Natural Language Processing

Introduction:

The VKU Chatbot is a Python-based chatbot that utilizes machine learning algorithms and natural language processing (NLP) techniques to provide automated assistance to users with college-related inquiries. The chatbot aims to improve the user experience by delivering quick and accurate responses to their questions.

Methodology:

The VKU chatbot is developed using a combination of natural language processing techniques and machine learning algorithms. The methodology involves data preparation, model training, and chatbot response generation. The data is preprocessed to remove noise and increase training examples using synonym replacement. Multiple classification models are trained and evaluated to find the best-performing one. The trained model is then used to predict the intent of user input, and a random response is selected from the corresponding intent's responses. The chatbot is developed as a web application using Flask, allowing users to interact with it in real-time but yet to be deployed.

Overview of using ML in Chatbot

A machine learning chatbot is an AI-driven computer program designed to engage in natural language conversations with users. These chatbots utilize machine learning techniques to comprehend and react to user inputs, whether they are conveyed as text, voice, or other forms of natural language communication.

Key characteristics of machine learning chatbots encompass their proficiency in Natural Language Processing (NLP), enabling them to grasp and interpret human language. They possess the ability to learn from user interactions, continually adjusting their responses for enhanced effectiveness. These chatbots excel at managing multi-turn conversations, making them adaptable to diverse applications. They heavily rely on data for both training and refinement, and they can be seamlessly deployed on websites or various platforms. Furthermore, they are built with an emphasis on ongoing improvement, ensuring their relevance and efficiency in evolving user contexts.

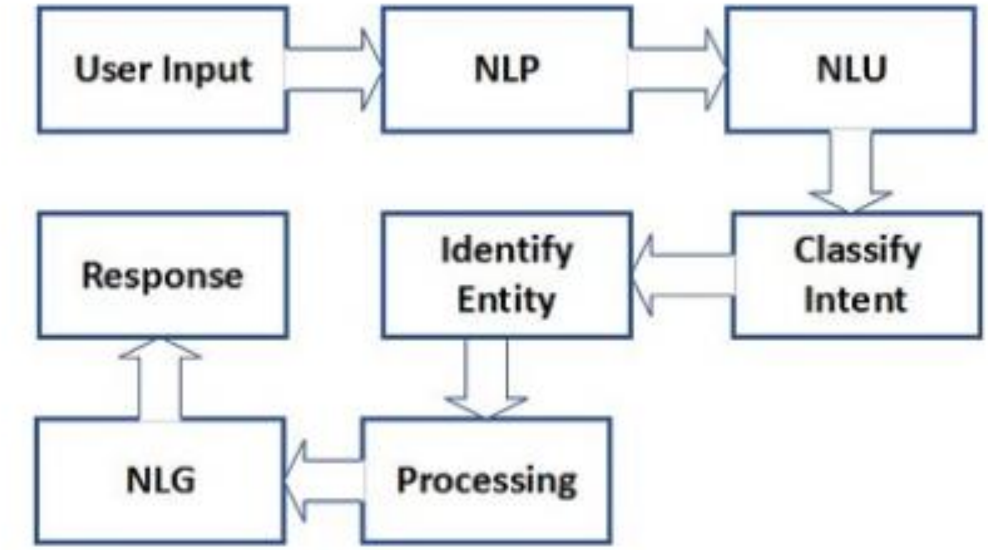


Figure. Process Flow and Basic Components of ML Chatbot

- Natural language processing (NLP) converts human input text into structured data so that the machine can understand it. So, NLP performs the following tasks such as speech recognition, tokenization, parsing information extraction, etc.
- Natural language understanding (NLU) uses an algorithm to classify the intent (verb) and recognize the entity (noun or action content).
- Natural language generation (NLG) converts structured data of the machine into text so that humans can understand it.

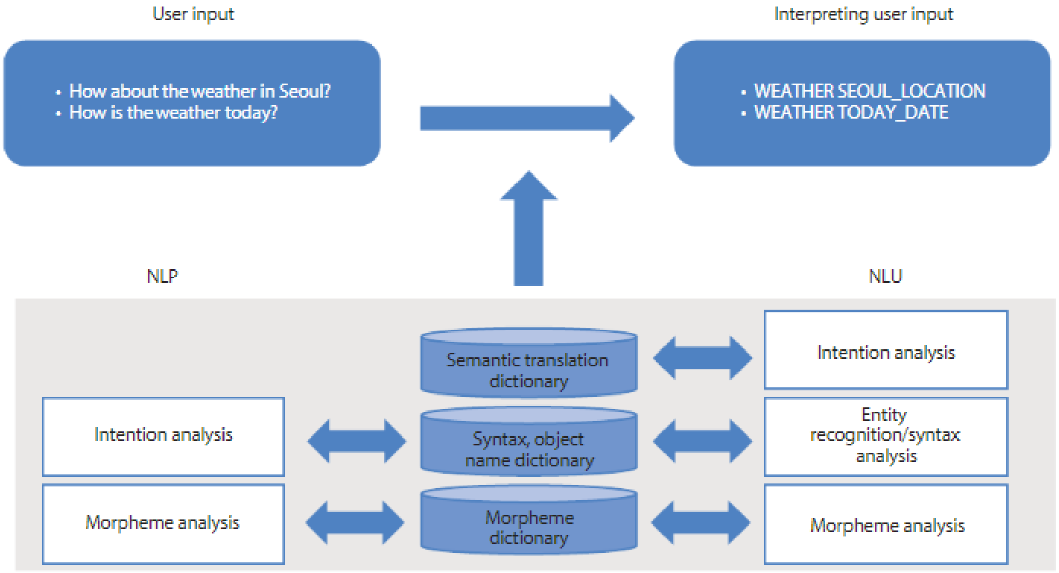


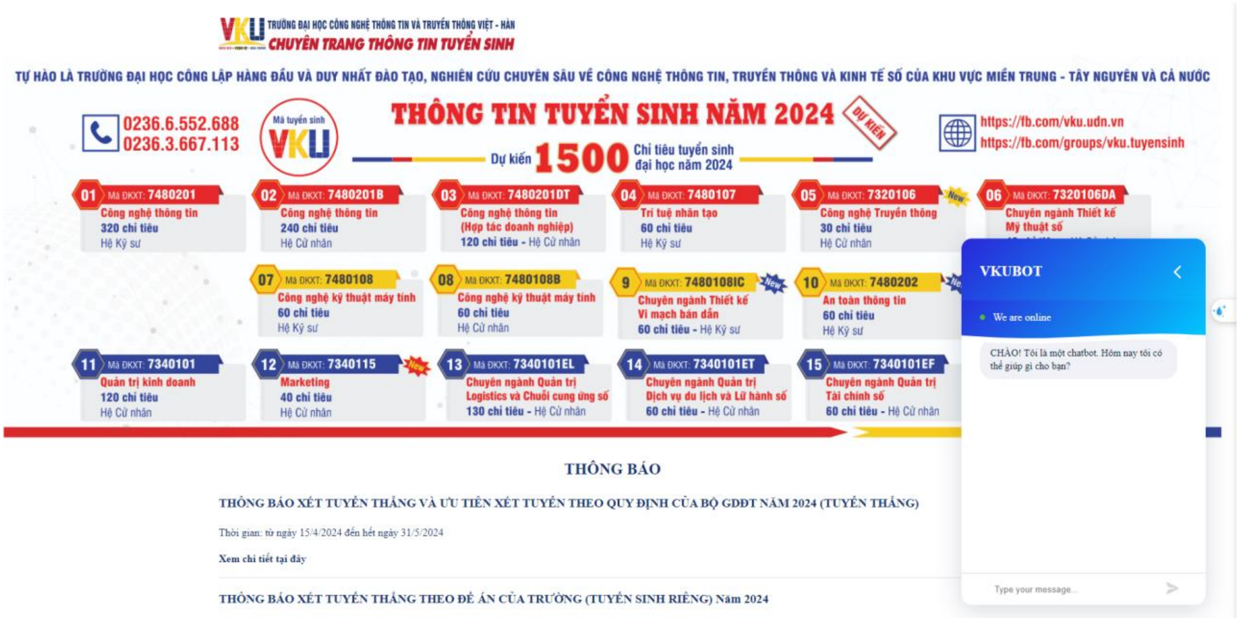
Figure. Machine learning-based chatbot

Motivation:

The motivation behind this project was to create a simple chatbot using my newly acquired knowledge of Natural Language Processing (NLP) and Python programming. As one of our first projects in this field, I wanted to put my skills to the test and see what I could create.

