smaller in size, the Vietnamese PropBank has more semantic roles than the English PropBank has – 25 roles compared to 21 roles. This makes the unavoidable data sparseness problem more severe for Vienamese SRL than for English SRL.

In addition, our extensive inspection and experiments on the Vietnamese PropBank have uncovered that this corpus has many annotation errors, largely due to encoding problems and inconsistencies in annotation. In many cases, we have to fix these annotation errors by ourselves. In other cases where only a proposition of a complex sentence is incorrectly annotated, we perform an automatic preprocessing procedure to drop it out, leave the correctly annotated propositions untouched. We finally come up with a corpus of 4,800 sentences which are semantic role annotated. This corpus will be released for free use for research purpose.

A major difficulty of Vietnamese SRL is due to the nature of the language, where its linguistic characteristics are different from occidental languages [19]. We first try to apply the common node-mapping algorithm which are widely used in English SRL systems to the Vietnamese corpus. However, this application gives us a very poor performance. Therefore, in the identification step, we develop a new algorithm for extracting candidate constituents which is much more accurate for Vietnamese than the node-mapping algorithm. Details of experimental results will be provided in the Section IV

In order to improve the accuracy of the classification step, and hence of our SRL system as a whole, we have integrated many useful features for use in two statistical classification models, namely Maximum Entropy (ME) and Support Vector Machines (SVM). On the one hand, we adapt the features which have been proved to be good for SRL of English. On the other hand, we propose some novel features, including function tags and word clusters.

In the next paragraph, we present our constituent extraction algorithm for the identification step. Details of the features for use in the classification step will be presented in Section IV.

1) Constituent Extraction Algorithm: This algorithm aims to extract constituents from a bracketed tree which are associated to their corresponding predicates of the sentence. If the sentence has multiple predicates, multiple constituent sets corresponding to the predicates are extracted. Pseudo code of the algorithm is described in Algorithm 1.

This algorithm uses several simple functions. The $root()$ function gets the root of a tree. The $children()$ function gets the children of a node. The $sibling()$ function gets the sisters

[1]To our knowledge, this is the first SRL corpus for Vietnamese which has been published for free research.

Algorithm 1: Constituent Extraction Algorithm

input : A bracketed tree T and its predicate
output: A tree with constituents for the predicate
begin
  $currentNode \leftarrow predicateNode$
  while $currentNode \neq$ *T.root()* **do**
    for $S \in currentNode.sibling()$ **do**
      if $|S.children()| > 1$ *and* $S.children().get(0).isPhrase()$ **then**
        $sameType \leftarrow true$
        $diffTag \leftarrow true$
        $phraseType \leftarrow S.children().get(0).phraseType()$
        $funcTag \leftarrow S.children().get(0).functionTag()$
        for $i \leftarrow 1$ **to** $|S.children()| - 1$ **do**
          if $S.children().get(i).phraseType() \neq phraseType$ **then**
            $sameType \leftarrow false$
            break
          if $S.children().get(i).functionTag() = funcTag$ **then**
            $diffTag \leftarrow false$
            break
        if $sameType$ *and* $diffTag$ **then**
          for $child \in S.children()$ **do**
            $T.collect(child)$
      else
        $T.collect(S)$
    $currentNode \leftarrow currentNode.parent()$
  return T

of a node. The $isPhrase()$ function checks whether a node is of phrasal type or not. The $phraseType()$ function and $functionTag()$ function extracts the phrase type and function tag of a node, respectively. Finally, the $collect(node)$ function collects words from leaves of the subtree rooted at a node and creates a constituent.

Figure 8 shows an example of running the algorithm on a sentence *Bà nói nó là con trai tôi mà* (She said that he is my son). First, we find the current predicate node V-H *là* (is). The current node has only one sibling NP. This node has one child, so its associated words are collected. After that, we set current node to its parent and repeat the process until reaching the root of the tree. Finally, we obtain a tree with constituents: *Bà*, *nói*, *nó*, and *con trai tôi mà*.

2) Our SRL System: Our SRL system is developed on the Vietnamese PropBank. It thus operates on fully bracketed trees. We employ ME and SVM as classifiers. Its classification model is of type independent and its input are C-by-C.

