НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
сентябрь–октябрь 2024 Том 24 № 5 http://ntv.ifmo.ru/
SCIENTIFIC AND TECHNICAL JOURNAL OF INFORMATION TECHNOLOGIES, MECHANICS AND OPTICS
September–October 2024 Vol. 24 No 5 http://ntv.ifmo.ru/en/
ISSN 2226-1494 (print) ISSN 2500-0373 (online)

doi: 10.17586/2226-1494-2024-24-5-779-787

ViSL model: The model automatically generates sentences of Vietnamese sign language

Khanh Dang[1]✉, Igor A. Bessmertny[2]

[1,2] ITMO University, Saint Petersburg, 197101, Russian Federation

[1] dangkhanhmta.2020@gmail.com✉, https://orcid.org/0009-0009-5882-7653
[2] bessmertny@itmo.ru, https://orcid.org/0000-0001-6711-6399

Abstract

The main problem in building intelligent systems is the lack of data for machine learning, which is especially important for sign language recognition for the deaf and hard of hearing. One of the ways to increase the amount of data for training is synthesis. Unlike speech synthesis, it is impossible to create a sequence of gestures in Vietnamese and some other languages that exactly repeat the text. This is due to the significant limitations of the gesture dictionary and the different word order in sentences. The aim of the work is to enrich the educational corpus of video data for use in creating recognition systems for the Vietnamese Sign Language (ViSL). Since it is impossible to translate the words of the source text into gestures one to one, the problem of translating from a regular language into a sign language arises. The paper proposes to use a two-phase process for this. The first phase involves pre-processing the text with standardization of the text format, segmentation of words and sentences, and then encoding the words using the sign language dictionary. At this stage, it should be noted that there is no need to remove punctuation marks and stop words, since they are related to the accuracy of the *N*-gram model. Next, instead of using syntactic analysis, a statistical method for forming a sequence of gestures is used, and the Markov model on the transition graph between words is taken as a basis in which the probability of the next word depends only on the two previous words. Transition probabilities are calculated on the existing marked corpus of the ViSL. The Breadth-first Search method is used to compile a list of all sentences generated based on a given grammatical rule and a matrix of semantic interactions between words. The inverse of the logarithm of the product of the probabilities of co-occurrence of consecutive 3-word phrases in a sentence is used to estimate the frequency of occurrence of that sentence in a given data set. Based on the ViSL data of 3,234 words, we calculated probability matrices representing the relationships between words based on Vietnamese natural language data with 50 million sentences collected from Vietnamese newspapers and magazines. For different grammar rules, we compare the number of generated sentences and evaluate the accuracy of the 50 most frequent sentences. The average accuracy is 88 %. The accuracy of the generated sentences is estimated by manual statistical methods. The number of generated sentences depends on the number of word parts that are labeled according to the grammar rules. The semantic accuracy of the generated sentences will be very high if the search words are labeled with the correct part-of-speech tagging. Compared with machine learning methods, our proposed method gives very good results for languages without inflections and word order that follow certain rules, such as Vietnamese, and does not require large computational resources. The disadvantage of this method is that its accuracy largely depends on the type of word, sentence, and word segmentation. The relationship of words depends on the observed dataset. Future research direction is to generate paragraphs in sign language. The obtained data can be used in machine learning models for sign language processing tasks.

Keywords

Vietnamese sign language, sign language model, automatic sentence generation, *n*-gram, Markov model, breadth-first search, data enrichment, grammatical rules

For citation: Dang Kh., Bessmertny I.A. ViSL model: The model automatically generates sentences of Vietnamese sign language. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2024, vol. 24, no. 5, pp. 779–787. doi: 10.17586/2226-1494-2024-24-5-779-787

УДК 004.932.72’1, 004.852

ViSL model: модель автоматической генерации предложений вьетнамского языка жестов

Хань Данг[1]✉, Игорь Александрович Бессмертный[2]

