

TRƯỜNG ĐẠI HỌC BÁCH KHOA

KHOA KHOA HỌC & KỸ THUẬT MÁY TÍNH



LUẬN VĂN TỐT NGHIỆP(CO4313)

---

# Xây dựng hệ thống giám sát và cảnh báo chất lượng không khí trong môi trường làm việc qua thiết bị di động

---

GVHD: PGS.TS. Phạm Trần Vũ  
GVPB: ThS. Phan Đình Thế Duy  
SVTH: Phạm Khánh Huy Hoàng – 1511145

TP.HỒ CHÍ MINH, THÁNG 12/2019

# Mục lục

|  |           |
|--|-----------|
| <b>Danh mục hình vẽ</b>  | <b>3</b>  |
| <b>Danh mục chữ viết tắt</b>   | <b>4</b>  |
| <b>1 Giới thiệu</b>  | <b>4</b>  |
| 1.1 Giới thiệu đề tài và đặt vấn đề . . . . .                          | 4         |
| 1.2 Động lực nghiên cứu . . . . .                                      | 5         |
| 1.3 Khó khăn, thử thách . . . . .                                      | 5         |
| 1.4 Mục tiêu của đề tài . . . . .                                      | 6         |
| 1.5 Ý nghĩa thực tiễn . . . . .  | 6         |
| <b>2 Cơ sở lý thuyết</b>   | <b>7</b>  |
| 2.1 Cách đánh giá chất lượng không khí . . . . .                       | 7         |
| 2.2 Mô hình Web API trong lập trình ứng dụng Web . . . . .             | 8         |
| 2.3 Deep learning . . . . .  | 9         |
| 2.3.1 Kiến trúc Mạng nơ-ron nhân tạo - Artificial Neural Network (ANN) | 9         |
| 2.3.2 Hoạt động của mạng nơ-ron nhân tạo . . . . .                     | 11        |
| 2.3.3 Mạng thần kinh tái phát - Recurrent neural network . . . . .     | 14        |
| <b>3 Công nghệ được áp dụng trong đề tài</b>                           | <b>15</b> |
| 3.1 Công nghệ Internet of Things IOT . . . . .                         | 15        |
| 3.2 Phương thức truyền dữ liệu MQTT . . . . .                          | 17        |
| 3.3 Chip ESP8266 . . . . .   | 18        |
| 3.4 Nodejs Platform . . . . .  | 19        |
| 3.5 Hệ quản trị dữ liệu MongoDB . . . . .                              | 21        |
| 3.6 ReactJS Platform . . . . .   | 23        |
| <b>4 Phân tích và Thiết kế hệ thống</b>                                | <b>25</b> |
| 4.1 Tổng quan . . . . .  | 25        |

|  |           |
|--|-----------|
| <i>MỤC LỤC</i>   | 2         |
| 4.1.1 Hệ thống thiết bị sensor . . . . .                             | 26        |
| 4.1.2 Server . . . . .   | 27        |
| 4.1.3 Cơ sở dữ liệu . . . . .  | 27        |
| 4.1.4 Front-End Client . . . . .                                     | 29        |
| 4.2 Áp dụng Recurrent Neural Network vào bài toán giám sát . . . . . | 29        |
| 4.2.1 Chuẩn bị dữ liệu . . . . .                                     | 29        |
| 4.2.2 Xử lý dữ liệu . . . . .  | 30        |
| <b>5 Kết quả và đánh giá</b>   | <b>31</b> |
| <b>6 Tổng kết và hướng phát triển</b>                                | <b>32</b> |

# Danh mục hình ảnh

|     |  |    |
|-----|--|----|
| 1.1 | Nhiều tòa nhà ở TP HCM mờ đục, không khí cảm giác khô rát. . . . .                             | 5  |
| 2.1 | Các nguồn gây ô nhiễm không khí, tác động và chỉ dẫn của WHO về các chất gây ô nhiễm . . . . . | 8  |
| 2.2 | Mô hình Web Api . . . . .  | 9  |
| 2.3 | Kiến trúc mạng nơ-ron nhân tạo . . . . .   | 10 |
| 2.4 | Quá trình xử lý thông tin của một mạng nơ-ron nhân tạo . . . . .                               | 11 |
| 3.1 | Minh họa Internet of Things . . . . .  | 15 |
| 3.2 | Minh họa cách MQTT truyền dữ liệu . . . . .  | 17 |
| 3.3 | Module NodeMCU v1 LUA - ESP8266ESP12E . . . . .  | 19 |
| 3.4 | Lợi thế khi dùng Nodejs Backend . . . . .  | 20 |
| 3.5 | Cơ chế hoạt động của mongodb . . . . .   | 22 |
| 3.6 | Số lượng schema của MONGODB ít hơn của RDBMS . . . . .   | 23 |
| 3.7 | Lợi thế của việc sử dụng ReactJS trong thiết kế UI . . . . .                                   | 24 |
| 4.1 | Mô hình tổng quan của hệ thống . . . . .   | 26 |
| 4.2 | Sơ đồ thiết kế phần cứng . . . . .   | 26 |
| 4.3 | Hiện thực phần cứng sensor . . . . .   | 27 |
| 4.4 | Lược đồ cơ sở dữ liệu của ứng dụng . . . . .   | 28 |
| 4.5 | Mẫu dữ liệu tại Madrid, Tây Ban Nha . . . . .  | 30 |
| 4.6 | Quy trình áp dụng học máy . . . . .  | 30 |

## Danh mục hình ảnh

# Lời cảm ơn

Đầu tiên, em xin gửi lời cảm ơn sâu sắc đến thầy Phạm Trần Vũ, nhờ thầy hướng dẫn và định hướng đề tài mà em đã chọn. Em cảm ơn thầy đã tận tình giúp đỡ và động viên em những lúc khó khăn nhất để em có thể hoàn thành đề tài luận văn này.

Em cũng xin chân thành gửi lời cảm ơn đến quý thầy cô công tác tại Khoa Khoa học và Kỹ thuật Máy tính trường Đại học Bách Khoa TP.HCM, những người đã nhiệt tình truyền đạt kiến thức, kỹ năng trong hơn 4 năm qua để em tự tin, vững bước trên con đường tiếp theo. Em cũng xin gửi lời cảm ơn đến gia đình, bạn bè, những người đã bên cạnh động viên, giúp đỡ khi em thực hiện đề tài này.

Cuối cùng, mặc dù đã cố gắng và nỗ lực nhưng với vốn hiểu biết hạn hẹp của mình chẵn chẵn luận văn của em không tránh khỏi những thiếu sót và hạn chế. Vì vậy, em rất mong nhận được sự góp ý từ các thầy cô để đề tài được hoàn thiện hơn.

