BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**🙞 🕮 🙜**



**LUẬN VĂN TỐT NGHIỆP**

**NGÀNH CÔNG NGHỆ THÔNG TIN**

**Đề tài**

**ỨNG DỤNG PHÁT CẢNH BÁO CHO GIÁO VIÊN VỀ TÌNH TRẠNG NGỦ TRƯA**

**CỦA TRẺ MẦM NON**

**Giáo viên hướng dẫn: Sinh viên thực hiện:**

**TS.Trần Nguyễn Minh Thư Trần Chiến Thành**

**Mã số: B1605365**

**Khóa : 42**

Cần Thơ, 12/2020

**NHẬN XÉT CỦA GIẢNG VIÊN**

**--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------**

**LỜI CẢM ƠN**

Để hoàn thành bài luận văn này, em xin được bày tỏ lòng biết ơn chân thành và sâu sắc đến Cô Trần Nguyễn Minh Thư – người đã trực tiếp tận tình hướng dẫn,giúp đỡ em.Trong suốt quá trình thực hiện niên luận, nhờ những sự chỉ bảo và hướng dẫn quý giá đó mà bài luận văn này được hoàn thành một cách tốt nhất.

Em cũng xin gửi lời cám ơn chân thành đến các Thầy Cô Giảng viên Đại học Cần Thơ, đặc biệt là các Thầy Cô ở Khoa CNTT & TT, những người đã truyền đạt những kiến thức quý báu trong thời gian qua.

Em cũng xin chân thành cảm ơn bạn bè cùng với gia đình đã luôn động viên, khích lệ và tạo điều kiện giúp đỡ trong suốt quá trình thực hiện để em có thể hoàn thành bài niên luận một cách tốt nhất.

Tuy có nhiều cố gắng trong quá trình thực hiện niên luận,nhưng không thể tránh khỏi những sai sót.Em rất mong nhận được sự đóng góp ý kiến quý báu của quý Thầy Cô và các bạn để bài niên luận hoàn thiện hơn.

Cần Thơ, ngày .. tháng 12 năm 2020

Người viết

Trần Chiến Thành

**MỤC LỤC**

[PHẦN GIỚI THIỆU 3](#_Toc26317870)

[1. Đặt vấn đề 3](#_Toc26317871)

[2. Lịch sử giải quyết vấn đề 3](#_Toc26317872)

[3. Mục tiêu đề tài 4](#_Toc26317873)

[4. Đối tượng và phạm vi nghiên cứu 4](#_Toc26317874)

[5. Phương pháp nghiên cứu 4](#_Toc26317875)

[6. Bố cục luận văn 5](#_Toc26317876)

[PHẦN NỘI DUNG 6](#_Toc26317877)

[CHƯƠNG 1 6](#_Toc26317878)

[MÔ TẢ BÀI TOÁN 6](#_Toc26317879)

[1. Mô tả chi tiết bài toán 6](#_Toc26317880)

[2. Vấn đề và giải pháp liên quan đến bài toán 6](#_Toc26317881)

[Các công việc chính để xây dựng Chatbot bao gồm: 6](#_Toc26317882)

[CHƯƠNG 2 7](#_Toc26317883)

[THIẾT KẾ VÀ CÀI ĐẶT 7](#_Toc26317884)

[1. Thiết kế hệ thống 7](#_Toc26317885)

[2. Xây dựng hệ thống 8](#_Toc26317886)

[CHƯƠNG 3 14](#_Toc26317887)

[KIỂM THỬ VÀ ĐÁNH GIÁ 14](#_Toc26317888)

[PHẦN KẾT LUẬN 17](#_Toc26317890)

[TÀI LIỆU THAM KHẢO 18](#_Toc26317891)

**TÓM TẮT**

Tiến hành xây dựng Chatbot trên máy tính có thể trả lời tự động cho sinh viên các câu hỏi liên quan đến điều kiện xét và công nhận tốt nghiệp. Sinh viên thực hiện đặt câu hỏi cho Chatbot bằng cách nói trực tiếp bằng giọng nói. Chatbot tiếp nhận câu hỏi thông qua microphone và chuyển đổi câu hỏi từ dạng tiếng nói sang dạng văn bản, nhận dạng văn bản bằng mô hình phân loại K láng giềng đã được huyến luyện với tập dữ liệu gồm 190 câu hỏi cho 19 câu trả lời từ. Các câu trả lời được lấy từ quy chế học vụ, mục IV – điều kiện xét và công nhận tốt nghiệp của trường đại học Cần Thơ. Kết quả thực nghiệm Chatbot sử dụng giải thuật K láng giềng để phân loại câu hỏi cho độ chính xác 78%.

