**Deploy a machine learning model using Reactjs and Tensorflow.JS**

# 1. Lý thuyết

## **a) ReactJS**

Là một opensource được phát triển bởi Facebook, ra mắt vào năm 2013, bản thân nó là một thư viện Javascript được dùng để để xây dựng các tương tác với các thành phần trên website. Một trong những điểm nổi bật nhất của ReactJS đó là việc render dữ liệu không chỉ thực hiện được trên tầng Server mà còn ở dưới Client nữa.

Bạn có thể học về Reactjs tại:

<https://www.taniarascia.com/getting-started-with-react>

<https://reactfordesigners.com>

<https://topdev.vn/blog/huong-dan-chi-tiet-xay-dung-web-app-voi-reactjs>

## **b) Tensorflow.JS**

TensorFlow.js là một Thư viện JavaScript để đào tạo (training) và triển khai (deploying) các mô hình (model) học máy trong trình duyệt (browser) và trong Node.js.

Tài liệu nên xem:

<https://toihocdesignpattern.com/tensorflow-js-la-gi.html>

<https://vntalking.com/machine-learning-voi-nodejs-ket-hop-tensorflow-js.html>

<https://www.geeksforgeeks.org/how-to-deploy-a-machine-learning-model-using-node-js>

LƯU Ý: Đây là tài liệu quan trọng và nên đọc: <https://www.tensorflow.org/js/tutorials>

# 2. Hướng đẫn nhúng một mô hình AI lên web sử dụng Reactjs và tensoflow

Trong bài này tôi sẽ dùng mô hình nhận diện ngôn ngữ cơ thể và sẽ nhận diện trực tiếp ngôn ngữ cơ thể trên Webcam

Video hướng dẫn chi tiết: <https://youtu.be/kXauXm0dBm>

File Code: <https://github.com/thangdomanh/Deploy_AI_Reactjs_TF>

**Yêu cầu:** tải file đã được cung cấp trước khi làm:

A screenshot of a computer

Description automatically generated with medium confidence

## **Bước 1:**

**Chuyển đổi mô hình TensorFlow từ file .h5 sang file .json (bước này dành cho người đã có kiến thức cơ bản về AI, train model. Nếu chưa hiểu thì bỏ qua bước này, bạn sẽ dùng file file JSON đã được convert để thực hiện ở bước 2)**

Ta sử dụng thư viện TensorFlow.js Converter (tensorflowjs\_converter):

Đầu tiên, tôi có một file Tutorial.ipynb:

A screenshot of a computer

Description automatically generated with medium confidence

Việc chuyển đổi khá đơn giản khi bạn chỉ cần thêm 2 phần như sau:

*Phần 1*

A screenshot of a computer

Description automatically generated with low confidence

Code:

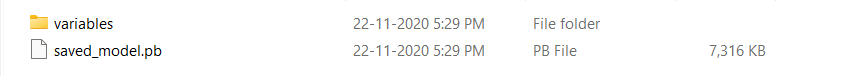
print("""python {}/research/object\_detection/exporter\_main\_v2.py --input\_type=image\_tensor --pipeline\_config\_path={}/{}/pipeline.config --trained\_checkpoint\_dir={} --output\_directory={}export""".format(APIMODEL\_PATH, MODEL\_PATH, CUSTOM\_MODEL\_NAME,CHECKPOINT\_PATH, CHECKPOINT\_PATH))

Sau khi bạn chạy đoạn code này sẽ cho ra ouput phía đưới, việc của bạn là copy chúng và chạy chúng trong CMP

Ta sẽ kiểm tra file và có được kết quả như sau:

A picture containing text, font, line, white

Description automatically generated



A picture containing text, font, screenshot

Description automatically generated

*Phần 2: Convert model*

A screenshot of a computer program

Description automatically generated with low confidence

Code:

!pip install tensorflowjs

------------------------------------------------------------------------------------------------------------

"""tensorflowjs\_converter --input\_format=tf\_saved\_model --output\_node\_names='detection\_boxes,detection\_classes,detection\_features,detection\_multiclass\_scores,detection\_scores,num\_detections,raw\_detection\_boxes,raw\_detection\_scores' --output\_format=tfjs\_graph\_model --signature\_name=serving\_default {}export/saved\_model {}converted""".format(CHECKPOINT\_PATH, CHECKPOINT\_PATH)

Kiểm tra kết quả:

A screenshot of a computer

Description automatically generated with low confidence

Như vậy ta đã convert thành công, quá trình train là khá khó, tôi sẽ không hướng dẫn phần train model mà chỉ hướng dẫn convert sang TFJS model

## **Bước 2:**

**Lưu trữ model đã convert lên đám mây**

Ở trường hợp này tôi sẽ sử dụng **github** để lưu trữ vì nó có các đặc điểm ưu việt sau đây:

* Miễn phí, tiết kiệm chi phí và không cần phải dùng các nền tảng như IBM, GG platform, …
* Nền tảng này đễ upload model lên mà không cần phải cấu hình nhiều
* Nền tảng này quen thuộc và có thể bạn còn chưa biết tính năng lưu trữ một model và Cho phép bạn truy cập trực tiếp vào nội dung của tệp tin.

Đầu tiên, ta có file model đã convert (tôi đã convert sẵn, chỉ cần tải về và làm theo hướng dẫn):

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with low confidence

Ta đăng nhập và truy cập vào trang github của bản thân và tạo một “new repository”:

A screenshot of a computer

Description automatically generated

Các bạn đặt tên, chọn public và chọn create repository:

A screenshot of a computer

Description automatically generated with medium confidence

Sau khi tạo các bạn sẽ có một repository, truy cập đến trang repository và kéo thả file chứa model TFJS vào:

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated

Kết quả:

A screenshot of a computer

Description automatically generated

Như vậy đã lưu trữ thành công nhưng nếu dùng phương thức link bình thường thì ta sẽ không thể truy cập vào model này để truyền dữ liệu cũng như lấy dữ liệu trả về từ nó.