Hồ Chí Minh, ngày 25 tháng 12 năm 2019  
Phạm Khánh Huy Hoàng

## Lời cam đoan

Tôi xin cam đoan Luận văn tốt nghiệp “Xây dựng hệ thống giám sát và cảnh báo chất lượng không khí trong môi trường làm việc qua thiết bị di động.” là công trình nghiên cứu của riêng tôi dưới sự hướng dẫn của PGS.TS Phạm Trần Vũ. Nội dung nghiên cứu và các kết quả đều là trung thực và chưa từng được công bố trước đây. Các số liệu được sử dụng cho quá trình phân tích, nhận xét được chính tôi thu thập từ nhiều nguồn khác nhau và sẽ được ghi rõ trong phần tài liệu tham khảo. Ngoài ra, tôi cũng có sử dụng một số nhận xét, đánh giá và số liệu của các tác giả khác, cơ quan tổ chức khác. Tất cả đều có trích dẫn và chú thích nguồn gốc. Nếu phát hiện có bất kì sự gian lận nào, tôi xin hoàn toàn chịu trách nhiệm về nội dung Luận văn tốt nghiệp của mình. Trường Đại học Bách Khoa Thành phố Hồ Chí Minh không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện.

Người thực hiện  
Phạm Khánh Huy Hoàng

# Tóm tắt

Luận văn đưa ra giải pháp cụ thể trong việc truyền dữ liệu từ cảm biến lên server theo thời gian thực bằng cách ứng dụng công nghệ Internet of Things và xây dựng được ứng dụng web để giúp cho việc giám sát được những thông số môi trường trở nên dễ dàng ở những nơi sử dụng Internet. Ngoài ra, luận văn còn áp dụng machine learning để dự đoán các giá trị tiếp theo của các thông số môi trường.



# Chương 1

## Giới thiệu

### 1.1 Giới thiệu đề tài và đặt vấn đề

Hiện nay, thế giới đang chạy đua trong kỉ nguyên của công nghệ và kỉ nguyên của cuộc cách mạng công nghiệp 4.0. Việc áp dụng khoa học kỹ thuật vào đời sống và công việc trở nên phổ biến và giúp ích rất lớn trong việc giải thoát sức lao động của con người. Nổi bật lên có những công nghệ đã “vượt mặt” con người ở những lĩnh vực tưởng chừng chỉ có con người mới làm được. Trong đề tài Luận văn “Xây dựng hệ thống giám sát và cảnh báo chất lượng không khí trong môi trường làm việc qua thiết bị di động.”, em hướng đến mục tiêu nghiên cứu và áp dụng công nghệ để xây dựng hệ thống quan trắc không khí tự động nhằm hỗ trợ việc thu thập dữ liệu của các tác nhân bên ngoài không khí có ảnh hưởng đến con người và môi trường sống xung quanh. Bên cạnh đó, sau khi thu thập dữ liệu ứng dụng được xây dựng với hệ thống cảnh báo cho người dùng nếu các tác nhân biến đổi đột ngột hay nồng độ của chúng vượt ngưỡng bình thường có thể ảnh hưởng xấu đến môi trường và con người xung quanh.

Ở Việt Nam, hệ thống quan trắc môi trường không khí cực kỳ ít ỏi. Tuy vậy, với những số liệu ít ỏi có được, bức tranh ô nhiễm không khí ở Việt Nam cũng không sáng sủa. Theo Báo cáo Môi trường Quốc gia về môi trường không khí (phát hành năm 2019) thì không khí ở các đô thị lớn có dấu hiệu ô nhiễm bụi, đặc biệt là ô nhiễm bụi mịn PM<sub>2,5</sub> và ô nhiễm ozon tăng mạnh. Mới đây, công bố từ các chuyên gia Chính phủ Đức qua khảo sát tình trạng ô nhiễm môi trường tại TPHCM cho thấy, bụi mịn PM<sub>2,5</sub> tăng cao, nhiều hơn gấp 23 lần so với các thành phố tại Đức. Nghiên cứu mới nhất về Chỉ số hiệu suất môi trường của Đại học Yale (Mỹ) cũng cho thấy, Việt Nam là một trong 10 quốc gia ô nhiễm không khí nhất thế giới.



HÌNH 1.1: Nhiều tòa nhà ở TP HCM mờ đục, không khí cảm giác khô rát.

## 1.2 Động lực nghiên cứu

Hơn lúc nào hết, việc giám sát mức độ ô nhiễm không khí và chia sẻ những thông tin này đang trở nên vô cùng quan trọng để có thể thực hiện những hành động hiệu quả. Hiện nay, những trạm đo, những thiết bị đo chỉ số không khí đang được lắp đặt ở nhiều nơi, những địa điểm mà việc giám sát là cần thiết như các thành phố lớn, các khu công nghiệp, Do đó, việc xác định những hành vi bất thường môi trường xung quanh có ý nghĩa hết sức thiết thực đến sức khỏe và đời sống. Kết quả của bài toán hoàn toàn có khả năng áp dụng vào thực tiễn để mang lại nhiều lợi ích cho xã hội. Đây là một đề tài hết sức thú vị do những tác động tích cực mà nó mang đến, bên cạnh đó đây cũng là một đề tài đầy thách thức khi đòi hỏi phải nắm bắt và vận dụng nhiều kiến thức mới có thể hoàn thành tốt đề tài.

## 1.3 Khó khăn, thử thách

Vấn đề nghiên cứu còn mới ở Việt Nam, dữ liệu và trang thiết bị bị hạn chế. Hệ thống gồm nhiều phần, liên quan đến nhiều vấn đề khoa học và kỹ thuật nên thực hiện đề tài khá gian nan.

## 1.4 Mục tiêu của đề tài

Đề tài hướng tới mục đích tìm hiểu, khảo sát, áp dụng và xây dựng hệ thống giám sát và cảnh báo các chỉ số chỉ hướng đến chất lượng không khí. Đánh giá chất lượng môi trường không khí trong phòng làm việc do ảnh hưởng của các tác nhân không khí khác nhau. Nghiên cứu, xây dựng phương pháp đánh giá tổng hợp chất lượng không khí. Xây dựng và phát triển hệ thống cảnh báo khi các tác nhân không khí biến đổi có khả năng gây hại cho con người.

## 1.5 Ý nghĩa thực tiễn

Khi khối trung tâm xử lý của IoT được tích hợp công nghệ trí tuệ nhân tạo (AI) sẽ đem lại khả năng dự báo, cảnh báo gần như là theo thời gian thực hay khi được tích hợp Bigdata với IoT sẽ mang lại khả năng phân tích xử lý dữ liệu, thống kê, lưu trữ dữ liệu rất tốt. Ứng dụng IoT mang lại những lợi ích thiết thực không chỉ cho người dân, doanh nghiệp, mà còn là các cơ quan quản lý như phân tích dữ liệu lớn trong các ứng dụng IoT như là số liệu thống kê để tham khảo phân tích các dự báo, hành vi người dùng và phương pháp xử lý dữ liệu nâng cao (bao gồm AI). Hệ thống được ứng dụng trong việc giám sát, cảnh báo mức độ ô nhiễm không khí sẽ giúp cho người dân sẽ nắm được thông tin về tình trạng môi trường không khí trong khu vực sinh sống và làm việc một cách nhanh chóng và chính xác nhất từ đó giúp họ đưa ra những quyết định phù hợp. Bên cạnh đó, khi các chỉ số môi trường không khí vượt ngưỡng cho phép, gây hại cho sức khỏe con người thì hệ thống IoT sẽ ngay lập tức phân tích xử lý và đưa ra các cảnh báo để người dân kịp thời ứng phó.