# PHẦN GIỚI THIỆU

# 

## 1. Đặt vấn đề

Chatbot là một cụm từ không còn xa lạ đối với nền công nghệ phát triển hiện nay trên thế giới. Về cơ bản, Chatbot là một chương trình máy tính tiến hành cuộc trò chuyện thông qua nhắn tin nhanh hoặc âm thanh, nó có thể tự động trả lời những câu hỏi hoặc xử lý tình huống. Phạm vi và sự phức tạp của Chatbot được xác định bởi thuật toán của người tạo nên chúng. Trên thực tế, Chatbot nhanh hơn hẳn con người về tốc độ đưa ra câu trả lời. Do vậy, Chatbot thường được sử dụng rộng rãi cho nhiều mục đích khác nhau bao gồm: giải trí, nghiên cứu, dịch vụ khách hàng, quảng bá sản phẩm hoặc thu thập thông tin,... và giáo dục cũng không phải là ngoại lệ. Đối với sinh viên, đặc biệt là những sinh viên năm cuối, còn có những bạn vẫn chưa nắm hết, nắm rõ chi tiết về các điều kiện để xét và nhận bằng tốt nghiệp. Tỷ lệ sinh viên không hiểu rõ về các điều kiện xét và công nhận tốt nghiệp trong quy chế học vụ dẫn đến không đủ điều kiện tốt nghiệp, dù ít những không phải là không tồn tại. Vì lẽ đó, việc xây dựng Chatbot có khả năng tư động trả lời, giải đáp thắc mắc về các điều kiện xét và công nhận tốt nghiệp là điều cần thiết. Với Chatbot, sinh viên có thể nhận được đáp án một cách đơn giản, chính xác và nhanh chóng trong vài giây thay vì tốn thời gian tra cứu quy chế học vụ và tìm câu trả lời.

## 2. Lịch sử giải quyết vấn đề

Cùng với sự đa dụng của Chatbot, không ít dự án Chatbot hỗ trợ giáo dục đã ra đời. Trong đó phải kể đến nghiêm cứu “Chatbot for information technology students” của thầy Đỗ Thanh Nghị - trường Đại học Cần Thơ và thầy Hoàng Tùng – trường Đại học Nguyễn Tất Thành, TP.Hồ Chí Minh. Trong dự án này, ChatBot được xây dựng trên máy tính Raspberry Pi có thể trả lời tự động cho sinh viên nghành Công nghệ thông tin (CNTT) các câu hỏi liên quan đến môi trường học tập và phương pháp học tập bậc đại học, kỹ năng nghề nghiệp, và xu hướng công nghệ. Sau khi biên soạn tập dữ liệu gồm 986 câu hỏi của 213 câu trả lời, các câu hỏi trải qua quá trình tách từ, tiền xử lý và biểu diễn dưới dạng mô hình túi từ (bag-of-words). Tiếp đó, tác giả huấn luyện cách bộ phân lớp như máy học véc-tơ hỗ trợ (support vector machine), mạng nơ-ron (neural networks), rừng ngẫu nhiên (random forest), k láng giềng (k nearest neighbors). Đối với phương pháp huấn luyện sử dụng giải thuật k láng giềng, độ chính xác khi dự đoán câu trả lời ở mức 65.66%. [1]

Không chỉ trong nước, Chatbot đã được áp dụng phổ biến và rộng khắp ở nước ngoài. Ví dụ như Mondly Conversational Chatbot, được tạo ra bởi Brasov, Transylvania và Romania. Mondly Chatbot tương tác với người sử dụng, thấu hiểu giọng nói đầu vào và đáp lời bằng giọng của con người. Ứng dụng sử dụng công nghệ nhận dạng giọng nói cùng với một bộ nhận dạng văn bản của Mondly để đưa ra những câu trả lời thích hợp. Mondly Chatbot đã hiểu được hơn hàng triệu cụm từ và từ vựng của nhiều ngôn ngữ và đang cải thiện từng ngày. [2]

## 3. Mục tiêu đề tài

* Đề xuất xây dựng ChatBot trên máy tính có thể trả lời tự động cho sinh viên trường Đại học Cần Thơ các câu hỏi liên quan đến các điều kiện xét và công nhận tốt nghiệp trong quy chế học vụ

## 4. Đối tượng và phạm vi nghiên cứu

* Đối tượng nghiêm cứu: sinh viên
* Phạm vi nghiên cứu: trường Đại học Cần Thơ

## 5. Phương pháp nghiên cứu

Dựa trên kết quả tìm hiểu về các bộ phân lớp máy học véc-tơ hỗ trợ, mạng nơ-ron, rừng ngẫu nhiên, k láng giềng và tập câu hỏi – câu trả lời dùng làm tập dữ liệu, tôi chọn bộ phân lớp văn bản máy học k láng giềng để xây dựng Chatbot. Khi xét về phương diện độ phức tạp giải thuật, k láng giềng là giải thuật có thể triển khai đơn giản nhất và giải thuật k láng giềng không có bước huấn luyện dữ liệu nên tốc độ dự đoán cũng nhanh hơn. Hạn chế của k láng giềng là độ chính xác khi áp dụng trên tập dữ liệu lớn. Tuy nhiên, xét trên tập dự liệu nhỏ như tập câu hỏi – câu trả lời về điều kiện xét và công nhận tốt nghiệp trong quy chế học vụ trường Đại học Cần Thơ, độ chính xác của giải thuật k láng giềng tương đối cao.

|  |  |  |  |
| --- | --- | --- | --- |
| Technique | Sức mạnh | Độ phức tạp giải thuật | Tốc độ huấn luyện dữ liệu |
| Máy học véc-tơ hỗ trợ | Cao | Rất khó | Rất chậm |
| Mạng nơ-ron | Rất cao | Rất khó | Rất chậm |
| Rừng ngẫu nhiên | Cao | Khó | Rất chậm |
| K láng giềng | Trung bình | Dễ | Nhanh |

## 6. Bố cục luận văn

**Phần giới thiệu**

Giới thiệu tổng quát về đề tài.