[1,2] Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация

[1] dangkhanhmta.2020@gmail.com✉, https://orcid.org/0009-0009-5882-7653

[2] bessmertny@itmo.ru, https://orcid.org/0000-0001-6711-6399

Аннотация

Введение. Основной проблемой при построении интеллектуальных систем является недостаточность данных для машинного обучения, что особенно актуально для распознавания языка жестов для глухих и слабослышащих людей. Одним из способов увеличения объема данных для обучения интеллектуальных систем является их синтез. В отличие от синтеза речи, создавать последовательность жестов на вьетнамском и некоторых других языках, в точности повторяющих текст, невозможно. Это связано с существенной ограниченностью словаря жестов и отличающимся порядком слов в предложениях. Целью работы является обогащение обучающего набора видеоданных для создания систем распознавания вьетнамского языка жестов (Vietnamese Sign Language, ViSL). **Метод.** Поскольку транслировать слова исходного текста в жесты невозможно, возникает задача перевода с обычного языка на жестовый. Для решения поставленной задачи в работе использован двухфазный процесс. На первой фазе выполняется предварительная обработка текста со стандартизацией текстового формата, сегментацией слов и предложений, а затем кодирование слов с помощью словаря языка жестов. На данном этапе не требуется удалять знаки препинания и стоп-слова, поскольку они связаны с точностью *N*-граммовой модели. На второй фазе вместо использования синтаксического анализа применяется статистический метод формирования последовательности жестов. При этом за основу берется марковская модель на графе переходов между словами, в которой вероятность следующего слова зависит только от двух предыдущих слов. Вероятности переходов вычисляются на существующем размеченном наборе ViSL. Метод графового поиска в ширину используется для составления списка всех предложений, сгенерированных на основе заданного грамматического правила и матрицы семантического взаимодействия между словами. Обратное значение логарифма произведения вероятности совместного появления последовательных словосочетаний из трех слов в предложении используется для оценки частоты встречаемости этого предложения в заданном наборе данных. **Основные результаты.** Основываясь на данных ViSL, состоящих из 3234 слов, рассчитаны матрицы вероятности, представляющие отношения между словами, на основе данных ViSL с 50 млн предложений, собранных из вьетнамских газет и журналов. Для различных грамматических правил выполнено сравнение количества сгенерированных предложений и оценка точности 50 наиболее часто встречающихся предложений. Средняя точность составила 88 %. Точность сгенерированных предложений оценена статистическими методами. Показано, что число сгенерированных предложений зависит от количества частей слова, которые помечены в соответствии с правилами грамматики. Семантическая точность сгенерированных предложений высока, если поисковые слова помечены правильными частями речи. **Обсуждение.** По сравнению с методами машинного обучения, предлагаемая модель дает хорошие результаты для языков без словоизменений и порядка слов, следующих определенным правилам, таких как вьетнамский язык, и не требует больших вычислительных ресурсов. Недостатком модели является зависимость точности от типа слова, предложения и сегментации слов. Взаимосвязь слов зависит от наблюдаемого набора данных. Будущее направление исследований — создание абзацев на языке жестов. Полученные данные могут быть использованы в моделях машинного обучения для задач обработки языка жестов.

Ключевые слова

вьетнамский язык жестов, модель языка жестов, автоматически генерация предложений, *n*-грамм, модель Маркова, метод графового поиска в ширину, обогащение данных, грамматические правила

Ссылка для цитирования: Данг Х., Бессмертный И.А. ViSL model: модель автоматической генерации предложений вьетнамского языка жестов // Научно-технический вестник информационных технологий, механики и оптики. 2024. Т. 24, № 5. С. 779–787 (на англ. яз.). doi: 10.17586/2226-1494-2024-24-5-779-787

Introduction

Recent years have been marked by the rapid development of artificial intelligence technologies which have significantly changed the quality of human life, especially for people with disabilities. In particular, research in the field of sign language recognition at the word [1–3] and sentence level [4–6] has yielded very good results opening promising directions for further development aimed at reducing the socialization gap of people with hearing and speech impairments. Sign language in each country has its own unique characteristics, but the common point is that gestures and facial expressions are the main elements of sign language. From a semantic point of view, sign language is closely related to natural language and carries national and cultural characteristics. To express the meaning of a sentence in sign language, it is necessary to perform a grammatical conversion process from natural language to sign language, and then match the words with the corresponding gestures. In the sign languages of some countries, such as Vietnam [7] and Russian [8], word order changes compared to natural languages. One of the problems of generating gestures is that the vocabulary of sign language is significantly smaller compared to the vocabulary of natural languages. In particular, Vietnamese Sign Language (ViSL) has only 3,234 words.

The task of recognizing sign language using machine learning requires large amounts of labeled data. A proven method for enriching data corpora is synthesis. The purpose of this study is to enrich labeled data corpora for ViSL recognition by generating sign language interpretation videos.

Problems with language models

In the field of natural language processing, language models are widely and effectively used in tasks such as: language recognition and machine translation models [9], spelling error detection and sentence editing [10], etc. Building a language model is necessary to create applications that require understanding language. Sign languages are closely related to natural languages, but using natural language models to understand sign languages is not possible. The reason is that word order and grammatical structure are different; many words in natural languages were not represented in sign language [7, 8].

Published research on sign language processing mainly focuses on problems and methods of recognizing sign language at the letter and word level [1–3], at the continuous level [4–6]. To advance research at higher and more complex levels, such as problems in machine translation from sign language, it is necessary to create a language model specifically for sign language. An effective model of sign language will help the computer take into account the semantics and representational context of sentences.

The construction of a natural language model can be carried out in accordance with three main approaches.

— Construction of a language model using a knowledge base created by language experts [11]. The language model is built on the basis of a set of knowledge base rules: Grammatical — Ungrammatical, Intra-grammatical — Extra-grammatical, Non-grammatical — Out-of-grammatical, Qualitative language model — Quantitative language model. The advantage of this method is that it does not require training data. The disadvantage is that it is difficult to develop and requires time and the involvement of language experts. This model produces highly accurate results for written language (formal), but the results may not be reliable for spoken language (informal). In addition, it is unable to predict the appearance of a word and is unable to generate text.