## Chương 2

# Cơ sở lý thuyết

### 2.1 Cách đánh giá chất lượng không khí

Có bốn vấn đề chính trong các khu vực đô thị xảy ra ô nhiễm môi trường không khí gây ảnh hưởng đến sức khỏe của người dân:

- Ô nhiễm không khí do các chất gây ô nhiễm hóa học và tác nhân sinh học
- Phát tán khí thải nhà kính và sự biến đổi khí hậu
- Ô nhiễm tiếng ồn
- Bức xạ và điện từ trường

Không khí bị ô nhiễm là do các chất gây ô nhiễm hóa học xảy ra trong môi trường cả trong nhà và ngoài trời, trong đó phần lớn là do môi trường trong nhà, nơi con người dành hầu hết thời gian ở đó. Tác nhân sinh học là nguyên nhân chính gây ô nhiễm không khí trong nhà. Tiếng ồn, chất phóng xạ, điện từ trường cũng là các tác nhân chính gây ô nhiễm cả môi trường ngoài trời và trong nhà. Trong đề tài này, tập trung chủ yếu vào ô nhiễm không khí ngoài trời do các hợp chất hóa học gây ra.

Các loại chất gây ô nhiễm không khí chính là các hạt bụi (PM10, PM2.5) bao gồm các phân tử các bon đen siêu nhỏ,  $O_3$ ,  $CO$ ,  $NO_2$ , các chất hữu cơ bay hơi, hiđrôcacbon và chất quang oxy hóa như ô zôn. Tuy nhiên, ở các thành phố đang phát triển, chất gây ô nhiễm không khí chính là hạt bụi và ô zôn. Về vấn đề các hạt bụi, chỉ số được báo cáo chung nhất là nồng độ PM10 (hạt có kích thước động học với đường kính nhỏ hơn 10  $\mu m$ ). Ở rất nhiều thành phố Châu Á, nồng độ trung bình PM10 vượt quá 100  $\mu g/m^3$  với mức cao nhất là 200  $\mu g/m^3$ . Hình 2.1 mô tả nguồn gây ô nhiễm không khí, tác động và chỉ dẫn của

WHO về các chất gây ô nhiễm. Qua đó xác định được khoảng vượt ngưỡng của các chất gây ô nhiễm.

| Chất gây ô nhiễm                        | Nguồn chính   | Tác động  | Chỉ dẫn về sức khỏe (WHO 2002a)  |
|---|---|---|--|
| <b>Carbon monoxide (CO)</b>             | Khí thải động cơ, các hoạt động công nghiệp   | Gây độc cho người khi hít phải, CO giảm khả năng vận chuyển ô xi trong máu và tăng áp lực lên tim và phổi | 10 mg/m <sup>3</sup> (10ppm) trên 8 tiếng; 30 mg/m <sup>3</sup> trên 1 tiếng (30,000 ug/m <sup>3</sup> ) |
| <b>Sulphur dioxide (SO<sub>2</sub>)</b> | Một phần nhỏ từ các nguồn di động. Nhiệt và năng lượng sản sinh từ việc sử dụng than và dầu chứa sulphur, sulphuric acid plants | Gây trở ngại cho con người, SO <sub>2</sub> tạo phản ứng với không khí tạo ra mưa a xít                   | 20 ug/m <sup>3</sup> trên 24 tiếng<br>500 ug/m <sup>3</sup> trên 10 phút                                 |
| <b>Hạt bụi PM<sub>10</sub></b>          | Đất, bụi nước biển (oceanic spray), cháy rừng, đun nấu trong nhà, phương tiện, hoạt   | Tăng khả năng ung thư, trường hợp tử vong, làm nghiêm   | 50 ug/m <sup>3</sup> trên 24 tiếng<br>20 ug/m <sup>3</sup> trung bình năm                                |
| <b>Hạt bụi PM<sub>2.5</sub></b>         |   |   | 25 ug/m <sup>3</sup> trên 24 tiếng   |

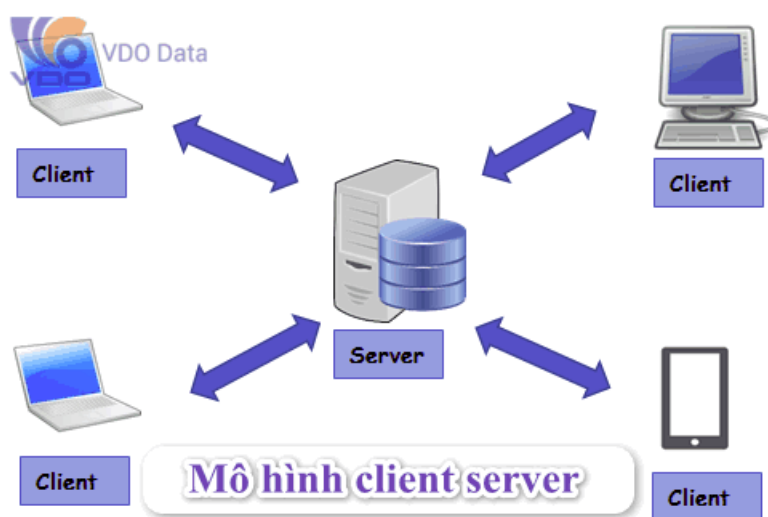
HÌNH 2.1: Các nguồn gây ô nhiễm không khí, tác động và chỉ dẫn của WHO về các chất gây ô nhiễm

## 2.2 Mô hình Web API trong lập trình ứng dụng Web

**API** là các phương thức, giao thức kết nối với các thư viện và ứng dụng khác. Nó là viết tắt của Application Programming Interface – giao diện lập trình ứng dụng. API cung cấp khả năng cung cấp khả năng truy xuất đến một tập các hàm hay dùng. Và từ đó có thể trao đổi dữ liệu giữa các ứng dụng. **Web API** là một phương thức dùng để cho phép các ứng dụng khác nhau có thể giao tiếp, trao đổi dữ liệu qua lại. Dữ liệu được Web API trả lại thường ở dạng JSON hoặc XML thông qua giao thức HTTP hoặc HTTPS. Web API hỗ trợ restful đầy đủ các phương thức: Get/Post/Put/Delete dữ liệu. Nó giúp bạn xây dựng các HTTP service một cách rất đơn giản và nhanh chóng. Nó cũng có khả năng hỗ trợ đầy đủ các thành phần HTTP: URI, request/response headers, caching, versioning, content format.