**Phần nội dung**

**Chương 1** : Mô tả bài toán

**Chương 2** : Thiết kế, cài đặt giải thuật, biễu diễn cơ sở dữ liệu, trình bày các bước xây dựng hệ thống

**Chương 3** : Kiểm thử hệ thống và đánh giá độ chính xác, tốc độ của hệ thống.

**Phần kết luận**

Trình bày kết quả đạt được và hướng phát triển hệ thống.

# PHẦN NỘI DUNG

# CHƯƠNG 1

# MÔ TẢ BÀI TOÁN

## 1. Mô tả chi tiết bài toán

Phần quan trọng nhất của Chatbot là tạo tập dữ liệu các câu hỏi để xác định câu hỏi đầu vào thuộc nhãn nào. Từ nhãn đã dự đoán, Chatbot tìm kiếm câu trả lời trong tập dữ liệu chứa câu trả lời. Để rút ngắn và tăng tính chính xác của quá trình xác định nhãn câu hỏi, tập dữ liệu câu hỏi cần loại bỏ những từ không mang hàm nghĩa chính trong câu như ‘vậy’, ‘thì’, etc. Sau khi đã lượt bỏ các từ không cần thiết, từng câu hỏi trong tập dữ liệu được chuyển thành những vector đa chiều. Để Chatbot có trả lời tự động quy chế học vụ - điều kiện xét và công nhận tốt nghiệp, Chatbot cần nghe được và lưu trữ câu hỏi của sinh viên. Tiếp đó, câu hỏi dạng giọng nói được chuyển về dạng văn bản để tiện việc xử lí về sau. Cũng như với tập dữ liệu câu hỏi, câu hỏi của sinh viên được lượt bỏ các từ không mang nghĩa chính trong câu và được chuyển thành vector. Để xác định nhãn câu hỏi, tôi sử dụng giải thuật K láng giềng để tìm các vector trong tập dữ liệu gần giống với vector câu hỏi đầu vào nhất để xác định nhãn. Sau khi đã dự đoán nhãn câu hỏi, tương ứng với từng nhãn là câu trả lời cho nhãn đó. Tiếp đó, câu trả lời được chuyển sang dạng giọng nói và phát ra cho sinh viên nghe.

## 2. Vấn đề và giải pháp liên quan đến bài toán

1. **Vấn đề**

### Các công việc chính để xây dựng Chatbot bao gồm:

* Lưu câu hỏi của sinh viên dưới dạng giọng nói
* Chuyển câu hỏi của sinh viên thành văn bản để xử lí
* Truy xuất tập dữ liệu câu hỏi và tập dữ liệu câu trả lời
* Tiền xử lí tập dữ liệu và vector hóa tập dữ liệu
* Áp dụng giải thuật K láng giềng để xác định nhãn câu hỏi của sinh viên
* Tìm câu trả lời theo nhãn và chuyển thành giọng nói

1. **Giải pháp**

- Để ghi âm câu hỏi của sinh viên, tôi sử dụng Pyaudio – thư viện Python cho phép kết nối đến microphone của mấy tính và lưu âm thanh được nói vào microphone.

- Để chuyển giọng nói thành văn bản, tôi sử dụng Speech Recognition python library, thư viện này chứa các hàm cho phép kết nối đến Google voice API để xử lí âm thành và chuyển âm thanh thành văn bản.

- Thư viện Pandas của python là một thư viện mạnh mẽ hỗ trợ thao tác dữ liệu. Đây cũng là bộ công cụ phân tích và xử lí mạnh mẽ python.

- Thư viện scikit-learn của python là thư viện về máy học chứa hầu hết các giải thuật hiện đại và luôn được cập nhật. Giải thuật K láng giềng củng được hỗ trợ trong thư viện này. Ngoài ra, sklearn cũng cung cấp các công cụ cần thiết để vector hóa văn bản.