— Building language models using statistical methods [12]. This is a method for calculating the probability distribution for a string of words of length k words: $w_1, \ldots, w_K$ denoted by $P(w_1, w_2, \ldots, w_K)$, where $\{w_1, w_2, \ldots, w_K\} \in W$ is a set of data belonging to a particular language. Then the probability of occurrence of the sequence $w_1, w_2, \ldots, w_K$ will be calculated using the following formula:

$$P(w_1, w_2, \ldots, w_K) = \\ = P(w_1)P(w_2|w_1) \ldots P(w_K|w_1, w_2, \ldots, w_{K-1}) = \\ = \prod_{k=1}^{K} P(w_K|w_1, w_2, \ldots, w_{K-1}),$$

where $P(w_K|w_1, w_2, \ldots, w_{K-1})$ this is the probability of the word w_K given the known probability of occurrence of the sequence $w_1, w_2, \ldots, w_{K-1}$. Given that the frequency of occurrence of the strings $w_1, w_2, \ldots, w_{K-1}$ and $w_1, w_2, \ldots, w_K$ is equal to f_{K-1} and f_K, respectively, we can calculate the probability of occurrence of the word w_K in a set of texts when we know that the probability of occurrence of the string $w_1, w_2, \ldots, w_{K-1}$ is equal to: $P(w_K|w_1, w_2, \ldots, w_{K-1}) = \frac{f_K}{f_{K-1}}$. Calculating the probability of occurrence of the word w_K, taking into account that it depends only on the occurrence of $N-1$ words before it in accordance with Markov's law, we obtain the formula for the N-gram Markov model [13]:

$$P(w_K|w_1, w_2, \ldots, w_{K-1}) = P(w_K|w_{K-N+1}, w_{K-N+2}, \ldots, w_{K-1}).$$

The disadvantage of this method is that it requires large computational and storage resources for large data sets. In addition, if a pair of words rarely occurs together in this data set, the probability will be close to 0. To overcome this disadvantage, data smoothing methods are used in N-gram calculations, such as Discounting, Back-off, and Interpolation. In addition to improving the accuracy of semantic structure, some studies have combined N-grams with a structured Language Model (Structured LM).

— Building language models using a neural network. For this method, the words in a sentence will be encoded into vectors, and the sentences will be a series of encoded vectors of words. According to this architecture, the input data will be fed to neural networks for processing time series data, such as Recurrent Neural Networks (RNN) [13], Long Short-Term Memory (LSTM) [14], Transform [15]. The main advantage of this method is that it allows you to perceive the context of words, producing accurate results without paying attention to the grammar of the language. The disadvantage is the need for large amounts of data and large computing resources.

In addition to the above approaches, the Large Language Model (LLM) [16] is currently attracting the most attention due to its accuracy and ability to understand language. However, implementing the LLM model is very difficult because it is very expensive in terms of computational resources as well as a huge amount of data. For sign languages, the vocabulary size is not large enough for us to build a model to obtain sign language data using statistical methods.

Description of the proposed model

Automatic generation of sentences in sign language

To build a model of machine translation into sign language, the task of recognizing gestures in videos is not enough. A sign language machine translation model is effective if it is implemented in conjunction with a language model for a specific sign language, since the vocabulary of sign languages is often much smaller. Thus, building a language model for sign language using statistical methods would be appropriate. In this study, the main tasks for

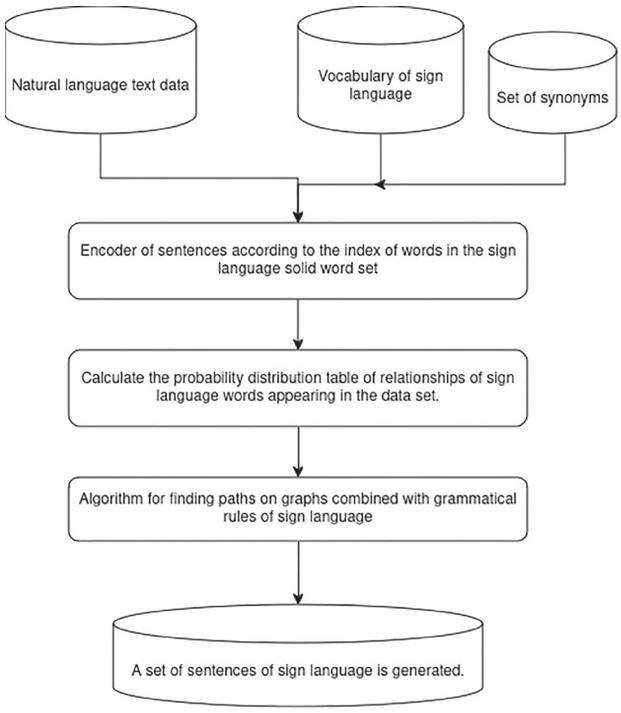


Fig. 1. Diagram summarizing the process of generating sentences in sign language

constructing a language model of ViSL and generating Sign Language sentences are summarized as shown in Fig. 1.

After performing pre-processing steps on the input text dataset, such as removing special characters, html tags, word segmentation, punctuation standardization, sentence segmentation, the sentences in the dataset will be encoded and indexed by words in the ViSL dictionary. A dataset of synonyms will be added to reduce variance given that the number of words in sign language is much less than in natural language. Words that are not in the sign language dictionary will be coded as 0.

We will calculate the probability distribution table of interactions between words in a sign language dictionary according to the Markov property of the *N*-gram model. Next, algorithms that map paths in a graph in combination with the grammatical rules of sign language will be used to generate sentences in sign language.

Constructing a probability matrix representing the interactions between words in the ViSL dictionary

ViSL sentences are a set of sign language words of size L words, with any two words from the dictionary set $\{W_i, W_j\} \in VS$, $0 \le i, j \le N - 1$. Then $P(W_{ij}, d)$ is the probability that the word W_j will appear after the word W_i at a distance of d words, will be calculated by the formula:

$$\{P(W_{ij}, d) = \frac{f_{(W_i|W_j,d)}}{f_{W_j}} index(j) - index(i) = d,\ d > 0,$$

where $f_{(W_j|W_i)}$ frequency of occurrence of the word W_i and the word W_j in one sentence and in the data set. Condition: $index(j) - index(i) = d$, $d > 0$ guarantees that the word W_j follows the word W_i at a distance of d words, f_{wj} is the frequency of occurrence of the word W_j in the data set.