- Đầu tiên là xây dựng URL API để bên thứ ba có thể gửi request dữ liệu đến máy chủ cung cấp nội dung, dịch vụ thông qua giao thức HTTP hoặc HTTPS.
- Tại web server cung cấp nội dung, các ứng dụng nguồn sẽ thực hiện kiểm tra xác thực nếu có và tìm đến tài nguyên thích hợp để tạo nội dung trả về kết quả.
- Server trả về kết quả theo định dạng JSON hoặc XML thông qua giao thức HTTP/HTTPS.

- Tại nơi yêu cầu ban đầu là ứng dụng web hoặc ứng dụng di động, dữ liệu JSON/XML sẽ được parse để lấy data. Sau khi có được data thì thực hiện tiếp các hoạt động như lưu dữ liệu xuống Cơ sở dữ liệu, hiển thị dữ liệu...



HÌNH 2.2: Mô hình Web Api

## 2.3 Deep learning

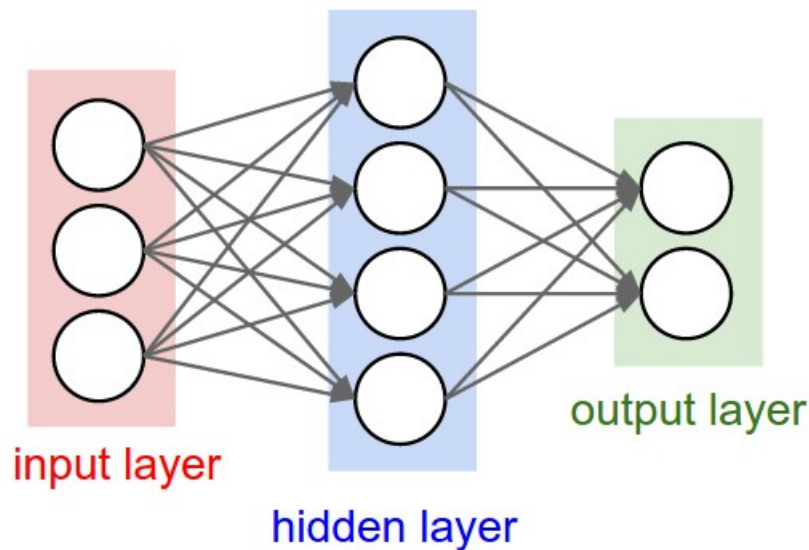
Trong phần này giới thiệu về cơ sở lý thuyết về mạng nơ-ron nhân tạo, cách thức hoạt động của mạng nơ-ron, các phiên bản mở rộng của mạng nơ-ron nhân tạo, như: Mạng nơ-ron tái phát (RNN). Mạng nơ-ron tái phát RNN là một trong những mô hình Deep learning được đánh giá có nhiều ưu điểm trong các tác vụ xử lý dữ liệu thời gian thực. Đây cũng là cơ sở thực hiện để xây dựng mô hình dự đoán trong đề tài luận văn.

### 2.3.1 Kiến trúc Mạng nơ-ron nhân tạo - Artificial Neural Network (ANN)

Mạng nơ-ron nhân tạo (Artificial Neural Network – ANN) là một mô hình xử lý thông tin được mô phỏng dựa trên hoạt động của hệ thống thần kinh của sinh vật, bao gồm số lượng lớn các Nơ-ron được gắn kết để xử lý thông tin. ANN hoạt động giống như bộ não của con người, được học bởi kinh nghiệm (thông qua việc huấn luyện), có khả năng lưu giữ các tri thức và sử dụng các tri thức đó trong việc dự đoán các dữ liệu chưa biết (unseen).

data).

Một mạng nơ-ron là một nhóm các nút nối với nhau, mô phỏng mạng nơron thần kinh của não người. Mạng nơ-ron nhân tạo được thể hiện thông qua ba thành phần cơ bản: mô hình của nơ-ron, cấu trúc và sự liên kết giữa các nơ-ron. Trong nhiều trường hợp, mạng nơ-ron nhân tạo là một hệ thống thích ứng, tự thay đổi cấu trúc của mình dựa trên các thông tin bên ngoài hay bên trong chạy qua mạng trong quá trình học.



HÌNH 2.3: Kiến trúc mạng nơ-ron nhân tạo

Kiến trúc chung của một ANN gồm 3 thành phần đó là **Input Layer**, **Hidden Layer** và **Output Layer**. Trong đó, lớp ẩn (Hidden Layer) gồm các nơ-ron, nhận dữ liệu input từ các Nơ-ron ở lớp (Layer) trước đó và chuyển đổi các input này cho các lớp xử lý tiếp theo. Trong một mạng ANN có thể có nhiều Hidden Layer.

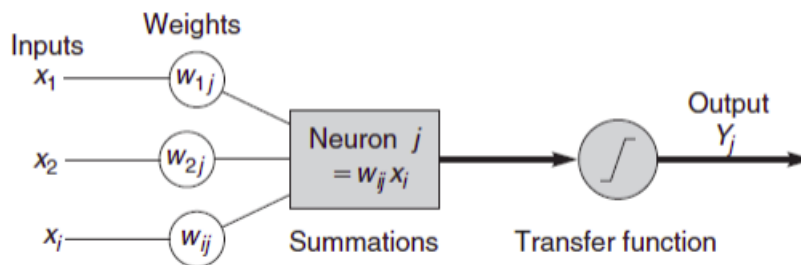
Lợi thế lớn nhất của các mạng ANN là khả năng được sử dụng như một cơ chế xấp xỉ hàm tùy ý mà “học” được từ các dữ liệu quan sát. Tuy nhiên, sử dụng chúng không đơn giản như vậy, một số các đặc tính và kinh nghiệm khi thiết kế một mạng nơ-ron ANN.

- **Chọn mô hình:** Điều này phụ thuộc vào cách trình bày dữ liệu và các ứng dụng. Mô hình quá phức tạp có xu hướng dẫn đến những thách thức trong quá trình học.
- **Cấu trúc và sự liên kết giữa các nơ-ron.**

- **Thuật toán học:** Có hai vấn đề cần học đối với mỗi mạng ANN, đó là học tham số của mô hình (parameter learning) và học cấu trúc (structure learning). Học tham số là thay đổi trọng số của các liên kết giữa các nơ-ron trong một mạng, còn học cấu trúc là việc điều chỉnh cấu trúc mạng bằng việc thay đổi số lớp ẩn, số nơ-ron mỗi lớp và cách liên kết giữa chúng. Hai vấn đề này có thể được thực hiện đồng thời hoặc tách biệt.

Nếu các mô hình, hàm chi phí và thuật toán học được lựa chọn một cách thích hợp, thì mạng ANN sẽ cho kết quả có thể vô cùng mạnh mẽ và hiệu quả.

