HỆ THỐNG TRẢ LỜI TỰ ĐỘNG ỨNG DỤNG KỸ THUẬT HỌC MÁY

AUTOMATIC ANSWERING SYSTEM APPLIES MACHINE LEARNING TECHNIQUES

Đậu Thị Huyền[1], Vũ Xuân Thành[1], Nguyễn Thị Hương[1], Lê Đức Duy[1], Khuất Thu Trang[1] , Vũ Thị Tuyết Mai[2,*]

TÓM TẮT

Bài báo trình bày cách xây dựng một hệ thống trả lời tự động ứng dụng kỹ thuật học máy. Từ các phương pháp học có giám sát của học máy áp dụng một phương pháp học có giám sát phù hợp đó là mạng noron vào xây dựng hệ thống trả lời tự động. Khảo sát bài toán, thu thập dữ liệu, xử lý dữ liệu, xây dựng model, huấn luyện mô hình và test dữ liệu là những việc cần làm trong quá trình xây dựng một hệ thống chatbot. Cụ thể là hệ thống trả lời tự động của khoa Công nghệ thông tin Trường Đại học Công nghiệp Hà Nội.

Từ khóa: *Trả lời tự động, mạng nơron, hệ thống.*

ABSTRACT

This paper presents how to build an automated answering system that applies machine learning techniques. From the supervised learning methods of machine learning apply a suitable supervised learning method that is the noron network into the construction of the automatic answering system. Surveying problems, collecting data, processing data, building models, training models, and testing data are things to do in the process of building a chatbot system. Specifically, the automatic answering system of the Faculty of Information Technology of Hanoi University of Industry.

Keywords: *Automatic replies, neural networks, system.*

[1]Lớp ĐH Công nghệ thông tin 06 - K13, Khoa Công nghệ thông tin, Trường Đại học Công nghiệp Hà Nội

[2]Khoa Công nghệ thông tin, Trường Đại học Công nghiệp Hà Nội

[*]Email: maivtt_fit@haui.edu.vn

1. MỞ ĐẦU

Trong những năm trở lại đây, trí tuệ nhân tạo đang dẫn đầu sự phát triển công nghệ thông tin của cuộc cách mạng công nghiệp 4.0. Thuật ngữ AI (Artificial Intelligence) ngày càng trở nên phổ biến không chỉ trong ngành công nghệ thông tin mà còn cả những lĩnh vực, ngành nghề khác. Sự bùng nổ của trí tuệ nhân tạo trong giai đoạn hiện nay thể hiện qua những chuyển biến rõ rệt. Trước hết xu hướng ứng dụng công nghệ AI, Machine Learning (học máy) tiếp tục được phát triển mạnh mẽ như vũ bão tại các doanh nghiệp cho các ngành nghề khác nhau như tài chính ngân hàng, du lịch, y tế và giáo dục. Cùng với các công nghệ như xử lí ảnh, nhận dạng khuôn mặt,… thì sự phát triển của các trợ lí ảo tự động đã có một vị thế lớn trong thế giới của ML. Học máy đã giúp máy tính thực hiện được những công việc mà máy tính tưởng như không thể như phân tích dữ liệu trong hàng triệu bức ảnh, bắt chước giọng nói và chữ viết của con người đồng thời trả lời tự động với con người. Thậm chí ngay cả sáng tác nhạc, thơ hay truyện.

Nhóm tác giả sử dụng các phương pháp để tiến hành nghiên cứu như: khảo sát hiện trạng, phân tích hệ thống, nghiên cứu các phương pháp học có giám sát của học máy và lựa chọn một phương pháp phù hợp để áp dụng vào bài toán, sử dụng công cụ lập trình và ngôn ngữ lập trình Python để cài đặt chương trình.

Nhóm nghiên cứu dựa trên các đối tượng trung tâm là sinh viên và các cố vấn học tập của khoa Công nghệ thông tin, trường Đại học Công nghiệp Hà Nội. Phạm vi nghiên cứu là trong khoa Công nghệ thông tin, trường Đại học Công nghiệp Hà Nội.