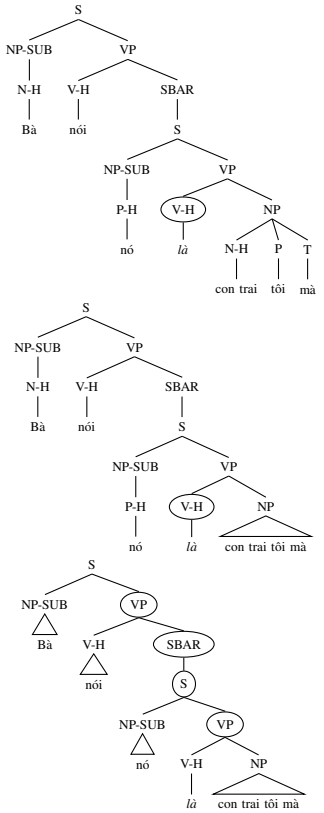


Figure 8: Extracting constituents of the sentence "Bà nói nó là con trai tôi mà" at predicate "*là*"

IV. Experiment

In this section, we first introduce the Vietnamese PropBank upon which our SRL system has been trained and tested. We then propose two feature sets in use. Finally, we present and discuss experimental results.

A. Dataset

We conduct experiments on the Vietnamese PropBank [18] containing about 5,460 sentences which are manually annotated with semantic roles. This corpus has a similar annotation schema to the English PropBank. Due to some inconsistency annotation errors of the corpus, notably in many complex sentences, we were not able to use all the corpus in our experiments. We focus ourselves in simple sentences which have only one predicate rather than complex sentences with multiple predicates. After extracting sentences, we have a corpus of about 4,860 simple sentences which are annotated with semantic roles.

The semantic roles covered by the Vietnamese PropBank are the following:

- **Core Arguments** (Arg0-Arg4): Arguments define predicate specific roles. These core arguments are similar to those of the English PropBank, however, there are 5 roles instead of 7, compared to the English PropBank.
- **Adjunct Arguments** (ArgM-): There are 20 types of adjuncts, as listed in Table I.

Role Name	Description	Role Name	Description
ArgM-ADV	general-purpose	ArgM-CAU	cause
ArgM-DIS	discourse marker	ArgM-DIR	direction
ArgM-NEG	negation marker	ArgM-MNR	manner
ArgM-PRD	predication	ArgM-PRP	purpose
ArgM-MOD	modal verb	ArgM-TMP	temporal
ArgM-REC	reciprocal	ArgM-GOL	goal
ArgM-LVB	light verb	ArgM-EXT	extent
ArgM-COM	comitative	ArgM-I	interjection
ArgM-Partice	partice	ArgM-PNC	purpose
ArgM-ADJ	unknown	ArgM-RES	unknown

Table I: Adjunct arguments in Vietnamese

- **Predicate** (V): In Vietnamese, a predicate is not only a verb, but it could be also a noun, an adjective or a preposition.

B. Feature Sets

We use two feature sets in this study. The first one is composed of basic features which are commonly used in SRL system for English. This feature set is used in the SRL system of Gildea and Jurafsky on the FrameNet corpus [5].

1) Basic Feature Set: This feature set consists of 6 feature templates, as follows:

1) **Phrase Type**: This is very useful feature in classifying semantic roles because different roles tend to have different syntactic categories. For example, in the sentence in Figure 8 *Bà nói nó là con trai tôi mà*, the phrase type of constituent *nó* is *NP*.
2) **Parse Tree Path**: This feature captures the syntactic relation between a constituent and a predicate in a bracketed tree. This is the shortest path from a constituent node to a predicate node in the tree. We use either symbol ↑ or symbol ↓ to indicate the upward direction or the downward direction, respectively. For example, the parse tree path from constituent *nó* to the predicate *là* is *NP*↑*S*↓*VP*↓*V-H*.
3) **Position**: Position is a binary feature that describes whether the constituent occurs after or before the predicate. It takes value *0* if the constituent appears before the predicate in the sentence or value *1* otherwise. For example, the position of constituent *nó* in Figure 8 is *0* since it appears before predicate *là*.
4) **Voice**: Sometimes, the differentiation between active and passive voice is useful. For example, in an active sentence, the subject is usually an *Arg0* while in a passive sentence, it is often an *Arg1*. Voice feature is also binary feature, taking value *1* for active voice or *0* for passive voice. The sentence in Figure 8 is of active voice, thus its voice feature value is *1*.
5) **Head Word**: This is the first word of a phrase. For example, the head word for the phrase *con trai tôi mà* is *con trai*.
6) **Subcategorization**: Subcategorization feature captures the tree that has the concerned predicate as its child. For example, in Figure 8, the subcategorization of the predicate *là* is *VP(V-H, NP)*.