The above formula corresponds to the properties of the Markov model, that is, we assume that the occurrence of the word W_j depends only on the previous n words. In this study, we consider the occurrence of the word W_j depending only on the word W_i. The d value reflects the interaction between two words. The smaller d, the more the word W_j will depend on the word W_i. We calculate three probability distribution tables corresponding to the values $d = 1, 2, 3$.

To optimize time, it is necessary to calculate a table of probability distributions for matching words in a data set. We propose a method for encoding a dataset and then reviewing each sentence. A data set of size N sentences will have $O(N)$ complexity.

BFS search algorithm for generating sentences in ViSL

Unlike natural language, the word order of sentences in ViSL varies compared to natural language [17]. In natural languages, the sentence structure is usually *S* (*subject*) — *V* (*predicate*) — *O* (*object*), while in sign languages the sentence order will be: *S* (*subject*) — *O* (*object object*) — *V* (*predicate*). This means that the object must first be identified before the appropriate gestures can be used to express the action. Here are some basic grammatical rules for constructing sentences in ViSL — Subject — Object — Predicate — Words of time; Adverb — Subject — Predicate and Object — Adverb.

The index of each word in a set of sign language words represents a vertex in the graph. The probability distribution table of the word occurrence ratio in a data set is considered as an adjacency matrix with the probability values as the weights of the directed graph. An example of the BFS algorithm on a graph used in combination with grammar rules and a Markov model to generate sign language sentences is shown in Fig. 2.

When calculating three matrices representing the relations of interaction of words in a dataset, if the distance between them is calculated for each d equal to 1, 2, 3, we get the corresponding matrices: **uniMatrix**, **biMatrix**, **triMatrix**. Given the **Rules** grammatical rule, a sign language sentence $S = \{w_0, w_1, \ldots, w_n\}$ is generated, which must meet the following conditions:

$$\{\mathbf{uniMatrix}[i, i + 1]! = 0, \forall i{:}0 \le i < \text{len}(\mathbf{Rules}) -$$

$$- 1\ \mathbf{biMatrix}[i, i + 2]! = 0, \forall i{:}0 \le i < \text{len}(\mathbf{Rules}) -$$

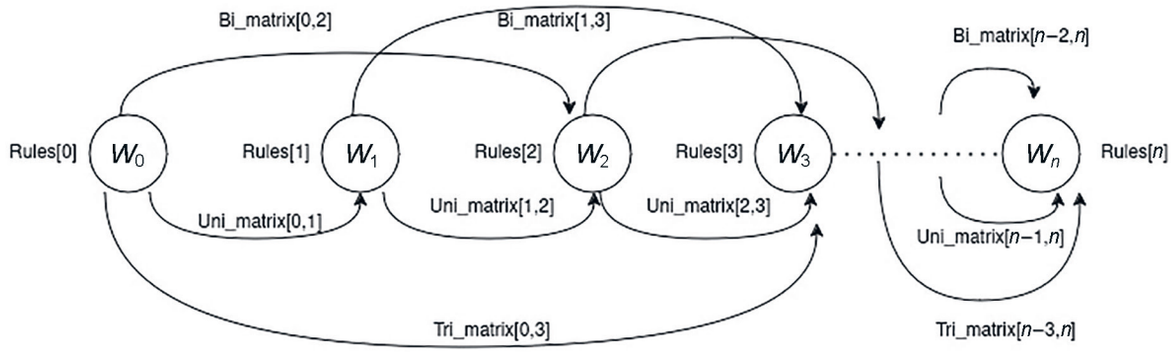
$$- 2\ \mathbf{triMatrix}[i, i + 3]! = 0, \forall i{:}0 \le i < \text{len}(\mathbf{Rules}) -$$

$$- 3\ \mathbf{mapIndex}[i]! = \mathbf{Rules}\ [i], \forall i{:}0 \le i < \text{len}(\mathbf{Rules}),$$

where, **mapIndex** is the coding map index of words corresponding to the marking of parts of speech. Then the probability of generating a sign language sentence of length n words is calculated by the formula of simultaneous probability of *N*-gram clusters as:

$$\text{Probability of sentence occurrence} - PS = \prod_{i=0}^{n-1} M[i, i + 1].$$

$$\prod_{i=0}^{n-2} M[i, i + 2]. \prod_{i=0}^{n-3} M[i, i + 3].$$



$W_0, W_1, \ldots W_n$ are words in the sign language dictionary, used to generate sentences.
Rules is an array representing a Vietnamese grammar rule.
Uni_matrix, Bi_matrix, Tri_matrix are matrices representing the interactions between words of the sign language dictionary, calculated based on the distance value from d equal to 1, 2, 3 respectively.