### 2.3.2 Hoạt động của mạng nơ-ron nhân tạo



HÌNH 2.4: Quá trình xử lý thông tin của một mạng nơ-ron nhân tạo

- **Inputs:** Mỗi Input tương ứng với 1 đặc trưng của dữ liệu. Ví dụ như trong ứng dụng của ngân hàng xem xét có chấp nhận cho khách hàng vay tiền hay không thì mỗi input là một thuộc tính của khách hàng như thu nhập, nghề nghiệp, tuổi, số con,...
- **Output:** Kết quả của một ANN là một giải pháp cho một vấn đề, ví dụ như với bài toán xem xét chấp nhận cho khách hàng vay tiền hay không thì output là yes hoặc no.
- **Connection Weights** (Trọng số liên kết): Đây là thành phần rất quan trọng của một ANN, nó thể hiện mức độ quan trọng, độ mạnh của dữ liệu đầu vào đối với quá trình xử lý thông tin chuyển đổi dữ liệu từ Layer này sang layer khác. Quá trình học của ANN thực ra là quá trình điều chỉnh các trọng số Weight của các dữ liệu đầu vào để có được kết quả mong muốn.



- **Summation Function (Hàm tổng):** Tính tổng trọng số của tất cả các input được đưa vào mỗi Nơ-ron. Hàm tổng của một Nơ-ron đối với n input được tính theo công thức sau:

$$Y = \sum_{i=1}^n X_i W_i$$

- **Transfer Function (Hàm chuyển đổi):** Hàm tổng của một nơ-ron cho biết khả năng kích hoạt của nơ-ron đó còn gọi là kích hoạt bên trong. Các nơ-ron này có thể sinh ra một output hoặc không trong mạng ANN, nói cách khác rằng có thể output của 1 Nơ-ron có thể được chuyển đến layer tiếp trong mạng Nơ-ron theo hoặc không. Mối quan hệ giữa hàm tổng và kết quả output được thể hiện bằng hàm chuyển đổi.

Việc lựa chọn hàm chuyển đổi có tác động lớn đến kết quả đầu ra của mạng ANN. Hàm chuyển đổi phi tuyến được sử dụng phổ biến trong mạng ANN là hoặc sigmoid hoặc tanh.

$$Y = \sum_{i=1}^n X_i W_i$$

Trong đó, hàm tanh là phiên bản thay đổi tỉ lệ của sigmoid, tức là khoảng giá trị đầu ra của hàm chuyển đổi thuộc khoảng  $[-1, 1]$  thay vì  $[0, 1]$  nên chúng còn gọi là hàm chuẩn hóa (Normalized Function).

Kết quả xử lý tại các nơ-ron (Output) đôi khi rất lớn, vì vậy hàm chuyển đổi được sử dụng để xử lý output này trước khi chuyển đến layer tiếp theo. Đôi khi thay vì sử dụng Transfer Function người ta sử dụng giá trị ngưỡng (Threshold value) để kiểm soát các output của các neuron tại một layer nào đó trước khi chuyển các output này đến các Layer tiếp theo. Nếu output của một neuron nào đó nhỏ hơn Threshold thì nó sẽ không được chuyển đến Layer tiếp theo.

Mạng nơ-ron của chúng ta dự đoán dựa trên lan truyền thẳng (forward propagation) là các phép nhân ma trận cùng với activation function để thu được kết quả đầu

ra. Nếu input  $x$  là vector 2 chiều thì ta có thể tính kết quả dự đoán  $\hat{y}$  bằng công thức sau

$$\begin{aligned}z_1 &= xW_1 + b_1 \\a_1 &= \tanh(z_1) \\z_2 &= a_1W_2 + b_2 \\a_2 &= \hat{y} = \text{softmax}(z_2)\end{aligned}$$

Trong đó,  $z_i$  là input của layer thứ  $i$ ,  $a_i$  là output của layer thứ  $i$  sau khi áp dụng activation function.  $W_1, b_1, W_2, b_2$  là các thông số (parameters) cần tìm của mô hình mạng nơ-ron. Huấn luyện để tìm các thông số cho mô hình tương đương với việc tìm các thông số  $W_1, b_1, W_2, b_2$  sao cho độ lỗi của mô hình đạt được là thấp nhất. Ta gọi hàm độ lỗi của mô hình là loss function. Đối với softmax function, ta dùng crossentropy loss (còn gọi là negative log likelihood). Nếu ta có  $N$  dòng dữ liệu huấn luyện, và  $C$  nhóm phân lớp (trường hợp này là hai lớp nam, nữ), khi đó loss function giữa giá trị dự đoán  $\hat{y}$  và  $y$  được tính như sau

$$L(y, \hat{y}) = -\frac{1}{N} \sum_{n \in N} \sum_{i \in C} y_{n,i} \log \hat{y}_{n,i}$$

Ý nghĩa công thức trên nghĩa là: lấy tổng trên toàn bộ tập huấn luyện và cộng dồn vào hàm loss nếu kết quả phân lớp sai. Độ dị biệt giữa hai giá trị  $\hat{y}$  và  $y$  càng lớn thì độ lỗi càng cao. Mục tiêu của chúng ta là tối thiểu hóa hàm lỗi này. Ta có thể sử dụng phương pháp gradient descent để tối thiểu hóa hàm lỗi. Có hai loại gradient descent, một loại với fixed learning rate được gọi là batch gradient descent, loại còn lại có learning rate thay đổi theo quá trình huấn luyện được gọi là SGD (stochastic gradient descent) hay minibatch gradient descent.

Gradient descent cần các gradient là các vector có được bằng cách lấy đạo hàm của loss function theo từng thông số  $\frac{\partial L}{\partial W_1}, \frac{\partial L}{\partial b_1}, \frac{\partial L}{\partial W_2}, \frac{\partial L}{\partial b_2}$ . Để tính các gradient này, ta sử dụng thuật toán backpropagation (lan truyền ngược). Đây là cách hiệu quả để tính gradient khởi điểm từ Output layer.

Áp dụng backpropagation ta có các đại lượng:

$$\begin{aligned}\delta_3 &= y - \hat{y} \\ \delta_2 &= (1 - \tanh^2 z_1) \circ \delta_3 W_2^T \\ \frac{\partial L}{\partial W_2} &= a_1^T \delta_3 \\ \frac{\partial L}{\partial b_2} &= \delta_3 \\ \frac{\partial L}{\partial W_1} &= x^T \delta_2 \\ \frac{\partial L}{\partial b_1} &= \delta_2\end{aligned}$$

### 2.3.3 Mạng thần kinh tái phát - Recurrent neural network

Mạng thần kinh tái phát (RNN) là một lớp ANN trong đó các kết nối giữa các đơn vị tạo thành một biểu đồ phối hợp dọc theo một chuỗi. Ý tưởng của RNN đó là thiết kế một Neural Network sao cho có khả năng xử lý được thông tin dạng chuỗi (sequential information), ví dụ một câu là một chuỗi gồm nhiều từ.

Recurrent có nghĩa là thực hiện lặp lại cùng một tác vụ cho mỗi thành phần trong chuỗi. Trong đó, kết quả đầu ra tại thời điểm hiện tại phụ thuộc vào kết quả tính toán của các thành phần ở những thời điểm trước đó.