2) Modified Features and New Features: Preliminary investigations on the basic feature set give us a rather poor result. Therefore, we propose some modified features and novel features so as to improve the accuracy of the system. These features are as follows:

1) **Function Tag**: Function tag is a useful information, especially for classifying adjunct arguments. It determines a constituent's role, for example, the function tag of constituent *nó* is *SUB*, indicating that this has a subjective role.
2) **Partial Parse Tree Path**: Many sentences have complicated structure. It can make parse tree path very long and infrequent. We propose to cut a path from the lowest common ancestor to its predicate, instead of using the full path. For example, the partial path from the constituent *nó* to the predicate *là* in Figure 8 is *NP*↑*S*.
3) **Distance**: This feature records the length of the full parse tree path before pruning. This feature helps retaining some information that might be lost when a partial path, instead of a full path, is used. For example, the distance from constituent *nó* to the predicate *là* is *3*.
4) **Predicate Type**: Unlike in English, the type of predicates in Vietnamese is much more complicated. It is not only a verb, but is also a noun, an adjective, or a preposition. Therefore, we propose a new feature which captures predicate types. For example, the predicate type of the concerned predicate is *V-H*.
5) **Word Cluster**: Word clusters have been shown to help improve the performance of many NLP tasks because they alleviate the severity of the data sparseness problem. Thus, in this work we propose to use word cluster features. We first produce distributed word representations (or word embeddings) of Vietnamese words, where each word is represented by a dense, real-valued vector of 50 dimensions, by using a Skip-gram model described in [20], [21]. We then cluster these word vectors into 128 groups using a Gaussian mixture model.[2] A word cluster feature is defined as the cluster identifier of the concerned word.

C. Results and Discussions

1) Evaluation Method: We use a 10-fold cross-validation method to evaluate our system. The final accuracy scores is the average scores of the 10 runs.

The evaluation metrics are the precision, recall and F_1-measure. The precision (P) is the proportion of labelled arguments identified by the system which are correct; the recall (R) is the proportion of labelled arguments in the gold results which are correctly identified by the system; and the F_1-measure is the harmonic mean of P and R, that is $F_1 = 2PR/(P + R)$.

2) Baseline System: In the first experiment, we compare our constituent extraction algorithm to the 1-1 node mapping

[2]Actually, there is an additional group for unknown words.

	1-1 Node Mapping	Our Extraction Alg.
Precision	29.53%	81.00%
Recall	45.60%	86.43%
F1	35.84%	83.63%

Table II: Accuracy of two extraction algorithms

algorithm. Table II shows the performance of two extraction algorithms.

We see that our extraction algorithm outperforms significantly the 1-1 node mapping algorithm, in both of the precision and the recall ratios. In particular, the precision of the 1-1 node mapping algorithm is only 29.53%; this means that this method captures many candidates which are not arguments. In contrast, our algorithm is able to identify a large number of correct argument candidates, particularly with the recall ratio of 86.43%. This result clearly demonstrates that we cannot take for granted that a good algorithm for English could also work well for another language of different characteristics.

In the second experiment, we continue to compare the performance of the two extraction algorithms, this time at the final classification step and get the baseline for Vietnamese SRL. The classifier we use in this experiment is a Maximum Entropy classifier.[3] Table III shows the accuracy of the baseline system.

	1-1 node mapping	Our Extraction Alg.
Precision	52.80%	53.79%
Recall	3.30%	47.51%
F1	6.20%	50.45%

Table III: Accuracy of baseline system

One again, this result confirms that our algorithm is much superior than the 1-1 node mapping algorithm. The F_1 of our baseline SRL system is 50.45%, compared to 6.20% of the 1-1 node mapping system. This result can be explained by the fact that the 1-1 node mapping algorithm has a very low recall ratio, because it identifies incorrectly many argument candidates.