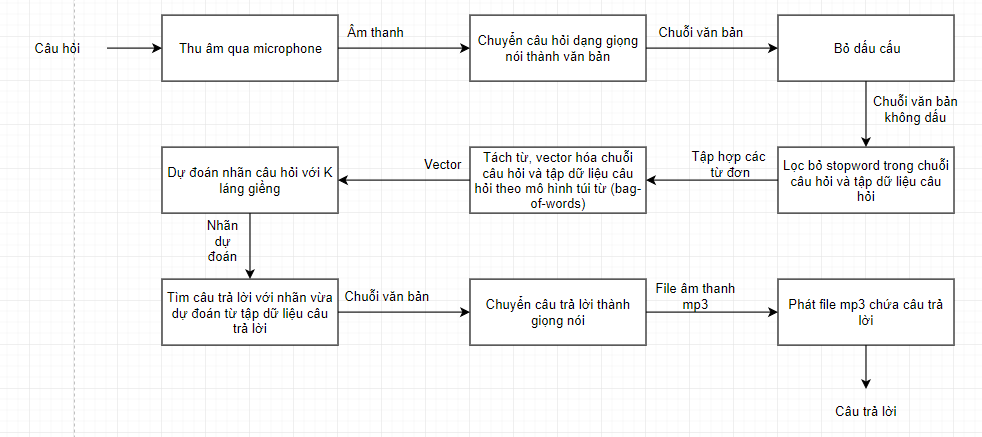
- Cuối cùng, để chuyển câu trả lời thành giọng nói, tôi sử dụng OpenFPT API - với khả năng chuyển đổi văn bản thành giọng nói tiếng việt với giọng đọc tự nhiên như con người.

# CHƯƠNG 2

# THIẾT KẾ VÀ CÀI ĐẶT

## 1. Thiết kế hệ thống

Mô hình hoạt động của Chatbot trên máy tính trả lời tự dộng cho sinh viên Cần Thơ các câu hỏi liên quan đến điều kiện xét và công nhận tốt nghiệp được trình bày trong hình bên dưới



Sinh viên có thể đặt câu hỏi cho Chatbot bằng cách nói trực tiếp vào microphone, Chatbot tiếp nhận và sử dụng Google API voice (voice-to-text) để chuyển đổi câu hỏi từ dạng tiếng nói sang dạng văn bản. Sau khi tiền xử lí và vector hóa dữ liệu, chuyển sang bộ nhận dạng câu hỏi bằng mô hình máy học để xác định câu trả lời bằng văn bản. Sau đó, sử dụng FPT API voice (text-to-speech) để phát ra câu trả lời bằng tiếng nói cho sinh viên nghe.

## 2. Xây dựng hệ thống

1. Tập dữ liệu

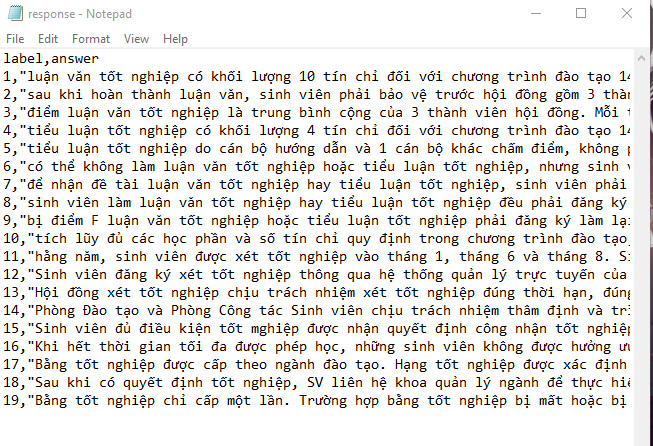
Tập dữ liệu bao gồm 2 file:

* dataset.csv : chứa câu hỏi và nhãn câu hỏi
* response.csv: chứa câu trả lời cho từng nhãn câu hỏi tương ứng

Dataset.csv: bao gồm 190 câu hỏi cho 19 nhãn



Response.csv: bao gồm 19 câu trả lời cho 19 nhãn tương ứng



Bởi vì Excel không thể hiển thị các kí tự có dấu trong tiếng việt nên khi mở file response.csv, các kí tự sẽ bị mã hóa. Tuy nhiên, ta vẫn có thể đọc được file thông qua notepad hoặc notepad++

1. Yêu cầu hệ thống

Trước khi bắt đầu thực hiện xây dựng Chatbot, máy tính cần cài đặt các thư viện python cần thiết:

* Speech Recognition: chuyển đổi giọng nói thành văn bản
* Pyaudio: thu âm giọng nói thông qua microphone
* Pyvi: thư viện xử lí ngôn ngữ tự nhiên tiếng việt
* Pandas: phân tích dữ liệu trong các bài toán máy học
* Requests: gửi yêu cầu đến server FPT để tạo file mp3 đã chuyển đổi từ văn bản
* Urllib.request: tải file mp3 đã chuyển đổi về máy
* Pygame: chạy file âm thanh ngầm dưới nền hệ thống
* scikit-learn: hỗ trợ các giải thuật máy học

Tiến hành cài đặt các thư viện trên:

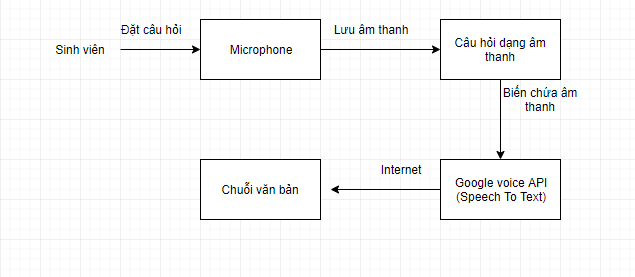
* Cài đặt thư viện Speech Recognition:
  + pip install SpeechRecogniton
* Cài đặt thư viện Pyaudio:
  + pip install pyaudio
  + pip install PyAudio-0.2.11-cp37-cp37m-win32.whl
* Cài đặt thư viện Pyvi:
  + pip install pyvi
* Cài đặt thư viện Pandas:
  + pip install pandas
* Cài đặt thư viện Requests:
  + Pip install requests
* Cài đặt thư viện Urllib.request:
  + pip install urllib.request
* Cài đặt scikit-learn
  + pip install -U scikit-learn