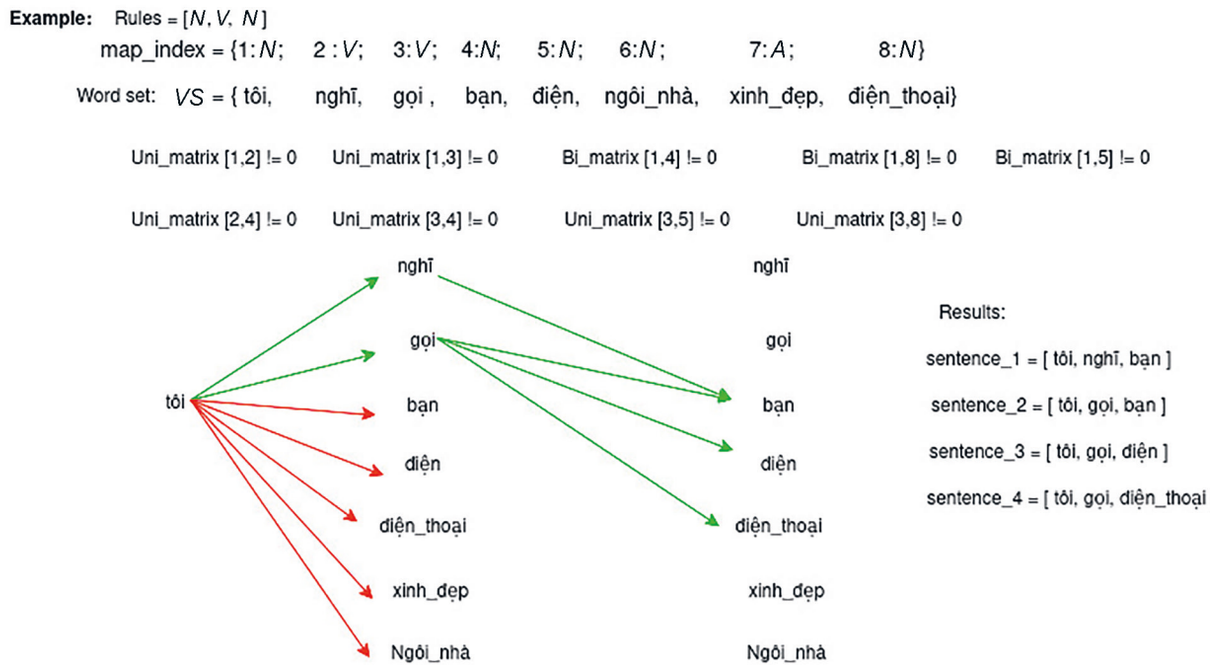


Fig. 2. Representation of a sentence in sign language generated in accordance with a grammatical rule and calculated with probability according to the Markov rule

For very large data sets, the frequency of words can be much higher than the frequency of phrases. To avoid losses in the calculation, we transform the logarithms, the results are as follows:

$$\text{Cost} = (-1) \times \log(PS) = (-1) \times \left(\sum_{i=0}^{n-1} \log(M[i, i+1]) + \sum_{i=0}^{n-2} \log(M[i, i+2]) + \sum_{i=0}^{n-3} \log(M[i, i+3] \right).$$

$M[i, i+1]$ refers to the probability of the word $S[i]$ appearing before the word $S[i+1]$ in the dataset. When 2 consecutive words correspond to the value $d = 1$, we use **uniMatrix** to calculate. When $d = 2$, we consider $S[i]$ and $S[i+2]$, then **biMatrix** is used for calculation. Similarly with $M[i, i+3]$, **triMatrix** is used for calculation.

Our goal is to find a set of words that correspond to the given grammatical rules and at the same time satisfy the probability condition. According to the Cost formula, the lower the Cost value, the higher the probability that the generated application is more likely to be encountered in the dataset.

We call the value Cost, because it has the same value as the cost in the task of finding the shortest path on the graph.

Graph path finding algorithms [17], such as Breadth-First Search (BFS) and Depth-First Search (DFS), are used to list all collocations according to a given grammatical rule. The BFS function for generating sign language sentences that satisfy the grammar rules and Markov states is configured as follows:

```
def BFS(map_index, uni_matrix, bi_matrix, tri_matrix, visited, path, node,
rules):
 paths = []
 visited[node] = True
 new_path = path + [node]
 if len(new_path) == len(rules):
 Cost = calculate_Cost(new_path, uni_matrix, bi_matrix, tri_matrix)
 paths.append((new_path, Cost))
 else:
 for x in range(len(uni_matrix[node])):
 if len(new_path) == 1:
 if map_index[x] == list_rules[len(new_path)] and visited[x] == False:
 if uni_matrix[node][x] != 0:
 paths.extend(BFS(map_index, uni_matrix, bi_matrix, tri_matrix, visited, new_
path, x, rules))
 if len(new_path) == 2:
 if map_index[x] == list_rules[len(new_path)] and visited[x] == False:
 if uni_matrix[node][x] != 0 and bi_matrix[new_path[-2]][x] != 0:
 paths.extend(BFS(map_index, uni_matrix, bi_matrix, tri_matrix, visited, new_
path, x, rules))
 if len(new_path) > 2:
 if map_index[x] == list_rules[len(new_path)] and visited[x] == False:
 if uni_matrix[node][x] != 0 and bi_matrix[new_path[-2]][x] != 0 and tri_
matrix[new_path[-3]][x] != 0:
 paths.extend(BFS(map_index, uni_matrix, bi_matrix, tri_matrix, visited, new_
path, x, rules))
 visited[node] = False
 paths.sort(key=lambda x: x[1], reverse=True)
 return paths

def BFS(map_index, uni_matrix, bi_matrix, tri_matrix, visited, path, node,
rules):
 paths = []
 visited[node] = True
 new_path = path + [node]
 if len(new_path) == len(rules):
 Cost = calculate_Cost(new_path, uni_matrix, bi_matrix, tri_matrix)
 paths.append((new_path, Cost))
 else:
 for x in range(len(uni_matrix[node])):
 if len(new_path) == 1:
 if map_index[x] == list_rules[len(new_path)] and visited[x] == False:
 if uni_matrix[node][x] != 0:
 paths.extend(BFS(map_index, uni_matrix, bi_matrix, tri_matrix, visited, new_
path, x, rules))
 if len(new_path) == 2:
 if map_index[x] == list_rules[len(new_path)] and visited[x] == False:
 if uni_matrix[node][x] != 0 and bi_matrix[new_path[-2]][x] != 0:
 paths.extend(BFS(map_index, uni_matrix, bi_matrix, tri_matrix, visited, new_
path, x, rules))
 if len(new_path) > 2:
 if map_index[x] == list_rules[len(new_path)] and visited[x] == False:
 if uni_matrix[node][x] != 0 and bi_matrix[new_path[-2]][x] != 0 and tri_
matrix[new_path[-3]][x] != 0:
 paths.extend(BFS(map_index, uni_matrix, bi_matrix, tri_matrix, visited, new_
path, x, rules))
 visited[node] = False
 paths.sort(key=lambda x: x[1], reverse=True)
 return paths
```