3) Labelling Strategy: In the third experiment, we compare two labelling strategies for Vietnamese SRL (cf. Section III). In addition to the ME classifier, we also try the Support Vector Machine (SVM) classifier, which usually gives good accuracy in a wide variety of classification problems.[4] Table IV shows the F_1 scores of different labelling strategies.

	ME	SVM
1-step strategy	50.45%	68.91%
2-step strategy	49.76%	68.55%

Table IV: Accuracy of two labelling strategies

[3]We use the logistic regression classifier with L_2 regularization provided by the `scikit-learn` software package. The regularization term is fixed at 1.

[4]We use a linear SVM provided in the `scikit-learn` software package with default parameter values.

We see that the SVM classifier outperforms ME the classifier by a large margin. The best accuracy is obtained by using 1-step stragegy with SVM classifier. The current SRL system achieves an F_1 score of 68.91%.

4) Feature Analysis: In the fourth experiment, we analyse and evaluate the impact of each individual feature to the accuracy of our system so as to find the best feature set for our Vietnamese SRL system. We start with the basic feature set presented previously, denoted by Φ_0 and augment it with modified and new features as shown in Table V. The accuracy of these feature sets are shown in Table VI.

Feature Set	Description
Φ_1	$\Phi_0 \cup$ {Function Tag}
Φ_2	$\Phi_0 \cup$ {Predicate Type}
Φ_3	$\Phi_0 \cup$ {Distance}

Table V: Feature sets

Feature Set	Precision	Recall	F1
Φ_0	72.27%	65.84%	68.91%
Φ_1	**76.49%**	**69.65%**	**72.91%**
Φ_2	72.26%	65.87%	68.92%
Φ_3	72.35%	65.86%	68.95%

Table VI: Accuracy of feature sets in Table V

We notice that amongst the three features, function tag is the most important feature which increases the accuracy of the baseline feature set by about 4% of F_1 score. The distance feature also helps increase slightly the accuracy. We thus consider the fourth feature set Φ_4 defined as

$$\Phi_4 = \Phi_0 \cup \{\text{Function Tag}\} \cup \{\text{Distance}\}.$$

In the fifth experiment, we modify the feature set Φ_4 by replacing the predicate with its cluster and similarly, replacing the head word with its cluster, replacing the full path with its partial path, resulting in feature sets Φ_5, Φ_6, and Φ_7 respectively (see Table VII). The accuracy of these feature sets are shown in Table VIII.

Feature Set	Description
Φ_5	$\Phi_4 \backslash$ {Predicate} $\cup$ {Predicate Cluster}
Φ_6	$\Phi_4 \backslash$ {Head Word} $\cup$ {Head Word Cluster}
Φ_7	$\Phi_4 \backslash$ {Full Path} $\cup$ {Partial Path}

Table VII: Feature sets (continued)

Feature Set	Precision	Recall	F1
Φ_4	76.60%	69.72%	73.00%
Φ_5	**76.86%**	**70.36%**	**73.47%**
Φ_6	72.50%	66.59%	69.41%
Φ_7	76.29%	69.58%	72.78%

Table VIII: Accuracy of feature sets in Table VII

Feature Set	Description
Φ_9	$\Phi_8 \backslash$ {Function Tag}
Φ_{10}	$\Phi_8 \backslash$ {Predicate Cluster}
Φ_{11}	$\Phi_8 \backslash$ {Head Word}
Φ_{12}	$\Phi_8 \backslash$ {Path}
Φ_{13}	$\Phi_8 \backslash$ {Position}
Φ_{14}	$\Phi_8 \backslash$ {Voice}
Φ_{15}	$\Phi_8 \backslash$ {Subcategorization}
Φ_{16}	$\Phi_{10} \cap \Phi_{15}$

Table IX: Feature sets (continued)

We observe that using the predicate cluster instead of the predicate itself helps improve the accuracy of the system by about 0.47% of F_1 score. For ease of later presentation, we rename the feature set Φ_5 as Φ_8.

In the sixth experiment, we investigate the significance of individual features to the system by removing them, one by one from the feature set Φ_8. By doing this, we can evaluate the importance of each feature to our overall system. The feature sets and their corresponding accuracy are presented in Table IX and Table X respectively.