Nói cách khác, RNN là một mô hình có trí nhớ (memory), có khả năng nhớ được thông tin đã tính toán trước đó. Không như các mô hình Neural Network truyền thống đó là thông tin đầu vào (input) hoàn toàn độc lập với thông tin đầu ra (output). Về lý thuyết, RNN có thể nhớ được thông tin của chuỗi có chiều dài bất kì, nhưng trong thực tế mô hình này chỉ nhớ được thông tin ở vài bước trước đó.

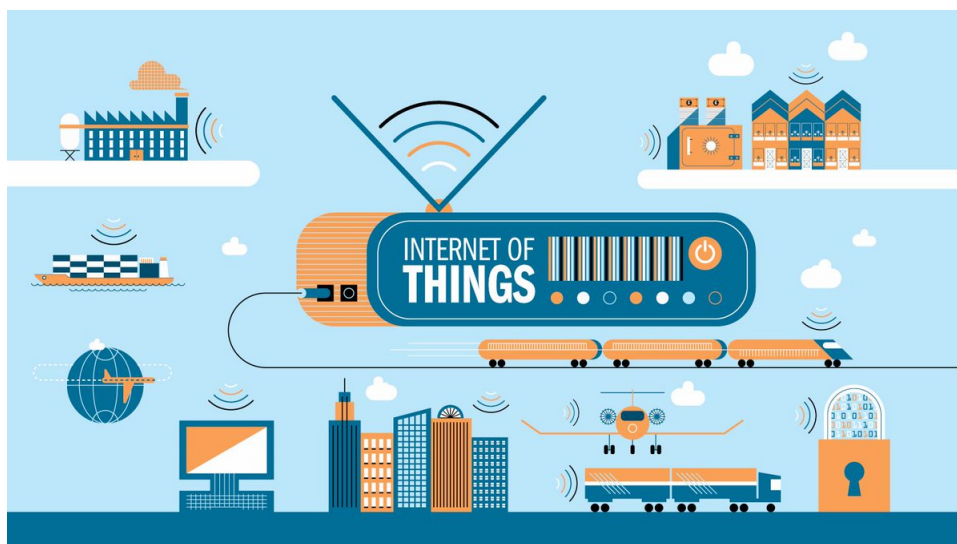
## Chương 3

# Công nghệ được áp dụng trong đề tài

### 3.1 Công nghệ Internet of Things IOT

#### Khái Niệm

Mạng lưới vạn vật kết nối Internet hoặc là mạng lưới thiết bị kết nối Internet viết tắt là IoT (tiếng Anh: Internet of Things) là một kịch bản của thế giới, khi mà mỗi đồ vật, con người được cung cấp một định danh của riêng mình, và tất cả có khả năng truyền tải, trao đổi thông tin, dữ liệu qua một mạng duy nhất mà không cần đến sự tương tác trực tiếp giữa người với người, hay người với máy tính. IoT đã phát triển từ sự hội tụ của công nghệ không dây, công nghệ vi cơ điện tử và Internet. Nói đơn giản là một tập hợp các thiết bị có khả năng kết nối với nhau, với Internet và với thế giới bên ngoài để thu thập thông tin về môi trường xung quanh để thực hiện một công việc cụ thể.



HÌNH 3.1: Minh họa Internet of Things

Controller đóng vai trò trung gian giữa Model và View. Xử lý các yêu cầu của người dùng. Làm việc với Model và lựa chọn View để thể hiện nội dung một cách thích hợp. Trong mô hình MVC, View chỉ hiển thị thông tin, Controller nhận, xử lý, phản hồi các yêu cầu, tương tác của người dùng và Model để xử lý các yêu cầu từ controller xuống cơ sở dữ liệu.

### Đặc điểm của Internet of Things

- Tính năng kết nối liên thông (interconnectivity): Đối với IOT bất cứ thứ gì cũng có thể kết nối được với nhau nhờ mạng lưới thông tin và các cơ sở hạ tầng liên lạc tổng thể.
- Những dịch vụ có liên quan đến “Things”: Hệ thống của IOT có khả năng cung cấp các dịch vụ liên quan đến “Things”. Như bảo vệ sự nhất quán và vô tư giữa Virtual với Physical Thing. Để cung cấp được dịch vụ này, Cả công nghệ thông tin (phần mềm) và công nghệ phần cứng sẽ được thay đổi.
- Tính năng không đồng nhất: Các thiết bị trong IOT không đồng nhất vì trong nó có các phần cứng khác nhau, network cũng khác nhau. Các thiết bị giữ network sẽ có khả năng tương tác với nhau nhờ vào sự liên kết của network.
- Tính năng thay đổi kinh hoạt: Những Status của cá thiết bị sẽ được tự động thay đổi. Ngoài ra số lượng thiết bị có thể sẽ bị thay đổi.
- Với quy mô lớn: Sẽ có một khối lượng lớn các thiết bị được quản lý cùng với việc giao tiếp với nhau. Các số lượng này sẽ lớn hơn nhiều so với lượng máy tính được kết nối mạng Internet hiện nay. Tất cả các số lượng thông tin được truyền tải bởi thiết bị sẽ lớn hơn rất nhiều so với việc được truyền qua con người.

### Ưu và nhược điểm của Internet of Things

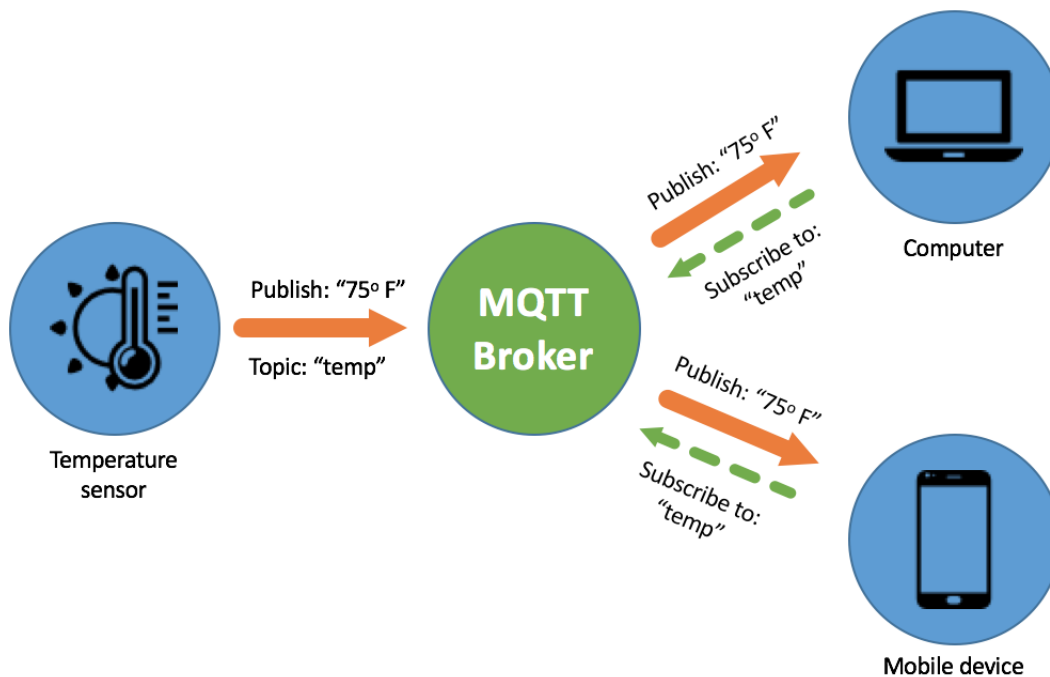
- Ưu điểm
  - Kết nối và truy xuất thông tin nhanh
  - Tiện dụng cho mọi hoạt động của con người
  - Tiết kiệm được thời gian khi sử dụng IOT
  - Nâng cao năng xuất trong lao động và sản xuất
- Nhược điểm