1. Thực hiện

**Bước 1: Ghi âm và chuyển đổi câu hỏi dạng giọng nói thành văn bản**

Trước tiên, sau khi sinh viên đọc câu hỏi thông qua microphone trên máy tính. Câu hỏi dạng âm thanh được lưu vào một biến tự định nghĩa nhờ sử dụng thư viện Pyaudio.

Kế đến, kết nối đến Google voice API thông qua thư viện Speech Recognition để chuyển giọng nói thành văn bản. Do phải gửi yêu cầu chuyển âm thanh thành văn bản nên cần kết nối Internet để thực hiện.



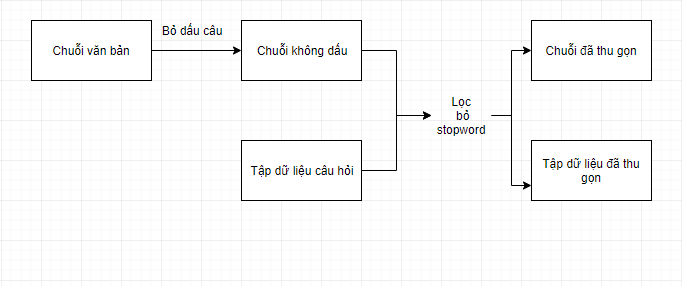
**Bước 2: Tiền xử lí câu hỏi dạng vă bản và tập dữ liệu**

Trong quá trình đọc câu hỏi vào microphone, có nhiều yếu tố gây ảnh hưởng và dẫn đến sai lệch khi ghi âm câu hỏi của sinh viên như: tiếng ồn, phát âm, cường độ âm, etc. Điều này dẫn đến việc dự đoán nhãn câu hỏi trở nên khó khăn hơn.

Ví dụ: Luận văn bằng bao nhiêu tín chỉ => luận ván bắng bao nhiêu tín chí

Do đó, Để tránh trường hợp như trên, loại bỏ dấu câu trong câu hỏi văn bản là vấn đề quan trọng và có ảnh hưởng tính chính xác của Chatbot.

Ngoài ra, trong bài toán phân loại, không phải mọi từ có trong câu đều được xem xét đến. Thông thường, những từ mang ý nghĩa chính trong câu, có thể dùng để phân loại nhãn câu hỏi được giữ lại. Những từ thừa còn lại được gọi là stopword( những từ không liên quan đến nội dung hay ý nghĩa trong câu) sẽ bị loại bỏ. Quá trình này không chỉ giúp làm giảm khối lượng dữ liệu đáng kể, mà còn tăng độ chính xác cũng như tốc độ phân loại câu hỏi.



**Bước 3: Tách từ, vector hóa từ theo mô hình túi từ (bag-of-words)**

K láng giềng là một thuật toán đơn giản có khả năng thực hiện các bài toán phân loại và hồi quy. Trong đó, dữ liệu đầu vào của thuật toán này luôn là dữ liệu số. Vì thế, để áp dụng được thuật toán K láng giềng trong bài toán phân loại văn bản, dữ liệu đầu vào phải trải qua bước chuyển đổi trước khi đưa vào thuật toán K láng giềng. Áp dụng mô hình túi từ (bag-of-words – BoW [Salton et at., 1975]) có thể thực hiện được việc này.

Mô hình túi từ (bag-of-words) là một biểu diễn đơn giản hóa được sử dụng trong xử lý ngôn ngữ tự nhiên và truy vấn thông tin (IR). Trong mô hình này, một văn bản (chẳng hạn như một câu hoặc một tập dữ liệu) được thể hiện dưới dạng túi (multiset) chứa các từ của nó, không quan tâm đến ngữ pháp và thậm chí trật tự từ nhưng vẫn giữ tính đa dạng. Mô hình túi từ cũng đã được sử dụng cho thị giác máy tính.

Ví dụ:

Số lượng tín chỉ của luận văn

Thời gian xét tốt nghiệp

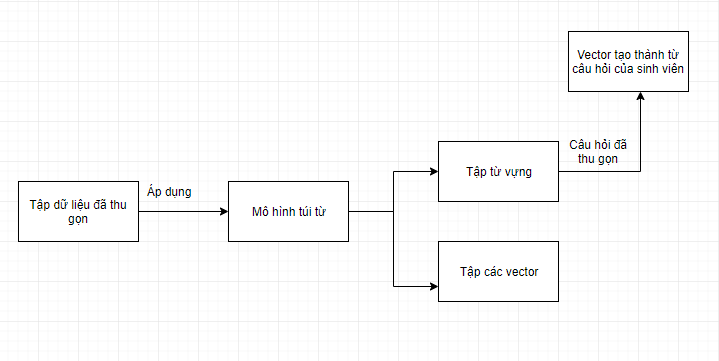
Từ 2 câu trên, tập từ vựng sẽ là {số, thời, lượng, gian, tín, xét, chỉ, tốt, của, luận, nghiệp, văn}