I followed a guide referenced in the documents and GitHub to learn the steps involved in creating an end-to-end chatbot. This included collecting data, choosing programming languages and NLP tools, training the chatbot, and testing and refining it before making it available to users.

Although this chatbot may not have exceptional cognitive skills or be state-of-the-art, it was a great way for me to apply my skills and learn more about NLP and chatbot development. I hope this project will bring something positive to VKU.



TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG VIỆT - HÀN
CHUYÊN TRANG THÔNG TIN TUYỂN SINH

TỰ HÀO LÀ TRƯỜNG ĐẠI HỌC CÔNG LẬP HÀNG ĐẦU VÀ DUY NHẤT ĐÀO TẠO, NGHIÊN CỨU CHUYÊN SÂU VỀ CÔNG NGHỆ THÔNG TIN, TRUYỀN THÔNG VÀ KINH TẾ SỐ CỦA KHU VỰC MIỀN TRUNG - TÂY NGUYÊN VÀ CẢ NƯỚC

☎ 0236.6.552.688
0236.3.667.113

Mã tuyển sinh VKU

THÔNG TIN TUYỂN SINH NĂM 2024 DỰ KIẾN

Dự kiến 1500 Chỉ tiêu tuyển sinh đại học năm 2024

https://fb.com/vku.udn.vn
https://fb.com/groups/vku.tuyensinh

01 Mã ĐKXT: 7480201
Công nghệ thông tin
320 chỉ tiêu
Hệ Kỹ sư

02 Mã ĐKXT: 7480201B
Công nghệ thông tin
240 chỉ tiêu
Hệ Cử nhân

03 Mã ĐKXT: 7480201DT
Công nghệ thông tin
(Hợp tác doanh nghiệp)
120 chỉ tiêu - Hệ Cử nhân

04 Mã ĐKXT: 7480107
Trí tuệ nhân tạo
60 chỉ tiêu
Hệ Kỹ sư

05 Mã ĐKXT: 7320106 New
Công nghệ Truyền thông
30 chỉ tiêu
Hệ Cử nhân

06 Mã ĐKXT: 7320106DA
Chuyên ngành Thiết kế
Mỹ thuật số

07 Mã ĐKXT: 7480108
Công nghệ kỹ thuật máy tính
60 chỉ tiêu
Hệ Kỹ sư

08 Mã ĐKXT: 7480108B
Công nghệ kỹ thuật máy tính
60 chỉ tiêu
Hệ Cử nhân

9 Mã ĐKXT: 7480108IC New
Chuyên ngành Thiết kế
Vi mạch bán dẫn
60 chỉ tiêu - Hệ Kỹ sư

10 Mã ĐKXT: 7480202
An toàn thông tin
60 chỉ tiêu
Hệ Kỹ sư

11 Mã ĐKXT: 7340101
Quản trị kinh doanh
120 chỉ tiêu
Hệ Cử nhân

12 Mã ĐKXT: 7340115 New
Marketing
40 chỉ tiêu
Hệ Cử nhân

13 Mã ĐKXT: 7340101EL
Chuyên ngành Quản trị
Logistics và Chuỗi cung ứng số
130 chỉ tiêu - Hệ Cử nhân

14 Mã ĐKXT: 7340101ET
Chuyên ngành Quản trị
Dịch vụ du lịch và Lữ hành số
60 chỉ tiêu - Hệ Cử nhân

15 Mã ĐKXT: 7340101EF
Chuyên ngành Quản trị
Tài chính số
60 chỉ tiêu - Hệ Cử nhân

THÔNG BÁO

THÔNG BÁO XÉT TUYỂN THẲNG VÀ ƯU TIÊN XÉT TUYỂN THEO QUY ĐỊNH CỦA BỘ GDĐT NĂM 2024 (TUYỂN THẲNG)

Thời gian: từ ngày 15/4/2024 đến hết ngày 31/5/2024

Xem chi tiết tại đây

THÔNG BÁO XÉT TUYỂN THẲNG THEO ĐỀ ÁN CỦA TRƯỜNG (TUYỂN SINH RIÊNG) Năm 2024

VKUBOT

● We are online

CHÀO! Tôi là một chatbot. Hôm nay tôi có thể giúp gì cho bạn?

Figure. Homepage VKU Chatbot

Training Model:

1. Overview

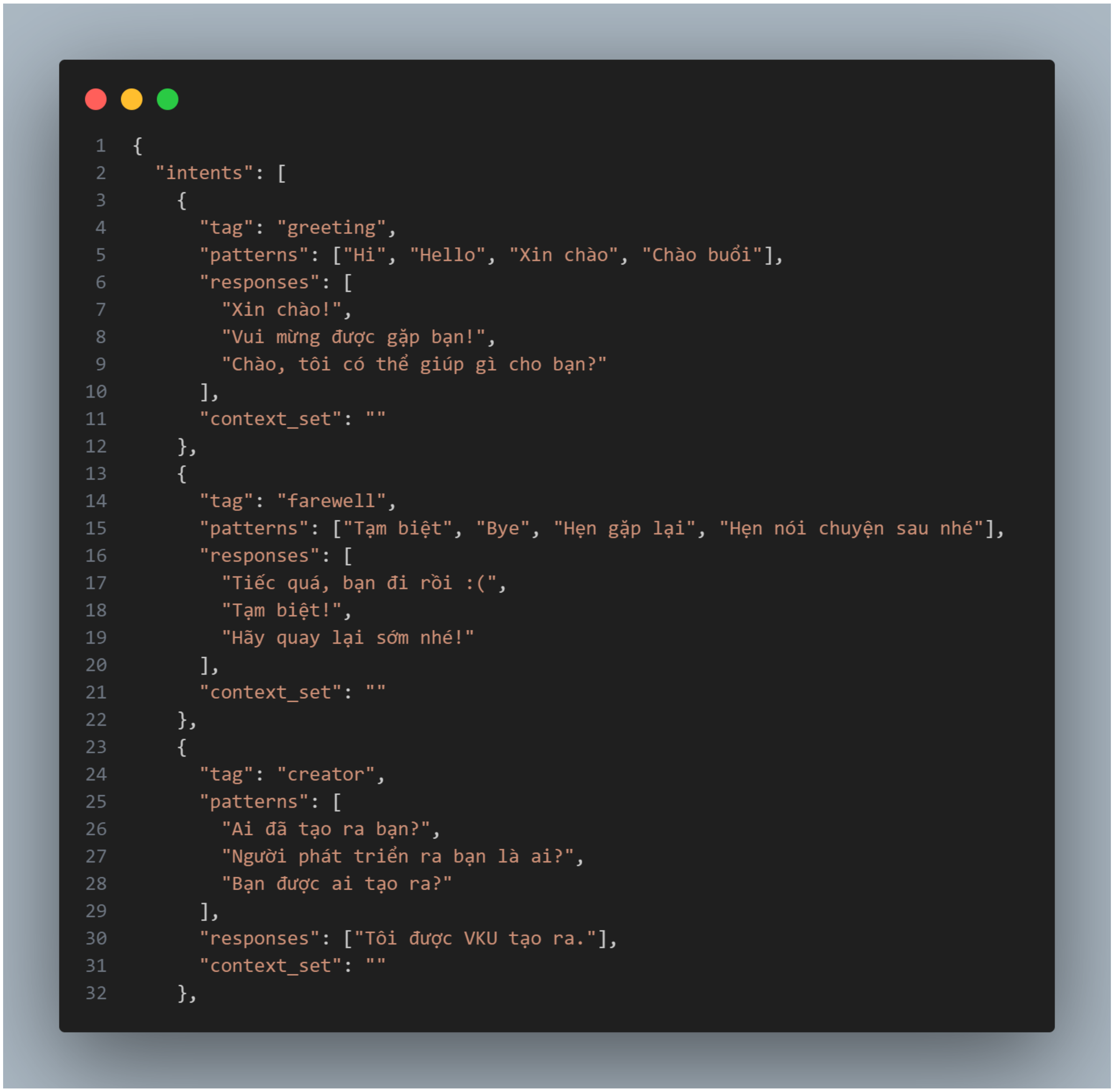
The VKU Chatbot project aims to develop a chatbot that can understand and respond to user queries related to college information. This document outlines the process of preparing the training data, preprocessing it, augmenting the data, extracting features, training multiple models, selecting the best-performing model, and saving the model and data for future use. The training process involves:

- Data Preprocessing
- Data Augmentation using Synonym Replacement
- Feature Extraction with TF-IDF Vectorization
- Model Selection using GridSearchCV for Hyperparameter Tuning
- Saving the Best Performing Model

2. Prepare Dataset

2.1. Intents Data

The chatbot training begins with defining intents data, which includes various user intents, patterns, and corresponding responses. This data is stored in a JSON-like format. Each intent represents a category of user queries and contains examples of possible user inputs (patterns) and appropriate chatbot responses. Below is an example of the intents data structure:



```
{
  "intents": [
    {
      "tag": "greeting",
      "patterns": ["Hi", "Hello", "Xin chào", "Chào buổi"],
      "responses": [
        "Xin chào!",
        "Vui mừng được gặp bạn!",
        "Chào, tôi có thể giúp gì cho bạn?"
      ],
      "context_set": ""
    },
    {
      "tag": "farewell",
      "patterns": ["Tạm biệt", "Bye", "Hẹn gặp lại", "Hẹn nói chuyện sau nhé"],
      "responses": [
        "Tiếc quá, bạn đi rồi :(",
        "Tạm biệt!",
        "Hãy quay lại sớm nhé!"
      ],
      "context_set": ""
    },
    {
      "tag": "creator",
      "patterns": [
        "Ai đã tạo ra bạn?",
        "Người phát triển ra bạn là ai?",
        "Bạn được ai tạo ra?"
      ],
      "responses": ["Tôi được VKU tạo ra."],
      "context_set": ""
    },
```

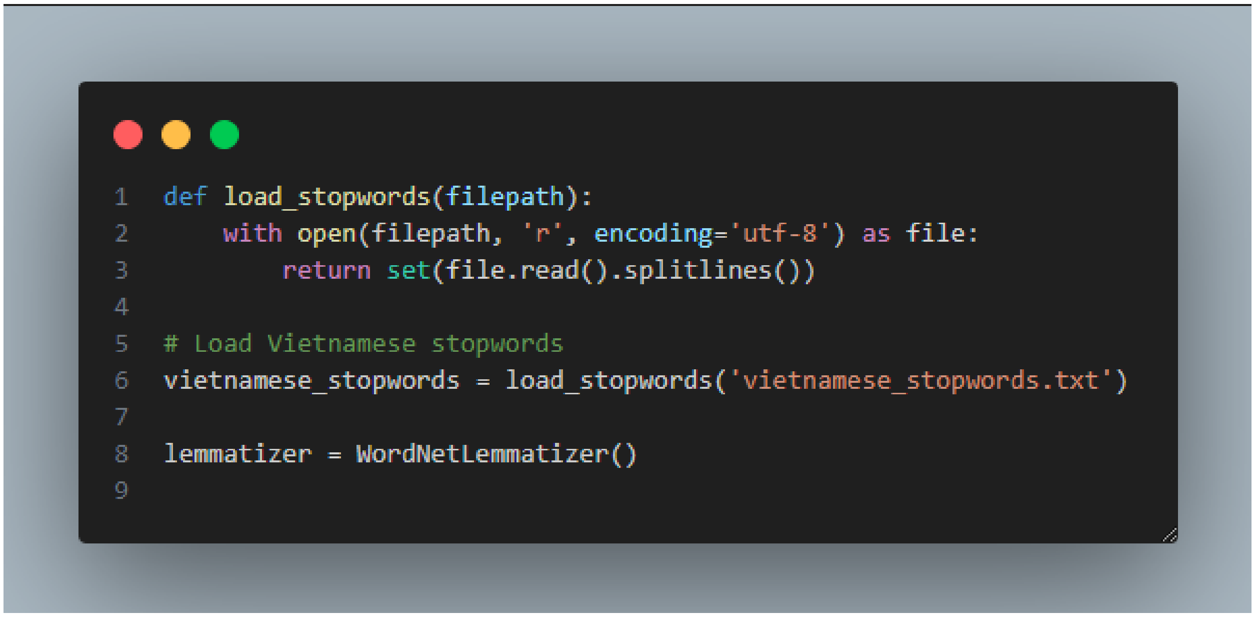
- tag (class label for user input)
- patterns - input sentence samples trained for classification
- responses - responses (bots) are mapped to respond to previous requests.

3. Data Preprocessing

Data preprocessing involves cleaning and preparing the raw text data by tokenizing, removing stopwords, and lemmatizing. This step is crucial for ensuring that the data is in a suitable format for training machine learning models.

3.1 Loading Vietnamese Stopwords

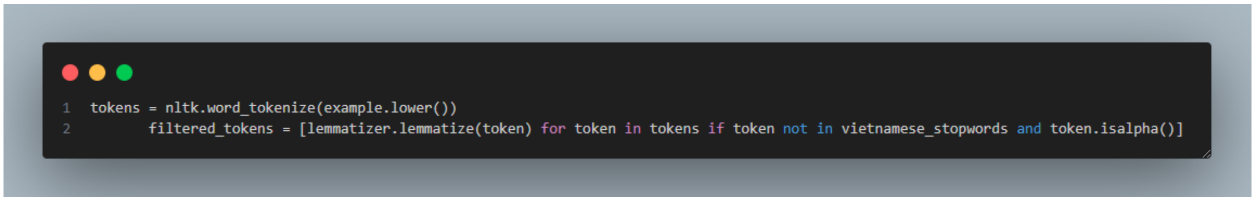
To ensure that the chatbot can handle Vietnamese input, Vietnamese stopwords are loaded from an external file. Stopwords are common words that do not carry significant meaning and can be removed to reduce noise in the data.



```python
def load_stopwords(filepath):
    with open(filepath, 'r', encoding='utf-8') as file:
        return set(file.read().splitlines())

# Load Vietnamese stopwords
vietnamese_stopwords = load_stopwords('vietnamese_stopwords.txt')

lemmatizer = WordNetLemmatizer()

```

3.2 Preprocessing Function

A function to preprocess each sentence by tokenizing, removing stopwords, and lemmatizing. Tokenization splits sentences into individual words, lemmatization reduces words to their base or root form, and stopword removal eliminates common words that do not contribute to the meaning.