Experiments and results

The Vietnamese text dataset is collected from articles with a total number of sentences after preprocessing: 50 million sentences[1].

After removing duplicate words, the ViSL dictionary has a size of 3,234 words[2].

We calculate the probability distribution table of interactions between words in sign language according to the distance value d = 1, 2, 3 words. The probability distribution tables of interactions between words are saved as a numpy file, the word encoding table and its index are saved as a text file.

When specifying the word "tôi – I" as the initial vertex with the grammatical rule: "*N* (noun) – *V* (verb) – *N* (noun)". We have generated 44,290 offers. Of these, the 10 most common sentences are shown in Table 1.

To compare the accuracy of three models with different d values, we generated 5 data samples for each model. Each data sample contains the 20 sentences with the highest probability of occurrence generated from the seed word and the grammar rule. When calculating the semantic accuracy of data samples, we get the following comparison Table 2.

The source code of the project can be viewed at the link[3].

Conclusion and discussion

In this study, we built a model to generate ViSL sentences. The advantage of this model is that it is very accurate and does not require large computational resources. We propose a method to construct a matrix representing the semantic interactions between words in the ViSL dictionary, and then apply grammar rules and breadth search algorithms to generate ViSL. The disadvantage of this method is that its accuracy depends on the accuracy of the data processing steps, such as part-word labeling, sentence segmentation, and Vietnamese word segmentation. Our model can be used to generate sign language sentences by estimating the probability of the next word in a sentence, but this probability is calculated based on the collected dataset. The next direction of research is to build a model to generate paragraphs in sign language. The findings can be applied to machine learning and neural models to solve more complex sign language processing problems.

Table 1. Example of generating sentences of sign language

Sign language sentences	Translate	Cost
tôi –> nghĩ –> bạn	I think you	16.450
tôi –> gọi –> điện	I call	16.644
tôi –> nghĩ –> chị	I think you	16.974
tôi –> muốn –> bạn	I want you	17.527
tôi –> gọi –> điện_thoại	I call	18.172
tôi –> nghe –> báo_cáo	I heard the report	18.336
tôi –> nhận –> trách_nhiệm	I take responsibility	18.383
tôi –> nghe –> chị	I hear you	18.488
tôi –> hỏi –> bạn	I ask you	18.523

Table 2. Comparison of the performance of language models with different grammar rules

Rules	Total number of sentences generated	Accuracy, %
N – V – N	4,636,156	94
P – V – N – E – Np	3,074,695	80
N – V – M – N	42,608,705	90
N – V – A	1,251,234	96
P – V – A	46,122	84
P – V – N	164,927	82
P – V – Nc – N	46,323	96
E – N –V – Nc – A	1,171,297	70
Np – V – Nc – N	51,276	92
P – V – M – N	1,474,344	96
Average value of the sum		88

Footnote: N — Noun, *V* — Verb, *A* — Adjective, *P* — Pronoun, *Np* — Proper Noun, *Nc* — Classification, *E* — Sentence, *M* — Numeral.

Conclusion and discussion

In this study, we built a model to generate Vietnamese Sign Language (ViSL) sentences. The advantage of this model is that it is very accurate and does not require large computational resources. We propose a method to construct a matrix representing the semantic interactions between words in the ViSL dictionary, and then apply grammar rules and breadth search algorithms to generate ViSLe. The disadvantage of this method is that its accuracy depends on the accuracy of the data processing steps, such as part-word labeling, sentence segmentation, and Vietnamese word segmentation. Our model can be used to generate sign language sentences by estimating the probability of the next word in a sentence, but this probability is calculated based on the collected dataset. The next direction of research is to build a model to generate paragraphs in sign language. The findings can be applied to machine learning and neural models to solve more complex sign language processing problems.

[1] Available at: https://drive.google.com/file/d/1GFbe-qs6HmCYs0JwJgivOy2Bvb06M8OI/view (date accessed: 14.04.2024).

[2] Available at: https://github.com/DangKhanhITMO/VnSignLanguage (date accessed: 14.04.2024).

[3] Available at: https://colab.research.google.com/drive/1-8_vp24tKNchhb4s3Q1WknxU46XsOg1O?usp=sharing (date accessed: 14.06.2024).