Feature Set	Precision	Recall	F1
Φ_8	76.86%	70.36%	73.47%
Φ_9	72.27%	66.12%	69.06%
Φ_{10}	**76.87%**	**70.41%**	**73.50%**
Φ_{11}	72.91%	67.05%	69.86%
Φ_{12}	76.81%	70.36%	73.44%
Φ_{13}	76.41%	70.21%	73.18%
Φ_{14}	76.85%	70.36%	73.46%
Φ_{15}	**76.83%**	**70.51%**	**73.53%**
Φ_{16}	76.70%	70.31%	73.36%

Table X: Accuracy of feature sets in Table IX

We see that the accuracy increases slightly when either the predicate cluster feature (Φ_{10}) or the subcategorization feature (Φ_{15}) is removed. However, removing both of the two features (Φ_{16}) makes the accuracy decrease. For this reason, we remove only the subcategorization feature. The best feature set includes the following features: predicate cluster, phrase type, function tag, parse tree path, distance, voice, position and head word. The best accuracy of our system is 73.53% of F_1 score.

5) Learning Curve: In the last experiment, we investigate the dependence of accuracy to the size of the training dataset. Figure 9 depicts the learning curve of our system when the data size is varied.

It seems that the accuracy of our system improves only slightly starting from the dataset of about 2,000 sentences. Nevertheless, the curve has not converged, indicating that the system could achieve a better accuracy when a larger dataset is available.

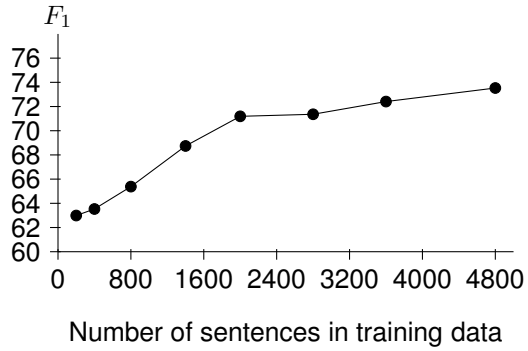


Figure 9: Learning Curve

V. Conclusion

In this paper, we have presented the first system for Vietnamese semantic role labelling. Our system achieves a good accuracy of about 73.5% of F_1 score in the Vietnamese PropBank.

We have argued that one cannot assume a good applicability of existing methods and tools developed for English and other Western languages and that they may not offer a cross-language validity. For an isolating language such as Vietnamese, techniques developed for inflectional languages cannot be applied "as is". In particular, we have developed an algorithm for extracting argument candidates which has a better accuracy than the 1-1 node mapping algorithm. We have proposed some novel features which are proved to be useful for Vietnamese SRL, notably predicate clusters and function tags. Our SRL system, including software and corpus, is available as an open source project for free research purpose and we believe that it is a good baseline for the development and comparison of future Vietnamese SRL systems.

In the future, we plan to improve further our system, in the one hand, by enlarging our corpus so as to provide more data for the system. On the other hand, we would like to investigate different models used in SRL, for example joint models [14] and recent inference techniques, such as integer linear programming [22], [23].

Acknowledgements

This work was supported by Vietnam National Foundation for Science and Technology Development (NAFOSTED Project No. 102.05-2014.28). We would also like to thank the FPT Technology Research Institute for providing us the corpora for use in the experiments.

References

[1] D. Shen and M. Lapata, "Using semantic roles to improve question answering," in *Proceedings of Conference on Empirical Methods on Natural Language Processing and Computational Natural Language Learning*, Prague, Czech Republic, 2007, pp. 12–21.

[2] C. kiu Lo and D. Wu, "Evaluating machine translation utility via semantic role labels." in *Proceedings of The International Conference on Language Resources and Evaluation*, Valletta, Malta, 2010, pp. 2873–2877.

[3] C. Aksoy, A. Bugdayci, T. Gur, I. Uysal, and F. Can, "Semantic argument frequency-based multi-document summarization," in *Proceedings of the 24th of the International Symposium on Computer and Information Sciences*, Guzelyurt, Turkey, 2009, pp. 460–464.

[4] J. Christensen, S. Soderland, O. Etzioni *et al.*, "Semantic role labeling for open information extraction," in *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics – Human Language Technologies*, Los Angeles, CA, USA, 2010, pp. 52–60.