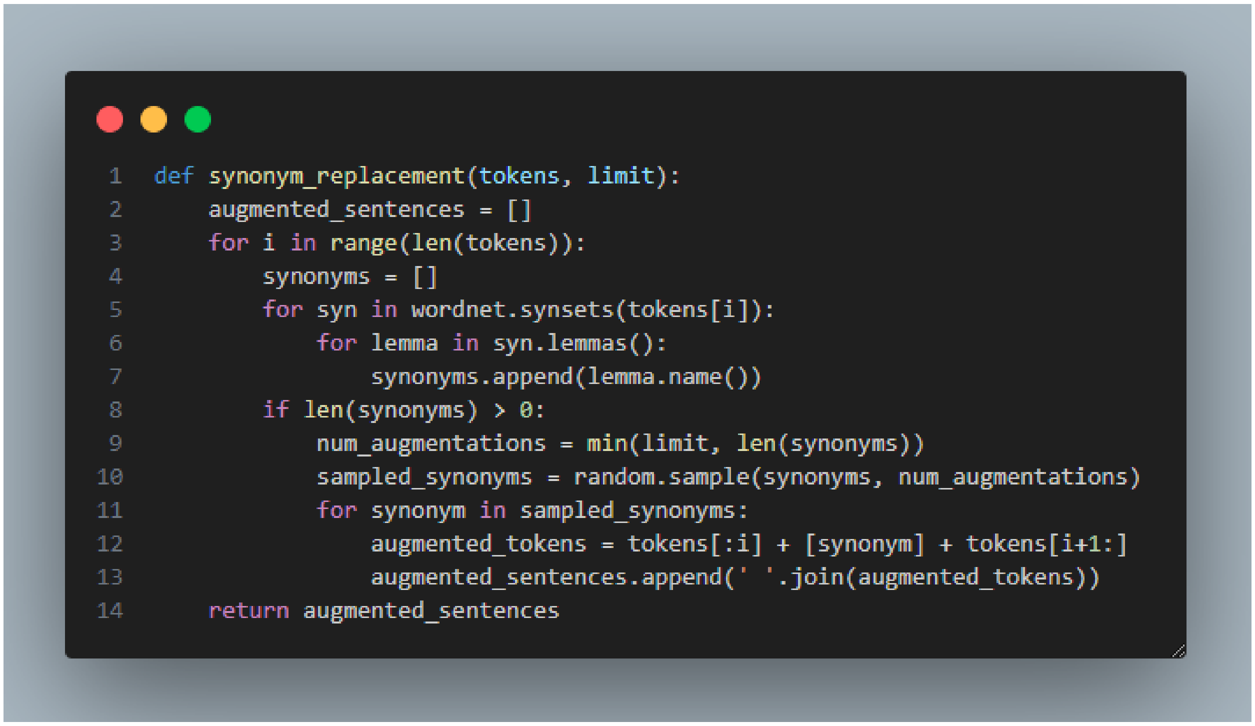
```python
tokens = nltk.word_tokenize(example.lower())
      filtered_tokens = [lemmatizer.lemmatize(token) for token in tokens if token not in vietnamese_stopwords and token.isalpha()]
```

4. Data Augmentation

Data augmentation expands the training dataset by generating new examples through synonym replacement. This technique helps increase the diversity and quantity of training data, improving the model's generalization capability.

4.1 Synonym Replacement Function

A function to perform synonym replacement on the tokenized sentences. This function replaces each word in a sentence with its synonyms to create new, augmented sentences.



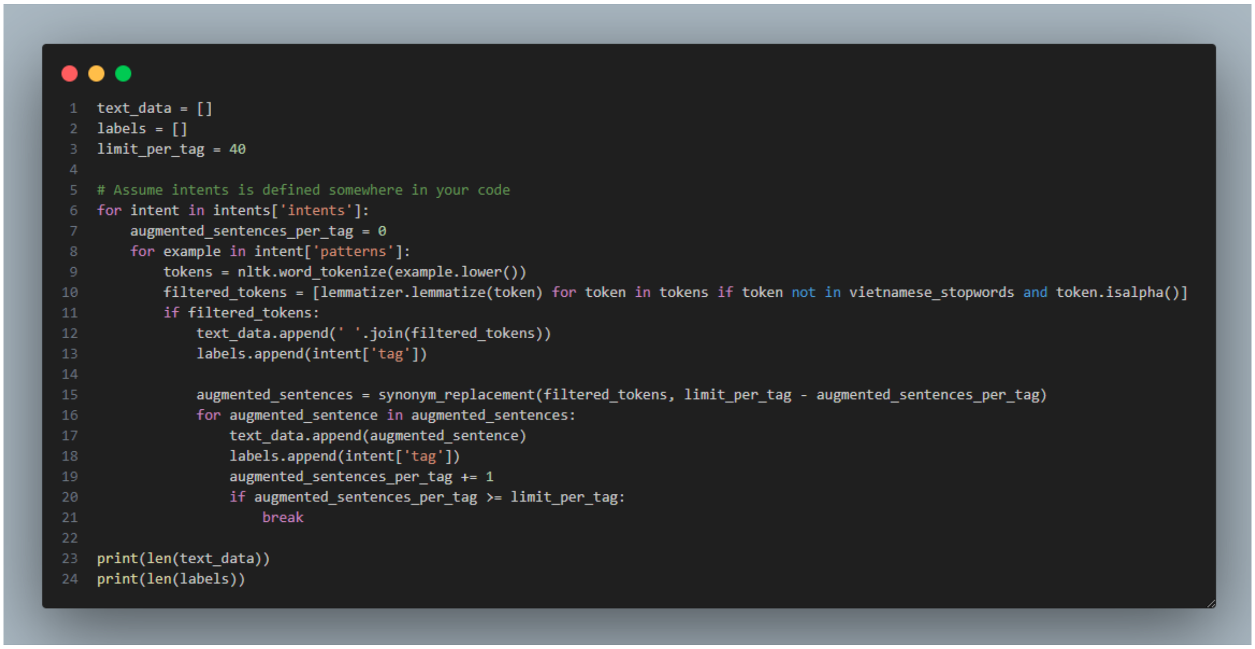
```python
def synonym_replacement(tokens, limit):
    augmented_sentences = []
    for i in range(len(tokens)):
        synonyms = []
        for syn in wordnet.synsets(tokens[i]):
            for lemma in syn.lemmas():
                synonyms.append(lemma.name())
        if len(synonyms) > 0:
            num_augmentations = min(limit, len(synonyms))
            sampled_synonyms = random.sample(synonyms, num_augmentations)
            for synonym in sampled_synonyms:
                augmented_tokens = tokens[:i] + [synonym] + tokens[i+1:]
                augmented_sentences.append(' '.join(augmented_tokens))
    return augmented_sentences
```

5. Preparing the Data

Prepare the data by preprocessing and augmenting it, then extract features using TF-IDF vectorization. This step transforms the text data into numerical features suitable for machine learning models.