References

1. Katti R.K., Sujatha C., Desai P., Shankar G. Character and word level gesture recognition of indian sign language. *Proc. of the 2023 IEEE 8th International Conference for Convergence in Technology (I2CT)*, 2023, pp. 1–6. https://doi.org/10.1109/I2CT57861.2023.10126314
2. Naz N., Sajid H., Ali S., Hasan O., Ehsan M.K. Signgraph: An efficient and accurate pose-based graph convolution approach toward sign language recognition. *IEEE Access*, 2023, vol. 11, pp. 19135–19147. https://doi.org/10.1109/ACCESS.2023.3247761
3. Boháček M., Hrúz M. Sign pose-based transformer for word-level sign language recognition. *Proc. of the 2022 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW)*, 2022, pp. 182–191. https://doi.org/10.1109/WACVW54805.2022.00024
4. Jiang Y., Li F., Li Z., Liu Z., Wang Z. Enhancing continuous sign language recognition with Self-Attention and MediaPipe Holistic. *Proc. of the 2023 8th International Conference on Instrumentation, Control, and Automation (ICA)*, 2023, pp. 97–102. https://doi.org/10.1109/ICA58538.2023.10273118
5. Nayan N., Ghosh D., Pradhan P.M. An unsupervised learning approach to handle movement epenthesis in continuous sign language recognition. *Proc. of the 2022 17th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, 2022, pp. 862–867. https://doi.org/10.1109/ICARCV57592.2022
6. Tran K.B., Nguyen U.D., Huynh Q.T. Continuous sign language recognition using MediaPipe. *Proc. of the 2023 International Conference on Advanced Technologies for Communications (ATC)*, 2023, pp. 493–498. https://doi.org/10.1109/ATC58710.2023.10318855
7. Quach L.-D., Nguyen C.-N. Conversion of the Vietnammese grammar into sign language structure using the example-based machine translation algorithm. *Proc. of the 2018 International Conference on Advanced Technologies for Communications (ATC)*, 2018, pp. 27–31. https://doi.org/10.1109/ATC.2018.8587584
8. Kagirov I., Ryumin D., Ivanko D., Axyonov A., Karpov A. Russian sign language: History, grammar and sociolinguistic situation in brief. *Proc. of the Language Technologies for All (LT4All)*, 2019, pp. 71–74.
9. Singh C., Bansal R.K., Bansal S. Machine translation techniques using AI: A review. *Proc. of the 2023 IEEE International Conference on Computer Vision and Machine Intelligence (CVMI)*, 2023, pp. 1–5. https://doi.org/10.1109/CVMI59935.2023.10464455
10. Tan M., Chen D., Li Z., Wang P. Spelling error correction with BERT based on character-phonetic. *Proc. of the 2020 IEEE 6th International Conference on Computer and Communications (ICCC)*, 2020, pp. 1146–1150. https://doi.org/10.1109/ICCC51575.2020.9345276
11. Huang C., Feng Y., Zhang Y., Zhang W. Knowledge Base System of Electrical equipment management and potential risk control based on natural language processing technology. *Proc. of the 2023 Asia-Europe Conference on Electronics, Data Processing and Informatics (ACEDPI)*, 2023, pp. 439–445. https://doi.org/10.1109/ACEDPI58926.2023.00090
12. Liu S., Tang R., Chai J. A news automatic tagging method based on statistical language model. *Proc. of the 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)*, 2017, pp. 1–5. https://doi.org/10.1109/CISP-BMEI.2017.8302092
13. Xiao J., Zhou Z. Research Progress of RNN Language Model. *Proc. of the 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, 2020, pp. 1285–1288. https://doi.org/10.1109/ICAICA50127.2020.9182390
14. Ganai F., Khursheed F. Predicting next Word using RNN and LSTM cells: Stastical Language Modeling. *Proc. of the 2019 Fifth International Conference on Image Information Processing (ICIIP)*, 2019, pp. 469–474. https://doi.org/10.1109/ICIIP47207.2019.8985885
15. Acheampong F.A., Nunoo-Mensah H., Chen W. Recognizing emotions from texts using an ensemble of transformer-based language models. *Proc. of the 2021 18th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, 2021, pp. 161–164. https://doi.org/10.1109/ICCWAMTIP53232.2021.9674102
16. Lee H., Kim J.-H., Hwang E.J., Kim J., Park J.C. Leveraging large language models with vocabulary sharing for sign language translation. *Proc. of the 2023 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW)*, 2023, pp. 1–5. https://doi.org/10.1109/ICASSPW59220.2023.10193533