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | số | thời | lượng | gian | tín | xét | chỉ | tốt | của | luận | nghiệp | văn |
| Số lượng tín chỉ của luận văn | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 1 |
| Thời gian xét tốt nghiệp | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 |

Như vậy, 2 câu trên đã được chuyển đổi thành vector

Số lượng tín chỉ của luận văn = [ 1, 0, 1, 0, 1, 0, 1, 0, 1, 1, 0, 1 ]

Thời gian xét tốt nghiệp = [ 0, 1, 0, 1, 0, 1, 0, 1, 0, 0, 1, 0 ]

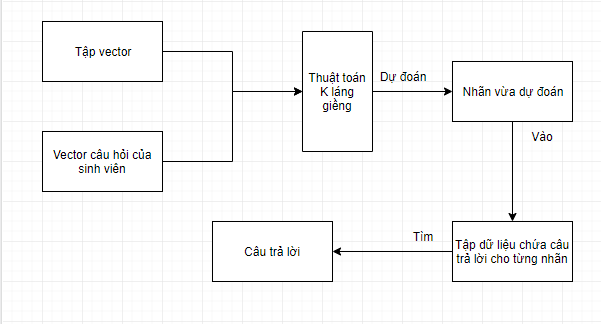


**Bước 4: Dự đoán nhãn câu hỏi với thuật toán K láng giềng và tìm câu trả lời**

K láng giềng (K-nearest neighbors – KNN ) là một trong những thuật toán supervised-learning đơn giản nhất (mà hiệu quả trong một vài trường hợp) trong Machine Learning. Khi training, thuật toán này không học một điều gì từ dữ liệu training (đây cũng là lý do thuật toán này được xếp vào loại lazy learning), mọi tính toán được thực hiện khi nó cần dự đoán kết quả của dữ liệu mới. K-nearest neighbor có thể áp dụng được vào cả hai loại của bài toán Supervised learning là phân loại và hồi quy. KNN còn được gọi là một thuật toán Instance-based hay Memory-based learning.

Với KNN, trong bài toán phân loại, nhãn của một điểm dữ liệu mới được suy ra trực tiếp từ K điểm dữ liệu gần nhất trong tập training. Nhãn của một điểm dữ liệu cần xét có thể được quyết định bằng major voting (bầu chọn theo số phiếu – các số phiếu có giá trị ngang nhau) giữa các điểm gần nhất. Hoặc đánh trọng số cho K láng giềng gần nhất với khoảng cách tỉ lệ nghịch với trọng số rồi suy ra nhãn dựa trên tổng trọng số nhãn của từng loại nhãn từ K láng giềng gần nhất.

Với tập vector được sử dụng làm tập training cho KNN, bắt đầu thực hiện gán nhãn cho vector tạo thành từ câu hỏi của sunh viên. Sau khi dự đoán, KNN trả về giá trị nhãn dự đoán được. Từ nhãn này, tiến hành tìm câu trả lời trong tập dữ liệu chứa câu trả lời bằng cách dùng các hàm trong thư viện Pandas.

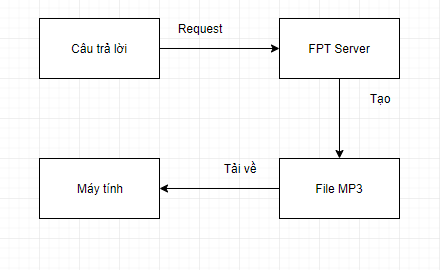


**Bước 5: Chuyển câu trả lời thành giọng nói thông qua FPT.AI Speech**

FPT.AI Speech là công nghệ chuyển văn bản tiếng Việt thành giọng nói (Text-to-speech) có thể biến các dòng chữ vô hồn thành giọng đọc giống như người thật. Từ đầu năm 2019, FPT.AI Speech cập nhật thêm 3 phiên bản giọng nói tiếng Việt mới cho chất lượng giọng đọc tự nhiên giống như con người.

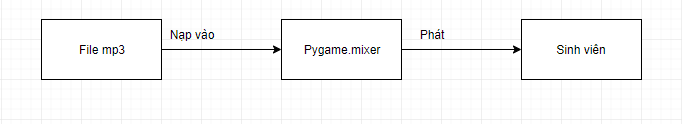
Theo Techinsight, để cho ra đời các phiên bản giọng đọc “thật” nhất, các chuyên gia và kỹ sư của Ban Công nghệ FPT (FTI) đã nghiên cứu một thời gian dài, thu thập, phân tích và xử lý hàng triệu dữ liệu, hàng nghìn giọng đọc nam và nữ từ cả ba miền Bắc, Trung, Nam. Đồng thời phát triển và ứng dụng Ngôn ngữ học, Âm học và các công nghệ: Xử lý ngôn ngữ tự nhiên (Natural Language Processing), học máy (Machine Learning), xử lý tín hiệu số (Digital Signal Processing).