5.1 Preprocessing and Augmenting Data

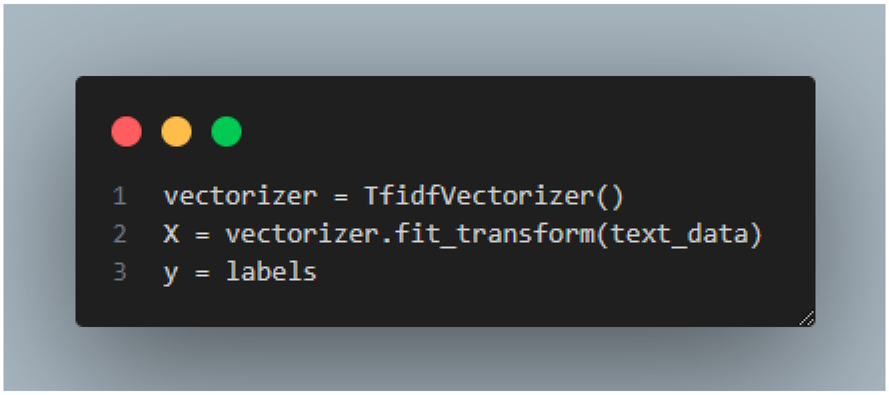
Iterate over the intents, preprocess the sentences, and augment the data using the synonym replacement function. This process increases the number of training examples for each intent.



```python
text_data = []
labels = []
limit_per_tag = 40

# Assume intents is defined somewhere in your code
for intent in intents['intents']:
    augmented_sentences_per_tag = 0
    for example in intent['patterns']:
        tokens = nltk.word_tokenize(example.lower())
        filtered_tokens = [lemmatizer.lemmatize(token) for token in tokens if token not in vietnamese_stopwords and token.isalpha()]
        if filtered_tokens:
            text_data.append(' '.join(filtered_tokens))
            labels.append(intent['tag'])

            augmented_sentences = synonym_replacement(filtered_tokens, limit_per_tag - augmented_sentences_per_tag)
            for augmented_sentence in augmented_sentences:
                text_data.append(augmented_sentence)
                labels.append(intent['tag'])
                augmented_sentences_per_tag += 1
                if augmented_sentences_per_tag >= limit_per_tag:
                    break

print(len(text_data))
print(len(labels))
```

5.2 Feature Extraction

Extract features from the text data using TF-IDF vectorization. TF-IDF (Term Frequency-Inverse Document Frequency) is a numerical statistic that reflects the importance of a word in a document relative to a collection of documents.



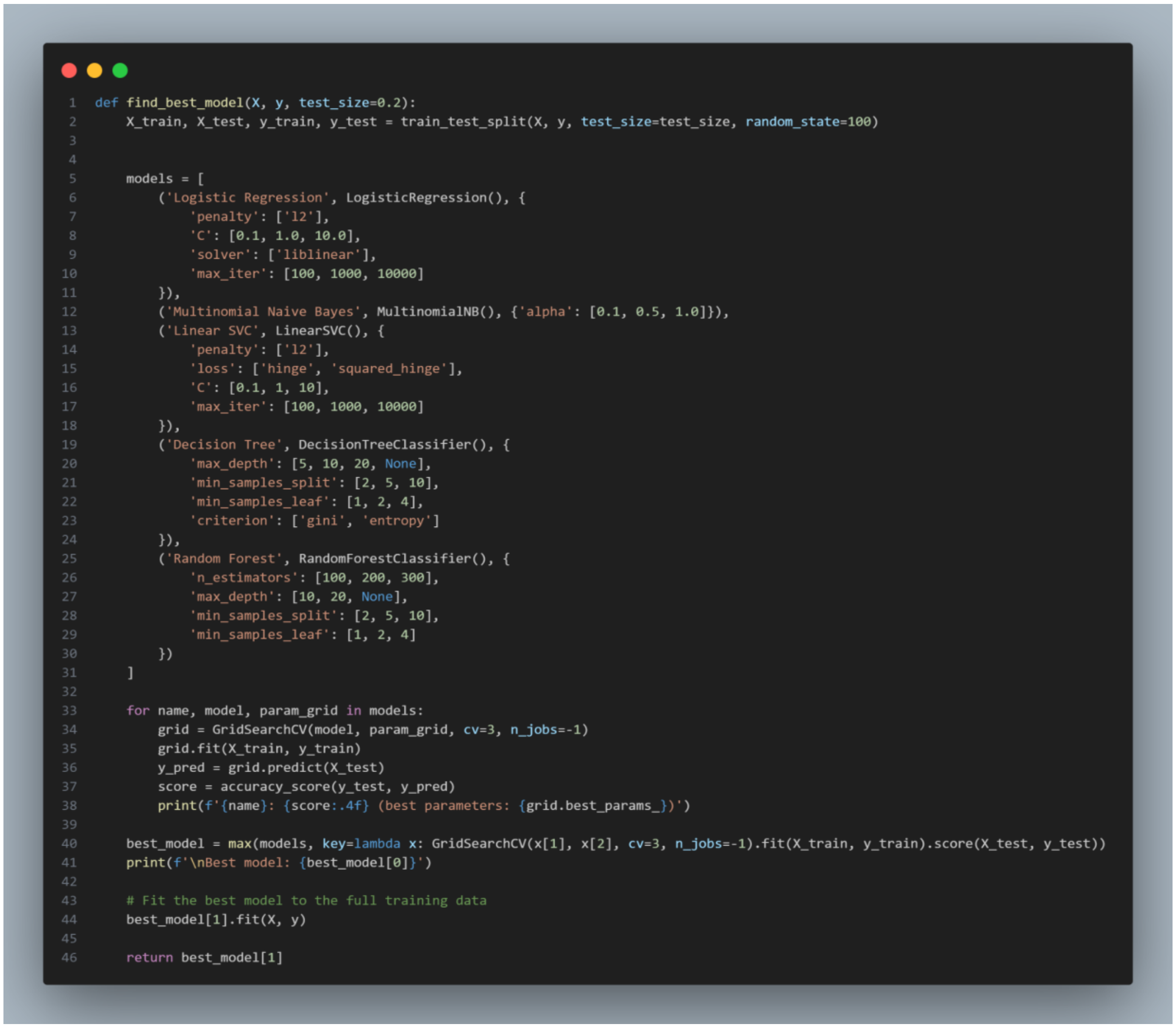
```python
vectorizer = TfidfVectorizer()
X = vectorizer.fit_transform(text_data)
y = labels
```

6. Model Training and Selection

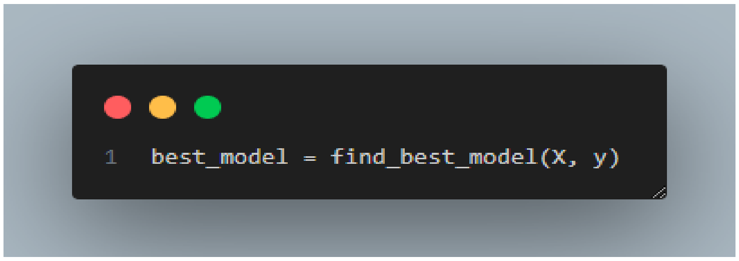
Train multiple models and select the best one using GridSearchCV for hyperparameter tuning. This step involves training different classifiers and evaluating their performance to choose the most accurate model.

6.1 Training Models

Define a function to train multiple models and select the best one based on accuracy. The function uses GridSearchCV to find the optimal hyperparameters for each model.