2. ĐỐI TƯỢNG VÀ PHƯƠNG PHÁP NGHIÊN CỨU

Chatbot trong lĩnh vực học máy

Trong lĩnh vực học máy, Chatbot hay các cô nàng trợ lý ảo đều được quy chung về một loại tài liệu đó là Question and Answwering system. Đối với các hệ chuyên gia như vậy, công việc cần làm:

Phân loại câu hỏi

Mapping câu trả lời (Trích chọn tài liệu liên quan)

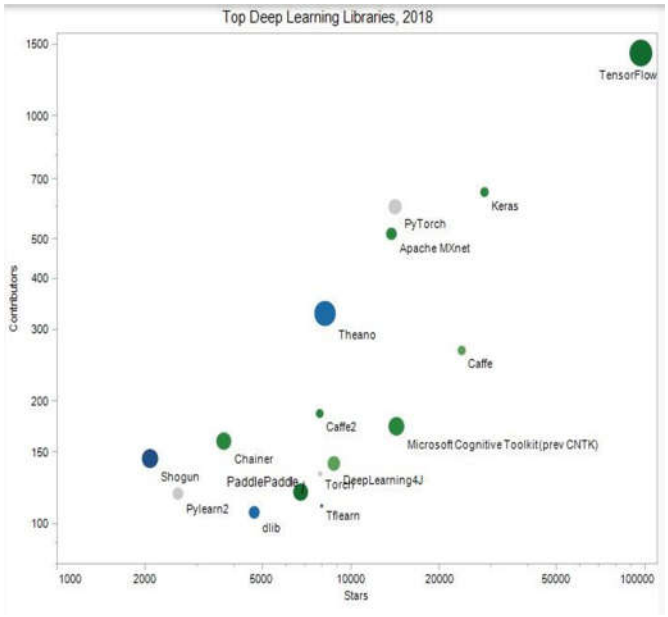
Trích xuất câu trả lời

Việc phân loại câu hỏi là bước khó nhất trong hệ thống hỏi đáp. Tuy nhiên, như chính cái tên của nó, bạn có thể chuyển bài toán này về các bài toán phân lớp đã biết đến. Đến đây, có lẽ bạn đã hình dung ra chatbot không hề khó như những gì bạn biết phải không nào?

Giới thiệu về Keras

Kể từ năm 2012 khi Deep learning có những bước đột phá lớn, hàng loạt framework hỗ trợ Deep learning ra đời. Các thư viện deep learning thường được 'chống lưng' bởi những hãng công nghệ lớn: Google (Keras, TensorFlow), Facebook (Caffe2, Pytorch), Microsoft (CNTK), Amazon (Mxnet), Microsoft và Amazon cũng đang bắt tay xây dựng Gluon (phiên bản tương tự như Keras).

Sau đây là một vài thống kê để mọi người có cái nhìn tổng quan về các thư viện hay được sử dụng nhất:



Những so sánh gần đây chỉ ra rằng Tensorflow, Keras and Caffe là các framework được sử dụng nhiều nhất. Tuy nhiên, Keras có cú pháp đơn giản, dễ sử dụng hơn Tensorflow rất nhiều. Keras được coi là một "high level" với phần "low level (còn được gọi là backend) có thể là Tensorflow, CNTK, hoặc Theano.. Với API bậc cao, dễ sử dụng, keras giúp người dùng xây dựng các deep learning model một các đơn giản. Vì vậy phần này chúng tôi sẽ sử dụng Keras.

Bạn có dữ liệu, bạn muốn máy tính học được các mô hình model từ dữ liệu, sau đó dùng mô hình đấy để dự đoán được các dữ liệu mới. Các bước cơ bản huấn luyện một mô hình neural network trong keras gồm các bước:

- Xây dựng bài toán
- Chuẩn bị dữ liệu (dataset)
- Xây dựng model (network)
- Huấn luyện mô hình
- Đánh giá mô hình

Xây dựng bài toán

Đề tài bài toán là xây dựng hệ thống trả lời tự động ứng dụng học máy nên Input của bài toán sẽ là câu hỏi dạng text.Và Ouput chúng ta cần tìm là câu hỏi input đó nằm trong bộ câu hỏi nào trong cơ sở dữ liệu. Từ đó sẽ lấy ra response để trả lời cho người dùng.