- Chi phí để triển khai hệ thống hoàn thiện cao
- Tính bảo mật về thông tin chưa cao

## 3.2 Phương thức truyền dữ liệu MQTT

### Khái niệm

MQTT là một giao thức publish/subscribe bản tin, được thiết kế cho công nghệ M2M (Machine to Machine) gọn nhẹ. MQTT kiểm soát các gói tin header được giữ sao cho càng nhỏ càng tốt. Mỗi gói tin điều khiển MQTT bao gồm ba phần, phần header cố định, phần header thay đổi và payload. Mỗi gói tin điều khiển MQTT có header cố định 2 byte. Không phải tất cả các gói tin điều khiển có phần header thay đổi và payload. Phần header thay đổi chứa thông tin nhận dạng của gói tin nếu nó được sử dụng bởi gói tin kiểm soát. Phần payload có dung lượng lên đến 256 MB được đính kèm trong các gói tin. Việc có một header nhỏ làm cho giao thức này phù hợp với ứng dụng IoT bằng cách giảm lưu lượng dữ liệu truyền qua mạng hạn chế băng thông.



HÌNH 3.2: Minh họa cách MQTT truyền dữ liệu

## Ưu điểm và nhược điểm của MQTT

- Ưu điểm: Ít tốn băng thông, độ tin cậy cao
- Nhược điểm: Độ bảo mật thông tin cao

## 3.3 Chip ESP8266

### Khái quát

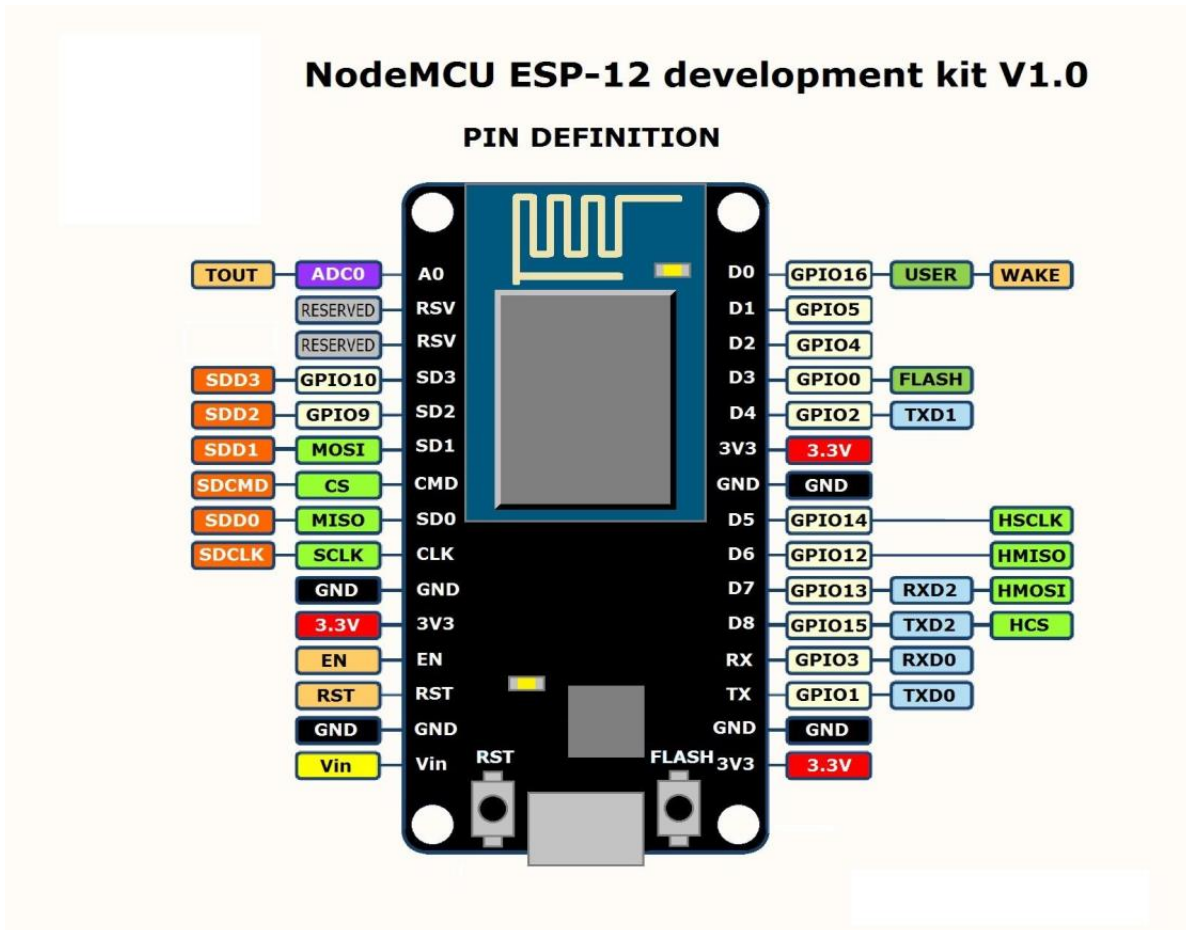
ESP8266 là dòng chip tích hợp Wi-Fi 2.4Ghz có thể lập trình được, giá rẻ được sản xuất bởi một công ty bán dẫn Trung Quốc: Espressif Systems. Đây là một dự án mã nguồn mở giúp hỗ trợ môi trường phát triển Arduino cho ESP8266. ESP8266 Arduino core đi kèm với thư viện kết nối WiFi hỗ trợ TCP, UDP và các ứng dụng HTTP, mDNS, SSDP, DNS Servers giúp cho việc tiếp cận và xây dựng ứng dụng rất nhanh.

### Thông số phần cứng

- 32-bit RISC CPU : Tensilica Xtensa LX106 chạy ở xung nhịp 80 MHz
- Hỗ trợ Flash ngoài từ 512KiB đến 4MiB
- 64KBytes RAM thực thi lệnh
- 96KBytes RAM dữ liệu
- 64KBytes boot ROM
- Chuẩn wifi IEEE 802.11 b/g/n, Wi-Fi 2.4 GHz
- Tích hợp giao thức TCP/IP
- Hỗ trợ nhiều loại anten
- 16 chân GPIO
- Hỗ trợ SDIO 2.0, UART, SPI, I2C, PWM, I2S với DMA
- 1 ADC 10-bit

- Dải nhiệt độ hoạt động rộng : -40C 125C

Phần mềm sử dụng lập trên máy tính cho Chip ESP8266 là Arduino, ngôn ngữ lập trình C/C++.