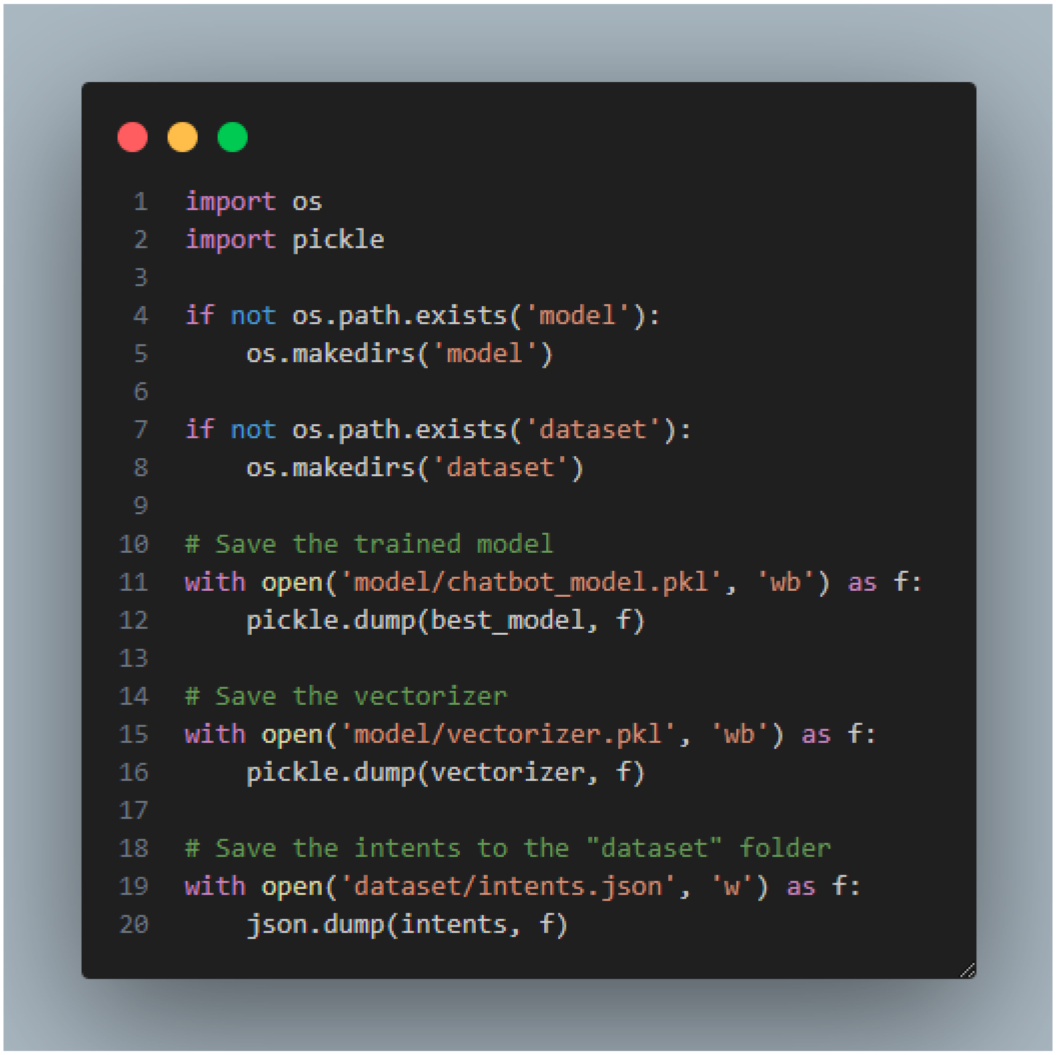
```python
def find_best_model(X, y, test_size=0.2):
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=test_size, random_state=100)


    models = [
        ('Logistic Regression', LogisticRegression(), {
            'penalty': ['l2'],
            'C': [0.1, 1.0, 10.0],
            'solver': ['liblinear'],
            'max_iter': [100, 1000, 10000]
        }),
        ('Multinomial Naive Bayes', MultinomialNB(), {'alpha': [0.1, 0.5, 1.0]}),
        ('Linear SVC', LinearSVC(), {
            'penalty': ['l2'],
            'loss': ['hinge', 'squared_hinge'],
            'C': [0.1, 1, 10],
            'max_iter': [100, 1000, 10000]
        }),
        ('Decision Tree', DecisionTreeClassifier(), {
            'max_depth': [5, 10, 20, None],
            'min_samples_split': [2, 5, 10],
            'min_samples_leaf': [1, 2, 4],
            'criterion': ['gini', 'entropy']
        }),
        ('Random Forest', RandomForestClassifier(), {
            'n_estimators': [100, 200, 300],
            'max_depth': [10, 20, None],
            'min_samples_split': [2, 5, 10],
            'min_samples_leaf': [1, 2, 4]
        })
    ]

    for name, model, param_grid in models:
        grid = GridSearchCV(model, param_grid, cv=3, n_jobs=-1)
        grid.fit(X_train, y_train)
        y_pred = grid.predict(X_test)
        score = accuracy_score(y_test, y_pred)
        print(f'{name}: {score:.4f} (best parameters: {grid.best_params_})')

    best_model = max(models, key=lambda x: GridSearchCV(x[1], x[2], cv=3, n_jobs=-1).fit(X_train, y_train).score(X_test, y_test))
    print(f'\nBest model: {best_model[0]}')

    # Fit the best model to the full training data
    best_model[1].fit(X, y)

    return best_model[1]
```



```python
best_model = find_best_model(X, y)
```

7. Saving the Best Model

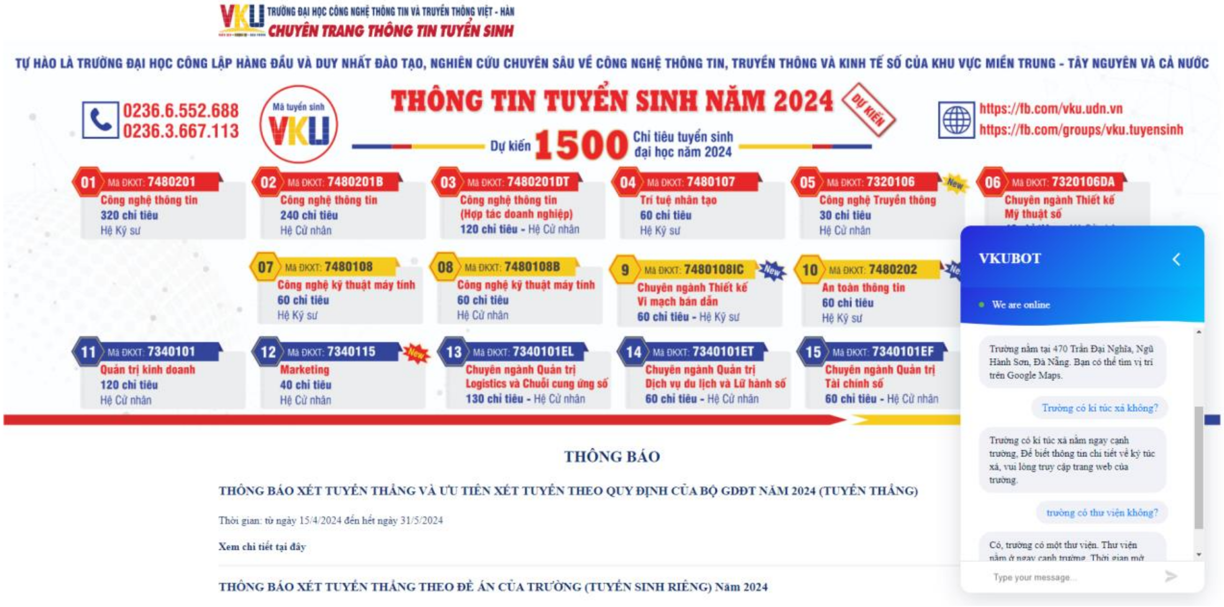
Save the trained model and the vectorizer for future use. This allows the chatbot to use the trained model and vectorizer during deployment.



```python
import os
import pickle

if not os.path.exists('model'):
    os.makedirs('model')

if not os.path.exists('dataset'):
    os.makedirs('dataset')

# Save the trained model
with open('model/chatbot_model.pkl', 'wb') as f:
    pickle.dump(best_model, f)

# Save the vectorizer
with open('model/vectorizer.pkl', 'wb') as f:
    pickle.dump(vectorizer, f)

# Save the intents to the "dataset" folder
with open('dataset/intents.json', 'w') as f:
    json.dump(intents, f)
```

Results:

The VKU Chatbot project achieved several key outcomes, demonstrating the application of Natural Language Processing (NLP) and Machine Learning (ML) techniques to create an interactive and user-friendly chatbot.


TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG VIỆT - HÀN
CHUYÊN TRANG THÔNG TIN TUYỂN SINH
TỰ HÀO LÀ TRƯỜNG ĐẠI HỌC CÔNG LẬP HÀNG ĐẦU VÀ DUY NHẤT ĐÀO TẠO, NGHIÊN CỨU CHUYÊN SÂU VỀ CÔNG NGHỆ THÔNG TIN, TRUYỀN THÔNG VÀ KINH TẾ SỐ CỦA KHU VỰC MIỀN TRUNG - TÂY NGUYÊN VÀ CẢ NƯỚC
0236.6.552.688
0236.3.667.113
Mã tuyển sinh VKU
THÔNG TIN TUYỂN SINH NĂM 2024
DỰ KIẾN
Dự kiến 1500 Chỉ tiêu tuyển sinh đại học năm 2024
https://fb.com/vku.udn.vn
https://fb.com/groups/vku.tuyensinh
01 Mã ĐKXT: 7480201 Công nghệ thông tin 320 chỉ tiêu Hệ Kỹ sư
02 Mã ĐKXT: 7480201B Công nghệ thông tin 240 chỉ tiêu Hệ Cử nhân
03 Mã ĐKXT: 7480201DT Công nghệ thông tin (Hợp tác doanh nghiệp) 120 chỉ tiêu - Hệ Cử nhân
04 Mã ĐKXT: 7480107 Trí tuệ nhân tạo 60 chỉ tiêu Hệ Kỹ sư
05 Mã ĐKXT: 7320106 New Công nghệ Truyền thông 30 chỉ tiêu Hệ Cử nhân
06 Mã ĐKXT: 7320106DA Chuyên ngành Thiết kế Mỹ thuật số
07 Mã ĐKXT: 7480108 Công nghệ kỹ thuật máy tính 60 chỉ tiêu Hệ Kỹ sư
08 Mã ĐKXT: 7480108B Công nghệ kỹ thuật máy tính 60 chỉ tiêu Hệ Cử nhân
9 Mã ĐKXT: 7480108IC New Chuyên ngành Thiết kế Vi mạch bán dẫn 60 chỉ tiêu - Hệ Kỹ sư
10 Mã ĐKXT: 7480202 An toàn thông tin 60 chỉ tiêu Hệ Kỹ sư
11 Mã ĐKXT: 7340101 Quản trị kinh doanh 120 chỉ tiêu Hệ Cử nhân
12 Mã ĐKXT: 7340115 New Marketing 40 chỉ tiêu Hệ Cử nhân
13 Mã ĐKXT: 7340101EL Chuyên ngành Quản trị Logistics và Chuỗi cung ứng số 130 chỉ tiêu - Hệ Cử nhân
14 Mã ĐKXT: 7340101ET Chuyên ngành Quản trị Dịch vụ du lịch và Lữ hành số 60 chỉ tiêu - Hệ Cử nhân
15 Mã ĐKXT: 7340101EF Chuyên ngành Quản trị Tài chính số 60 chỉ tiêu - Hệ Cử nhân
VKUBOT
We are online
Type your message...

THÔNG BÁO

THÔNG BÁO XÉT TUYỂN THẲNG VÀ ƯU TIÊN XÉT TUYỂN THEO QUY ĐỊNH CỦA BỘ GDĐT NĂM 2024 (TUYỂN THẲNG)

Thời gian: từ ngày 15/4/2024 đến hết ngày 31/5/2024

Xem chi tiết tại đây

THÔNG BÁO XÉT TUYỂN THẲNG THEO ĐỀ ÁN CỦA TRƯỜNG (TUYỂN SINH RIÊNG) Năm 2024

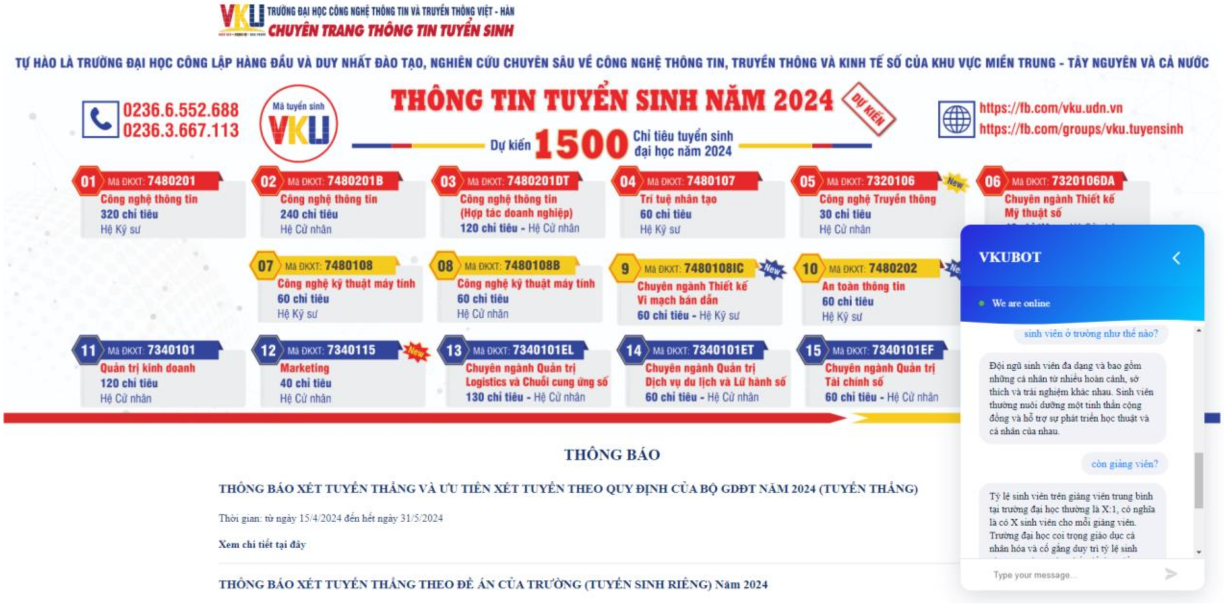
Trường nằm tại 470 Trần Đại Nghĩa, Ngũ Hành Sơn, Đà Nẵng. Bạn có thể tìm vị trí trên Google Maps.

Trường có kí túc xá không?

Trường có kí túc xá nằm ngay cạnh trường, Để biết thông tin chi tiết về ký túc xá, vui lòng truy cập trang web của trường.

trường có thư viện không?

Có, trường có một thư viện. Thư viện nằm ở ngay cạnh trường. Thời gian mở


TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG VIỆT - HÀN
CHUYÊN TRANG THÔNG TIN TUYỂN SINH
THÔNG TIN TUYỂN SINH NĂM 2024
VKUBOT
We are online
Type your message...