Để chuyển câu trả lời thành giọng nói, đầu tiên thực hiện gửi gói tin request đến FPT server chứa các thông tin như giọng đọc, nội dung, tốc độ đọc. Sau khi chờ FPT server tạo file mp3 chứa nội dụng vừa gửi trong gói tin đã được chuyển đổi thành giọng nó, thực hiện tải file này về máy tính bằng cách sử dụng thư viện Urllib.request của Python.



**Bước 6: Phát file âm thanh chứa câu trả lời cho sinh viên nghe**

Sử dụng thư viện Pygame của python để thực hiện phát file âm thanh cho sinh viên nghe dưới nền hệ thống.



# CHƯƠNG 3

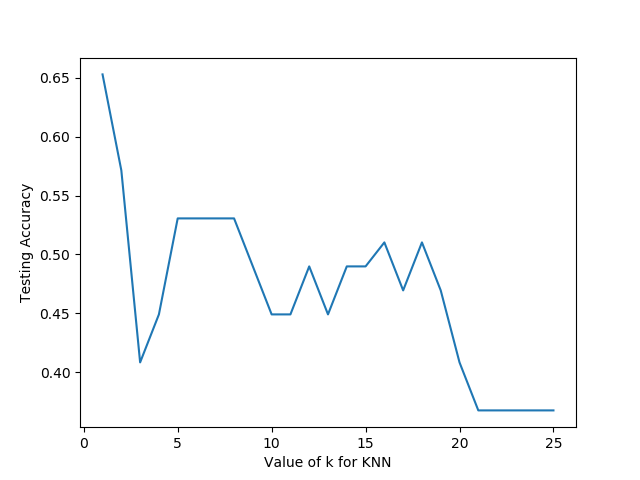
# KIỂM THỬ VÀ ĐÁNH GIÁ

## Để tiến hành thực nghiệm trên tập dữ liệu gồm 190 câu hỏi cho 19 câu trả lời, tôi đã tiến hành chia tập dữ liệu thành tập test và tập training theo tỉ lệ tương ứng là 1:5. Đồng thời, trong thuật toán K láng giềng, có 3 tham số có thể ảnh hưởng đến kết quả trong quá trình thực nghiệm, đó là:

* Algorithm: là thuật toán được dùng để tính toán khoảng cách đến K láng giềng gần nhất
* P: công thức tính toán khoảng cách
* Weight: trọng số của các láng giềng

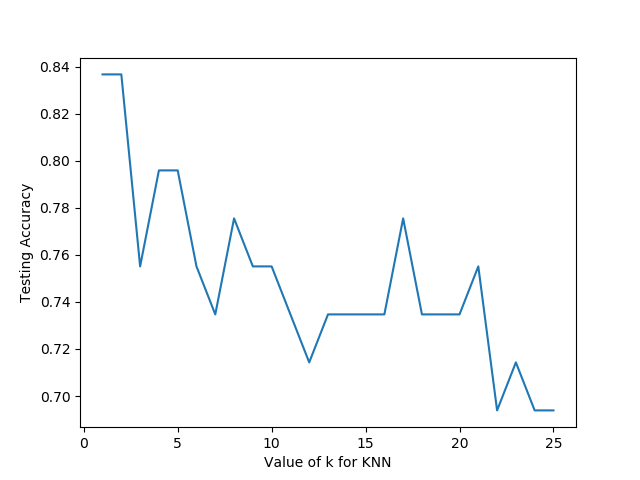
Ngoài ra, số lượng K láng giềng cần xét để dự đoán nhãn cho điểm dữ liệu mới cũng rất quan trọng. Hình bên dưới thể hiện kết quả đo đạc từ thuật toán K láng giền với các tham số ở giá trị mặc định và K trong đoạn [1, 25]

* Algorithm: auto – tự động chọn thuật toán tính khoảng cách thích hợp
* P = 1 – sử dụng khoảng cách manhattan (manhattan distance)
* Weight: uniform – mọi điểm dữ liệu có trọng số như nhau bất kể khoảng cách đến điểm cần xác định nhãn.



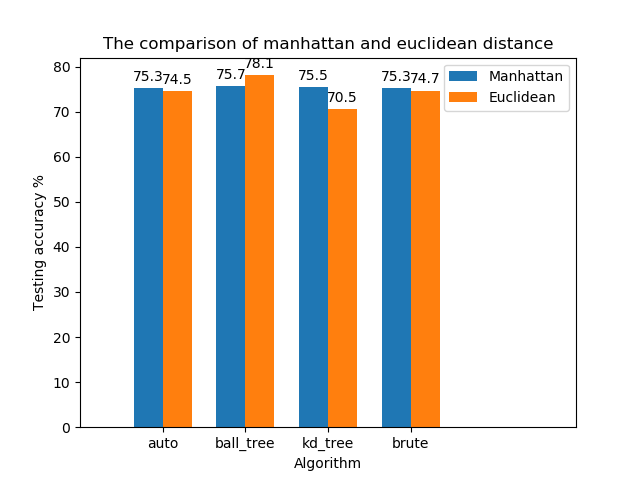
## Từ hình trên, độ chính xác dự đoán cao nhất chỉ ở mức xấp xỉ 65% (với K = 1). Để tăng tính chính xác của thuật toán K láng giềng, thông thường mỗi điểm dữ liệu đều được đánh trọng số. Theo đó, điểm dữ liệu càng gần điểm cần xét nhãn thì trọng số(độ tin cậy) càng cao. Ngược lại, những điểm càng xa, trọng số(độ tin cậy) càng thấp đi. Do đó, tôi thử thay đổi tham số weight