Литература

1. Katti R.K., Sujatha C., Desai P., Shankar G. Character and word level gesture recognition of indian sign language // Proc. of the 2023 IEEE 8th International Conference for Convergence in Technology (I2CT). 2023. P. 1–6. https://doi.org/10.1109/I2CT57861.2023.10126314
2. Naz N., Sajid H., Ali S., Hasan O., Ehsan M.K. Signgraph: An efficient and accurate pose-based graph convolution approach toward sign language recognition // IEEE Access. 2023. V. 11. P. 19135–19147. https://doi.org/10.1109/ACCESS.2023.3247761
3. Boháček M., Hrúz M. Sign pose-based transformer for word-level sign language recognition // Proc. of the 2022 IEEE/CVF Winter Conference on Applications of Computer Vision Workshops (WACVW). 2022. P. 182–191. https://doi.org/10.1109/WACVW54805.2022.00024
4. Jiang Y., Li F., Li Z., Liu Z., Wang Z. Enhancing continuous sign language recognition with Self-Attention and MediaPipe Holistic // Proc. of the 2023 8th International Conference on Instrumentation, Control, and Automation (ICA). 2023. P. 97–102. https://doi.org/10.1109/ICA58538.2023.10273118
5. Nayan N., Ghosh D., Pradhan P.M. An unsupervised learning approach to handle movement epenthesis in continuous sign language recognition // Proc. of the 2022 17th International Conference on Control, Automation, Robotics and Vision (ICARCV). 2022. P. 862–867. https://doi.org/10.1109/ICARCV57592.2022.10004317
6. Tran K.B., Nguyen U.D., Huynh Q.T. Continuous sign language recognition using MediaPipe // Proc. of the 2023 International Conference on Advanced Technologies for Communications (ATC). 2023. P. 493–498. https://doi.org/10.1109/ATC58710.2023.10318855
7. Quach L.-D., Nguyen C.-N. Conversion of the Vietnammese grammar into sign language structure using the example-based machine translation algorithm // Proc. of the 2018 International Conference on Advanced Technologies for Communications (ATC). 2018. P. 27–31. https://doi.org/10.1109/ATC.2018.8587584
8. Kagirov I., Ryumin D., Ivanko D., Axyonov A., Karpov A. Russian sign language: History, grammar and sociolinguistic situation in brief // Proc. of the Language Technologies for All (LT4All). 2019. P. 71–74.
9. Singh C., Bansal R.K., Bansal S. Machine translation techniques using AI: A review // Proc. of the 2023 IEEE International Conference on Computer Vision and Machine Intelligence (CVMI). 2023. P. 1–5. https://doi.org/10.1109/CVMI59935.2023.10464455
10. Tan M., Chen D., Li Z., Wang P. Spelling error correction with BERT based on character-phonetic // Proc. of the 2020 IEEE 6th International Conference on Computer and Communications (ICCC). 2020. P. 1146–1150. https://doi.org/10.1109/ICCC51575.2020.9345276
11. Huang C., Feng Y., Zhang Y., Zhang W. Knowledge Base System of Electrical equipment management and potential risk control based on natural language processing technology // Proc. of the 2023 Asia-Europe Conference on Electronics, Data Processing and Informatics (ACEDPI). 2023. P. 439–445. https://doi.org/10.1109/ACEDPI58926.2023.00090
12. Liu S., Tang R., Chai J. A news automatic tagging method based on statistical language model // Proc. of the 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI). 2017. P. 1–5. https://doi.org/10.1109/CISP-BMEI.2017.8302092
13. Xiao J., Zhou Z. Research Progress of RNN Language Model // Proc. of the 2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA). 2020. P. 1285–1288. https://doi.org/10.1109/ICAICA50127.2020.9182390
14. Ganai F., Khursheed F. Predicting next Word using RNN and LSTM cells: Stastical Language Modeling // Proc. of the 2019 Fifth International Conference on Image Information Processing (ICIIP). 2019. P. 469–474. https://doi.org/10.1109/ICIIP47207.2019.8985885
15. Acheampong F.A., Nunoo-Mensah H., Chen W. Recognizing emotions from texts using an ensemble of transformer-based language models // Proc. of the 2021 18th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP). 2021. P. 161–164. https://doi.org/10.1109/ICCWAMTIP53232.2021.9674102
16. Lee H., Kim J.-H., Hwang E.J., Kim J., Park J.C. Leveraging large language models with vocabulary sharing for sign language translation // Proc. of the 2023 IEEE International Conference on Acoustics, Speech, and Signal Processing Workshops (ICASSPW). 2023. P. 1–5. https://doi.org/10.1109/ICASSPW59220.2023.10193533

17. Garg H., Gupta I., Kumar K., Kaur B., Pundir D. Artificial intelligence based dynamic approach to visualize the graphs. *Proc. of the 2023 International Conference on Computational Intelligence, Communication Technology and Networking (CICTN)*, 2023, pp. 663–667. https://doi.org/10.1109/CICTN57981.2023.10140873

17. Garg H., Gupta I., Kumar K., Kaur B., Pundir D. Artificial intelligence based dynamic approach to visualize the graphs // Proc. of the 2023 International Conference on Computational Intelligence, Communication Technology and Networking (CICTN). 2023. P. 663–667. https://doi.org/10.1109/CICTN57981.2023.10140873

Authors

Khanh Dang — PhD Student, ITMO University, Saint Petersburg, 197101, Russian Federation, sc 59166515900, https://orcid.org/0009-0009-5882-7653, dangkhanhmta.2020@gmail.com
Igor A. Bessmertny — D.Sc., Full Professor, ITMO University, Saint Petersburg, 197101, Russian Federation, sc 36661767800, https://orcid.org/0000-0001-6711-6399, bessmertny@itmo.ru

Авторы

Данг Хань — аспирант, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, sc 59166515900, https://orcid.org/0009-0009-5882-7653, dangkhanhmta.2020@gmail.com
Бессмертный Игорь Александрович — доктор технических наук, профессор, профессор, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, sc 36661767800, https://orcid.org/0000-0001-6711-6399, bessmertny@itmo.ru

Received 16.04.2024
Approved after reviewing 17.07.2024
Accepted 16.09.2024

Статья поступила в редакцию 16.04.2024
Одобрена после рецензирования 17.07.2024
Принята к печати 16.09.2024