Tuy nhiên, máy không thể đọc được ngôn ngữ dưới dạng chữ (text) mà chúng ta phải mã hóa nó về dạng số. Vì vậy, trước tiên chúng ta cần phải tiền xử lý ngôn ngữ tự nhiên, mã hóa chúng về dạng túi từ (vecto) để phù hợp cho việc trainning.

Chuẩn bị dữ liệu

Một bộ dữ liệu training đủ nhiều và chuẩn chỉnh có thể nâng cao độ chính xác lên rất nhiều. Nhóm nghiên cứu định nghĩa cấu trúc dữ liệu mà chúng tôi thu thập được theo dạng json dưới đây:

```
{"intents": [
        {
            "tag": "greeting",
            "patterns": [
                "xin chào",
                "chào bạn",
                "có ai ở đây",
                "chào",
                "Hey",
                "hi",
                "Hello"
            ],
            "responses": [
                "chào",
                "chào bạn, mình có thể giúp gì nhỉ?",
                "chào bạn",
                "hello",
                "hi"
            ],
            "context": [
                ""
            ]
        },
```

Trong đó:

- tag là nhãn lớp cho nội dung nhập của người dùng
- patterns - mẫu câu đầu vào được trainning để phân lớp
- responses - các câu trả lời được mapping (trích chọn) để hồi đáp những câu hỏi trước đó của người dùng.

Xử lý ngôn ngữ tự nhiên

Đầu tiên, tạo một túi từ chứa tất cả các từ có thể có trong tất cả câu hỏi trong Data trainning. Những từ đó có thể là từ riêng lẻ hoặc từ có nghĩa được ghép lại với nhau. Túi từ được đánh số theo thứ tự từ điển, tuy nhiên đã được loại bỏ những ký tự không có nghĩa như là dấu chấm, dấu hỏi,…

Giả sử túi từ của chúng ta có dạng như sau:

xin	chào	bạn	có	ai	ở	đây	không	.	.	.	.	.

Bây giờ chúng ta chuyển các mẫu câu hỏi trong data về dạng túi từ. Ví dụ:

túi từ	xin	chào	bạn	có	ai	ở	đây	không	.	.	.	.	.
xin chào	1	1	0	0	0	0	0	0	0	0	0	0	0
chào bạn	0	1	1	0	0	0	0	0	0	0	0	0	0
…..													

Và như thế, tất cả mẫu câu hỏi trong dữ liệu của chúng ta đã được chuyển thành số.

Tiếp theo, chúng ta đưa vào một câu hỏi, chúng ta phải xem xem câu hỏi đó thuộc mẫu câu hỏi nào trong các bộ data trainning mà chúng ta đã làm. Nhưng trước tiên, câu hỏi này cũng phải được biểu diễn thành dạng túi từ để so sánh. Quá trình so sánh câu hỏi đưa vào với những câu hỏi trong data thực chất là quá trình so sánh hai bộ vecto. Và so sánh như thế nào, thì chúng ta sẽ xây dựng mô hình để so sánh.

Xây dựng model

Vì input của model ở dạng text nên ta nghĩ ngay đến Neural Network (mạng noron nhân tạo). Mặc dù Keras có hỗ trợ sẵn cho chúng ta classification_model() cho các bài

toán phân loại classification nhưng trong bài này chúng tôi sẽ giới thiệu đến cách xây dựng model bằng Sequential Model.

Ở đây, ta sẽ sử dụng mô hình mạng noron truyền thẳng (chúng tôi đã giới thiệu ở chương 2) với cấu trúc 3 lớp để xây dựng model.

```
#khởi tạo mô hình #sequential model
model = Sequential()
model.add(Dense(128, input_shape=(len(train_x[0]),), activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(64, activation='relu')) # sử dụng hàm kích hoạt là relu và softmax
model.add(Dropout(0.5))
model.add(Dense(len(train_y[0]), activation='softmax'))
```

• Lớp thứ nhất là lớp input gồm 128 mạng noron, dữ liệu đầu vào của mạng noron chính bằng độ dài của câu hỏi trong data trainning hay nói một cách đơn giản input bằng kích thước của túi từ. Tuy nhiên, dữ liệu đầu vào của mình là dữ liệu thô (dạng text) nên trước khi đưa vào model thì phải tiền xử lý dữ liệu này sang dạng số (phần xử lý ngôn ngữ tự nhiên đã nói qua) để mạng noron có thể chạy. Ở bước này chúng tôi sử dụng hàm kích hoạt là relu.