THÔNG BÁO

THÔNG BÁO XÉT TUYỂN THẲNG VÀ ƯU TIÊN XÉT TUYỂN THEO QUY ĐỊNH CỦA BỘ GDĐT NĂM 2024 (TUYỂN THẲNG)

Thời gian: từ ngày 15/4/2024 đến hết ngày 31/5/2024

Xem chi tiết tại đây

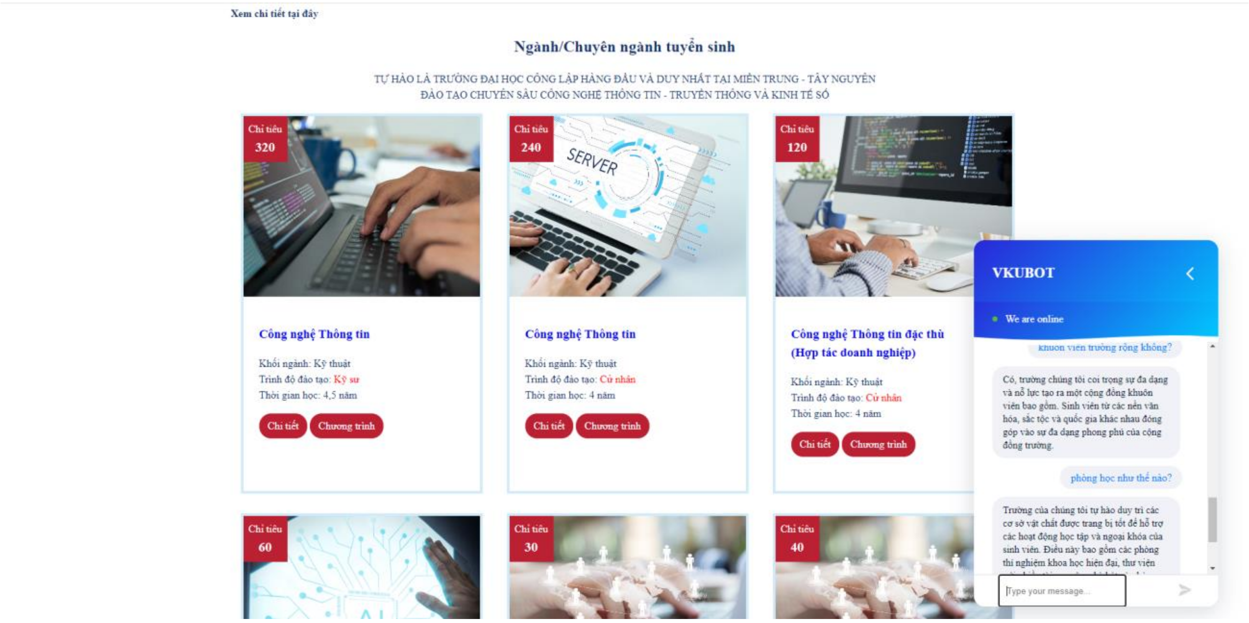
THÔNG BÁO XÉT TUYỂN THẲNG THEO ĐỀ ÁN CỦA TRƯỜNG (TUYỂN SINH RIÊNG) Năm 2024

sinh viên ở trường như thế nào?

Đội ngũ sinh viên đa dạng và bao gồm những cá nhân từ nhiều hoàn cảnh, sở thích và trải nghiệm khác nhau. Sinh viên thường nuôi dưỡng một tinh thần cộng đồng và hỗ trợ sự phát triển học thuật và cá nhân của nhau.

còn giảng viên?

Tỷ lệ sinh viên trên giảng viên trung bình tại trường đại học thường là X:1, có nghĩa là có X sinh viên cho mỗi giảng viên. Trường đại học coi trọng giáo dục cá nhân hóa và cố gắng duy trì tỷ lệ sinh



Xem chi tiết tại đây

Ngành/Chuyên ngành tuyển sinh

TỰ HÀO LÀ TRƯỜNG ĐẠI HỌC CÔNG LẬP HÀNG ĐẦU VÀ DUY NHẤT TẠI MIỀN TRUNG - TÂY NGUYÊN
ĐÀO TẠO CHUYÊN SÂU CÔNG NGHỆ THÔNG TIN - TRUYỀN THÔNG VÀ KINH TẾ SỐ

Chỉ tiêu 320

Công nghệ Thông tin

Khối ngành: Kỹ thuật
Trình độ đào tạo: Kỹ sư
Thời gian học: 4,5 năm

Chỉ tiêu 240

Công nghệ Thông tin

Khối ngành: Kỹ thuật
Trình độ đào tạo: Cử nhân
Thời gian học: 4 năm

Chỉ tiêu 120

Công nghệ Thông tin đặc thù (Hợp tác doanh nghiệp)

Khối ngành: Kỹ thuật
Trình độ đào tạo: Cử nhân
Thời gian học: 4 năm

Chỉ tiêu 60

Chỉ tiêu 30

Chỉ tiêu 40

VKUBOT

We are online

khuon vien trường rộng không?

Có, trường chúng tôi coi trọng sự đa dạng và nỗ lực tạo ra một cộng đồng khuôn viên bao gồm. Sinh viên từ các nền văn hóa, sắc tộc và quốc gia khác nhau đóng góp vào sự đa dạng phong phú của cộng đồng trường.

phòng học như thế nào?

Trường của chúng tôi tự hào duy trì các cơ sở vật chất được trang bị tốt để hỗ trợ các hoạt động học tập và ngoại khóa của sinh viên. Điều này bao gồm các phòng thí nghiệm khoa học hiện đại, thư viện

Conclusion:

The VKU Chatbot project represents a significant endeavor in applying natural language processing (NLP) and machine learning techniques to create an intelligent conversational agent. Designed to assist users with college-related inquiries, the chatbot aims to enhance the user experience by providing quick and accurate responses. Here are the key takeaways and future directions:

Key Achievements:

- Comprehensive Data Handling: The project began with careful preparation of the intent data, defining clear patterns and responses to ensure the chatbot can understand and respond to a wide range of user inputs effectively.

- Effective Data Preprocessing: By tokenizing, lemmatizing, and removing stopwords, the project ensured that the text data fed into the model was clean and relevant, which is crucial for accurate model training.

- Innovative Data Augmentation: Utilizing synonym replacement techniques, the project expanded the training dataset, increasing its diversity and improving the model's ability to generalize to new, unseen data.

- Advanced Feature Extraction: The use of TF-IDF vectorization transformed the textual data into numerical features, enabling the application of machine learning algorithms.

- Robust Model Training and Selection: Multiple machine learning models were trained and evaluated, with GridSearchCV employed to fine-tune hyperparameters and select the best-performing model.

- Model Persistence: The trained model and vectorizer were saved for future use, ensuring the chatbot can be deployed and utilized without retraining.

Impact:

The VKU Chatbot is designed to significantly improve user interaction by:

- Providing instant and accurate responses to common queries.
- Reducing the workload on human support staff by handling routine inquiries.
- Enhancing the overall user experience with a reliable, accessible source of information.

Future Directions:

- Model Enhancement: Continuously updating the model with new data and feedback to improve its accuracy and robustness.

- Deployment: Integrating the model into a user-friendly interface, such as a web application using Flask, to make it accessible to users in real-time.

- Monitoring and Maintenance: Regular monitoring of the chatbot's performance and periodic updates to maintain its relevance and accuracy.

- Expanding Capabilities: Adding support for multiple languages and expanding the range of queries the chatbot can handle.

- Integration with Other Systems: Connecting the chatbot to other college systems (e.g., course registration, events schedules) to provide a more comprehensive assistance tool.

The VKU Chatbot project stands as a testament to the power of NLP and machine learning in creating practical and valuable applications. While it may not be state-of-the-art, it represents a meaningful application of theoretical knowledge to solve real-world problems. The project serves as a foundational step toward more advanced and capable conversational agents, contributing positively to the VKU community and setting the stage for future innovations.

By documenting each step in detail and ensuring a thorough understanding of the processes involved, the VKU Chatbot project not only provides a useful tool but also serves as an educational resource for those looking to delve into the world of NLP and chatbot development.

References:

https://www.ijmerr.com/uploadfile/2020/0312/20200312023706525.pdf
https://cyfuture.com/blog/a-comprehensive-guide-to-chatbot-machine-learning/
https://viblo.asia/p/ap-dung-machine-learning-xay-dung-ung-dung-chatbot-cua-rieng-ban-3P0lPk38Zox
https://nimapinfotech.com/how-to-make-chatbot-using-ml/
https://www.weathercompany.com/blog/the-ultimate-guide-to-machine-learning-chatbots-and-conversational-ai/
https://www.researchgate.net/publication/349546138_Chatbots_and_Communication_The_Growing_Role_of_Artificial_Intelligence_in_Addressing_and_Shaping_Customer_Needs#pf3