* Algorithm : auto
* P = 1 - khoảng cách manhattan
* Weight = distance (đánh trọng số cho các điểm dữ liệu)



## Kết quả, độ chính xác dự đoán cao nhất tăng lên đến gần 84% (K = 2). Ngoài ra, ta có thể thay đổi các tham số algorithm (auto, ball\_tree, kd\_tree, brute) và P (manhattan distance, euclidean distance). Kết quả tính toán được thể hiện như hình bên dưới

## 



Hình trên là kết quả so sánh độ chính xác của 2 công thức tính khoảng cách là Manhattan Distance và Euclidean Distance đối với từng thuật toán dùng để tính toán khoản cách đến K láng giềng gần nhất – auto, ball\_tree, kd\_tree, brute. Theo đó, phương pháp sử dụng thuật toán K láng giềng kết hợp với thuật toán ball\_tree tính toán khoảng cách đến các láng giềng theo công thức Euclidean Distance đạt độ chính xác cao nhất (78.1 %).

# PHẦN KẾT LUẬN

Phía trên là phương pháp xây dựng Chatbot trên máy tính trả lời tự động cho sinh viên các câu hỏi liên quan đến điều kiện xét và công nhận tốt nghiệp. Tập dữ liệu cho Chatbot đã biên soạn gồm 190 câu hỏi cho 19 câu trả lời từ quy chế học vụ - xét và công nhận tốt nghiệp. Sau khi được huấn luyện bằng giải thuật K láng giềng để phân lớp tự động câu hỏi, Chatbot cho độ chính xác là 78%.

Trong tương lai, tôi dự định cải thiện tốc độ phản hồi của Chatbot khi trả lời thắc mắc của sinh viên. Tiếp theo là mở rộng Chatbot cho các vấn đề khác trong quy chế học vụ như cách đánh giá kết quả học tập hoặc phương pháp tổ chức đào tạo.

# TÀI LIỆU THAM KHẢO

1. Đỗ Thanh Nghị, Hoàng Tùng (2019), Chatbot cho sinh viên công nghệ thông tin, kỷ yếu Hội nghị Quốc gia lần thứ XII về nghiêm cứu cơ bản và ứng dụng Công nghệ thông tin (FAIR), Huế.
2. Alexandru Iliescu (2016), Mondly launches first voice chatbot for learning languages,<<https://www.mondly.com/blog/2016/08/25/mondly-chatbot-press-release/>>, [online: October 10, 2019]
3. James D.McCaffrey, A comparison of Ten Machine Learning Classification Algorithms, <<https://jamesmccaffrey.wordpress.com/2018/11/07/a-comparison-of-ten-machine-learning-classification-algorithms/>> [online: November 2, 2019]
4. Chathuranga Widanapathirana, Ahmet Sekercioglu, Paul Fitzpatrick, Milosh Ivanovich, Jonathan C.li, Automated Inference System for End-To-End Diagnosis of Network Performance Issues in Client-Terminal Devices, <<https://www.researchgate.net/publication/229090754_Automated_Inference_System_for_End-To-End_Diagnosis_of_Network_Performance_Issues_in_Client-Terminal_Devices>>, [online: November 2, 2019]
5. Open FPT, <<https://docs.openfpt.vn/?python#openfpt-api>>, [online: October 14, 2019]
6. sklearn.neighbors.KneighborsClassifier, <<https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.KNeighborsClassifier.html>>, [online: November 4, 2019]
7. KNN Classification using Scikit-learn, <<https://www.datacamp.com/community/tutorials/k-nearest-neighbor-classification-scikit-learn>>, [online: November 4, 2019]
8. A complete guide for beginning with K-nearest neighbours algorithm in python, <<https://analyticsindiamag.com/a-complete-guide-for-beginning-with-k-nearest-neighbours-algorithm-in-python/>>, [online: November 4, 2019]
9. Kaggle (2016), Text classification using bag of word features , <<https://www.kaggle.com/aneeshc/text-classification-using-bag-of-word-features/data>>, [online: November 4, 2019]
10. Ravi Shankar (2017), Measuring Text Similarity in Python <<https://www.linkedin.com/pulse/measuring-text-similarity-python-ravi-shankar>>, [online: November 4, 2019]
11. Machine Learning cơ bản, <<https://machinelearningcoban.com/2017/01/08/knn/>> [online: November 4, 2019]
12. Tân phong(2019), Giọng mới của FPT.AI, <<https://chungta.vn/cong-nghe/giong-moi-cua-fpt-ai-chat-hon-chi-google-1123692.html>> [online: November 4, 2019]