• Lớp thứ hai là hidden layer gồm 64 noron với hàm kích hoạt là Relu (tương tự như lớp 1).

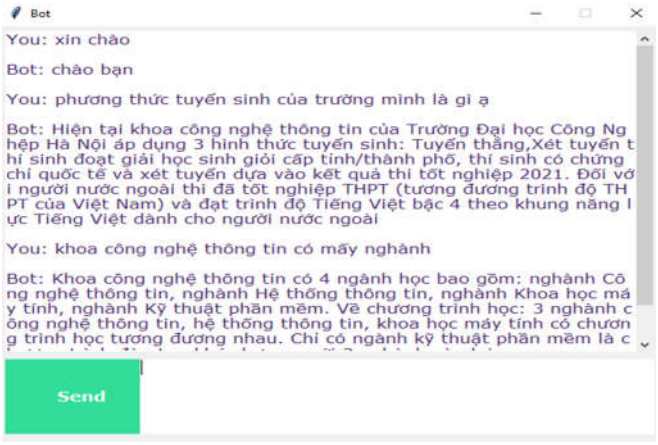
• Lớp thứ 3 là lớp output với đầu ra là số classes (số "tag" trong bộ dữ liệu mà chúng tôi đưa vào (file data.json nói trên). Mỗi đầu ra tương tứng với một số a% bất kỳ và tổng tất cả đầu ra là bằng 100%. Đến đây, chúng ta sẽ xét xem là đầu ra nào có số lượng % lớn nhất thì đầu vào của mình sẽ thuộc lớp đó. Và ở đây, chúng ta sử dụng hàm kích hoạt là softmax.

Trong mạng noron, ở mỗi layer sẽ thực hiện hai bước: tính tổng tuyến tính các node ở layer trước và thực hiện activation function (ví dụ sigmoid function, softmax function,…).

3. KẾT QUẢ VÀ THẢO LUẬN

Sau khi xây dựng được model, chúng ta tiến hành việc huấn luyện mô hình bằng các dữ liệu chúng ta đã chuẩn bị và xử lý sang dạng số. Khi đã huấn luyện xong, kiểm tra mô hình bằng cách đưa câu hỏi cần hỏi vào để test.

Kết quả đạt được sau khi chạy chương trình như sau:



4. KẾT LUẬN VÀ KIẾN NGHỊ

Thông qua quá trình nghiên cứu, nhóm tác giả đã tìm hiểu về học máy (Machine learning), phân loại học máy thành hai nhóm chính: học có giám sát và không có giám sát. Từ đó tìm hiểu được một số phương pháp học có giám sát trong Machine learning và ứng dụng một thuật toán cụ thể là Mạng noron truyền thẳng nhiều lớp để xây dựng hệ thống trả lời tự động.

TÀI LIỆU THAM KHẢO

[1]. Giáo trình *Trí tuệ nhân tạo,* Trường Đại học Công nghiệp Hà Nội

[2]. https://vi.wikipedia.org/wiki/M%E1%BA%A1ng_th%E1%BA%A7n_kinh_nh%C3%A2n_t%E1%BA%A1o – Mạng thần kinh nhân tạo – Wikipedia

[3]. https://chienuit.wordpress.com/2015/08/28/tim-hieu-ve-neural-network-perceptron/

[4]. https://dlapplications.github.io/2018-06-11-perceptron/

[5]. https://kipalog.com/posts/Neural-Network

[6]. https://dominhhai.github.io/vi/2018/04/nn-intro/#4-h%E1%BB%8Dc-v%E1%BB%9Bi-m%E1%BA%A1ng-nnBattle of the Deep Learning frameworks — Part I: 2017, even more frameworks and interfaces

[7]. Keras hompage

[8]. Comparison of deep learning software – Wikipedia

[9]. https://machinelearningcoban.com/2018/07/06/deeplearning/

[10]. https://www.codespeedy.com/the-sequential-model-in-keras-in-python