[5] D. Gildea and D. Jurafsky, "Automatic labeling of semantic roles," *Computational Linguistics*, vol. 28, no. 3, pp. 245–288, 2002.

[6] X. Carreras and L. Màrquez, "Introduction to the CoNLL-2004 shared task: semantic role labeling," in *Proceedings of the 8th Conference on Computational Natural Language Learning*, Boston, MA, USA, 2004.

[7] ——, "Introduction to the CoNLL-2005 shared task: semantic role labeling," in *Proceedings of the 9th Conference on Computational Natural Language Learning*, Ann Arbor, MI, USA, 2005, pp. 152–164.

[8] N. Xue and M. Palmer, "Automatic semantic role labeling for Chinese verbs," in *Proceedings of International Joint Conferences on Artificial Intelligence*, Edinburgh, Scotland, UK, 2005, pp. 1160–1165.

[9] H. Tagami, S. Hizuka, and H. Saito, "Automatic semantic role labeling based on Japanese FrameNet–A Progress Report," in *Proceedings of Conference of the Pacific Association for Computational Linguistics*, Hokkaido University, Sapporo, Japan, 2009, pp. 181–186.

[10] C. F. Baker, C. J. Fillmore, and B. Cronin, "The structure of the FrameNet database," *International Journal of Lexicography*, vol. 16, no. 3, pp. 281–296, 2003.

[11] H. C. Boas, "From theory to practice: Frame semantics and the design of Framenet," in *Semantisches Wissen im Lexikon*. Tübingen: Narr., 2005, pp. 129–160.

[12] O. Babko-Malaya, "PropBank annotation guidelines," Colorado University, Tech. Rep., 2005.

[13] P. Koomen, V. Punyakanok, D. Roth, and W. tau Yih, "Generalized inference with multiple semantic role labeling systems," in *Proceedings of the 9th Conference on Computational Natural Language Learning*, Ann Arbor, MI, USA, 2005, pp. 181–184.

[14] A. Haghighi, K. Toutanova, and C. D. Manning, "A joint model for semantic role labeling," in *Proceedings of the 9th Conference on Computational Natural Language Learning*, Ann Arbor, MI, USA, 2005, pp. 173–176.

[15] M. Surdeanu and J. Turmo, "Semantic role labeling using complete syntactic analysis," in *Proceedings of the 9th Conference on Computational Natural Language Learning*, Ann Arbor, MI, USA, 2005, pp. 221–224.

[16] L. Màrquez, P. Comas, J. Giménez, and N. Catala, "Semantic role labeling as sequential tagging," in *Proceedings of the 9th Conference on Computational Natural Language Learning*, Ann Arbor, MI, USA, 2005, pp. 193–196.

[17] S. Pradhan, K. Hacioglu, W. Ward, J. H. Martin, and D. Jurafsky, "Semantic role chunking combining complementary syntactic views," in *Proceedings of the 9th Conference on Computational Natural Language Learning*, Ann Arbor, MI, USA, 2005, pp. 217–220.

[18] T.-L. N. My-Linh Ha, V.-H. Nguyen, T.-M.-H. Nguyen, P. Le-Hong, and T.-H. Phan, "Building a semantic role annotated corpus for Vietnamese," in *Proceedings of the 17th National Symposium on Information and Communication Technology*, Daklak, Vietnam, 2014, pp. 409–414.

[19] P. Le-Hong, A. Roussanaly, and T.-M.-H. Nguyen, "A syntactic component for Vietnamese language processing," *Journal of Language Modelling*, vol. 3, no. 1, pp. 145–184, 2015.

[20] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," in *Proceedings of Workshop at ICLR*, Scottsdale, Arizona, USA, 2013.

[21] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in Neural Information Processing Systems 26*, C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Weinberger, Eds. Curran Associates, Inc., 2013, pp. 3111–3119.

[22] O. Täckström, K. Ganchev, and D. Das, "Efficient inference and structured learning for semantic role labeling," *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 29–41, 2015.

[23] V. Punyakanok, D. Roth, W. tau Yih, and D. Zimak, "Semantic role labeling via integer linear programming inference," in *Proceedings of the 20th International Conference on Computational Linguistics*, University of Geneva, Switzerland, 2004, pp. 1346–1352.