Bạn cần phải đổi đường liên kết như sau

Trước: <https://github.com/thangdomanh/Model_TFJS_LanguageDetection/blob/main/converted/model.json>

Sau: <https://raw.githubusercontent.com/thangdomanh/Model_TFJS_LanguageDetection/main/converted/model.json>

Chúng ta đã thay thế github.com bằng raw.githubusercontent.com và loại bỏ /blob trong đường dẫn. Điều này cho phép truy cập trực tiếp vào nội dung của tệp tin.

Nếu chưa hình dung được đoạn link này sẽ được dùng làm gì:



A picture containing screenshot, text, multimedia software, software

Description automatically generated

## **Bước 3:**

**Tiến hành code và chạy model trên web dùng Reactjs**

**Yêu cầu:**

Tôi sài VSCode xanh để thực hiện bài làm, tuy nhiên bạn vẫn có thể sử dụng các phần mềm khác để thực hiện.

Đảm bảo bạn đã cài đặt nodejs, nếu chưa cài đặt bạn có thể tham khảo truy cập link sau hoặc truy cập vào GG để tiến hành tải file về

* Windows: <https://openplanning.net/11921/cai-dat-nodejs-tren-windows>
* Linux: <https://tenten.vn/help/huong-dan-cai-dat-node-js-tren-ubuntu-20-04/>
* MAC: <https://thuthuatmaytinh.vn/thu-thuat-mac-os/cach-cai-dat-node-js-va-npm-tren-mac-os/>

Mở file tôi đã gửi để copy code cho nhanh.

Tôi sẽ nói lại về các mục được bôi đen bên dưới:

A screenshot of a computer

Description automatically generated with medium confidence

* File Tensorflow là file model đã được convert, file tutorial là file để train và convert model được dùng trong bước 2.
* File my-app là file dùng React để thực hiện việc chạy model trên web.

Sau đây tôi sẽ hướng dẫn các bạn thực hiện việc cài đặt react và nhúng model vào:

**Lưu ý:** trước hết bạn cần phải mở file Code và đọc nó, trong đó có phần giải thích code.

Đầu tiên bạn tao một thư mục trống (đặt tên theo quy tắc không dấu, không kí tự đặc biệt, không khoảng có Sapce, …).

A screenshot of a computer

Description automatically generated with medium confidence

Mở VSCode lên và mở Folder lên

*Cách khác:* Mở Terminal hoặc Command Prompt và điều hướng đến thư mục nơi bạn muốn tạo dự án của bạn. Tuy nhiên bạn **không nên dùng cách nà**y vì sẽ khó cho tôi trong các hướng dẫn tiếp theo.

A screenshot of a computer menu

Description automatically generated with medium confidence

Sau đó, chọn New Terminal

A screenshot of a computer program

Description automatically generated with medium confidence

Sau đó, gõ lệnh sau và chờ đợi:

npx create-react-app my-app //trong đó my-app là tên dự án bạn muốn đặt

A screenshot of a computer

Description automatically generated

Kết quả sau khi chạy lệnh:

A screenshot of a computer

Description automatically generated

**Lưu ý**: nhà phát triển cũng đã cho ta một vài hướng dẫn như sau:

npm start

Starts the development server.

npm run build

Bundles the app into static files for production.

npm test

Starts the test runner.

npm run eject

Removes this tool and copies build dependencies, configuration files

and scripts into the app directory. If you do this, you can’t go back!

We suggest that you begin by typing:

cd my-app

npm start

Tiếp theo bạn gõ lệnh “cd my-app” và “npm start” vào terminal (để chạy thử và có kết quả như sau:

A computer screen shot of a computer screen

Description automatically generated with low confidence

*Bấm CTR + C trong terminal để thoát*

Sau đó, bạn truy cập vô thư mục đã tạo và tìm folder src: Xóa các file trong đó và Copy các file trong thư mục src (Trong file tôi gửi) vào thư mục này:

A screenshot of a computer

Description automatically generated with low confidence

A screenshot of a computer

Description automatically generated with low confidence

Và bạn sẽ phát hiện ra có 2 lỗi như sau:

A screen shot of a computer program

Description automatically generated with low confidence

Để fix bạn gõ các dòng lệnh sau trong terminal:

npm install @tensorflow/tfjs

npm install react-webcam

Ngoài ra bạn còn thấy một số dòng bị lỗi do chưa format:

A picture containing screenshot, text, multimedia software, software

Description automatically generated

Để fix, bạn cần cài đặt tiện ích mở rộng trong VSCode: Prettier

A screenshot of a computer

Description automatically generated

Sau đó bạn nhấn tổ hợp "Alt + Shift + F" để format.

Tiếp theo, bạn xóa các file package và package-lock trong thư mục của bạn và copy các thư mục package, package-lock và yarn.lock từ thư mục của tôi qua:

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidence

Cuối cùng ta đã hoàn thành bài làm và bạn hãy vào lại VSCode và gõ lệnh ”npm start” để chạy thử.

## **Bước 4: Kết quả**

Hãy bấm “Ctr + Shift + I” để kiểm tra xem có các tensor ở phần console chưa:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

Cảm ơn thầy và các bạn đã xem bài báo cáo và hướng dẫn chạy một model trên web.