HÌNH 3.3: Module NodeMCU v1 LUA - ESP8266ESP12E

## 3.4 Nodejs Platform

### Khái niệm

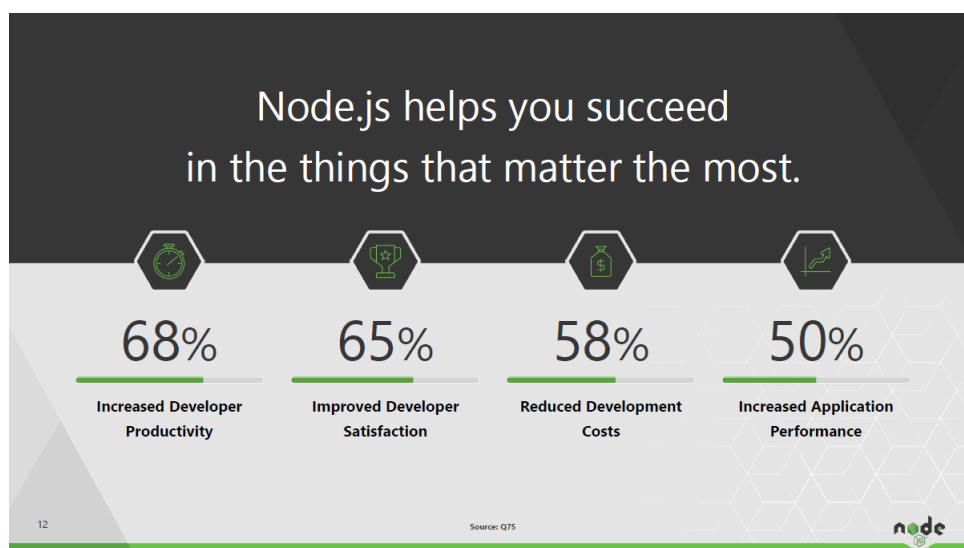
Node.js là một Javascript Run time Cross Platform được xây dựng dựa trên mã nguồn mở Google's V8 JavaScript engine cho Chrome (Browser). Node.js cho phép các lập trình viên có thể xây dựng ứng dụng Server Side, truy cập vào tài nguyên hệ thống và thực hiện được phần lớn các tác vụ hệ điều hành có thể thực hiện bằng ngôn ngữ Javascript, hoặc liên kết C++.



Nền tảng Cloud của gần như tất cả các nhà phát triển lớn hiện nay đều hỗ trợ thực thi Node.js, điển hình như Amazon Lambda, Google Script, IBM Bluemix, Microsoft Azure ...

Ngôn ngữ lập trình Javascript được cải tiến liên tục, hiện nay là EcmaScript 6 (ES5, ES2015) và đang được cải tiến rất nhanh, với nhiều ưu điểm như dễ học, xúc tích, OOP...

Một lý do Node.js được ưa chuộng nữa là đa phần các lập trình viên viết Web, Mobile đều biết, và giờ đây, nhờ Node.js mà họ có thể triển khai các ứng dụng Server Side bằng Javascript, mà không cần dùng ngôn ngữ nào khác (như trước kia phải cần Java, PHP ...)



HÌNH 3.4: Lợi thế khi dùng Nodejs Backend

### Lý do sử dụng Nodejs

Một hệ thống Internet of Things đầy đủ khá phức tạp, bao gồm thiết bị, Server xử lý kết nối, Server dữ liệu (Database), các hệ thống cân bằng tải, các hệ thống phân tích, báo cáo dữ liệu, trí tuệ nhân tạo. Server là một thành phần không thể thiếu trong hệ thống IoT. Với nhiều ưu điểm của Node.js thì nó rất phù hợp trong việc phát triển các Server cho IoT trong tương lai. Ngoài ra, Node.js được cộng đồng hỗ trợ rất nhiều, và không khó để tìm thấy 1 package cần thiết, tiết kiệm rất nhiều thời gian phát triển ứng dụng.

## 3.5 Hệ quản trị dữ liệu MongoDB

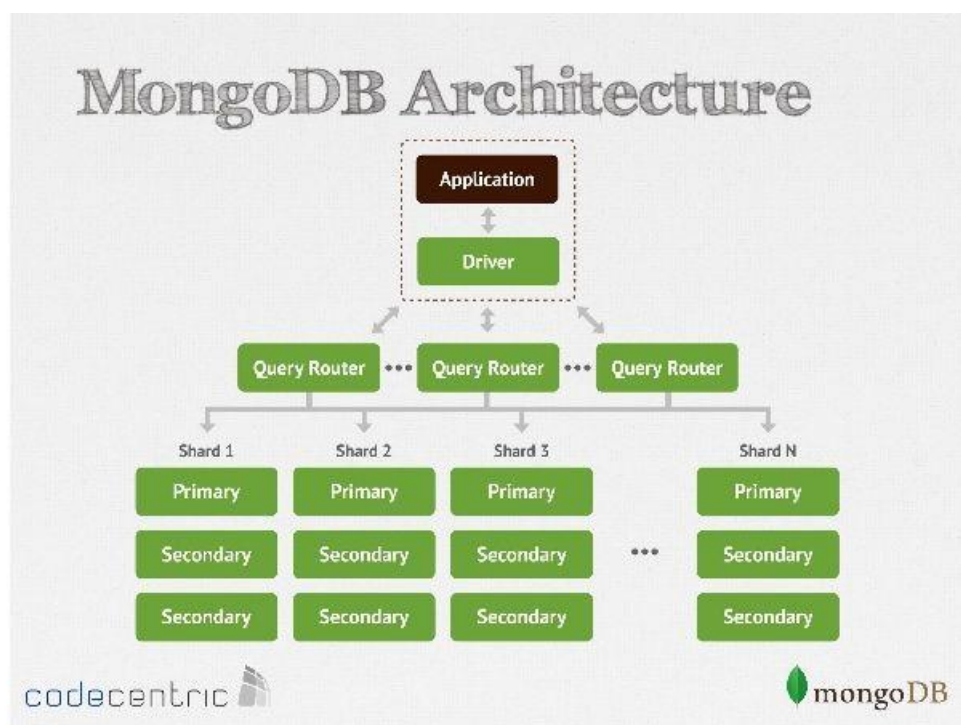
### Khái niệm

MongoDB là một cơ sở dữ liệu mã nguồn mở và là cơ sở dữ liệu NoSQL(\*) hàng đầu, được hàng triệu người sử dụng. MongoDB được viết bằng C++. Ngoài ra, MongoDB là một cơ sở dữ liệu đa nền tảng, hoạt động trên các khái niệm Collection và Document, nó cung cấp hiệu suất cao, tính khả dụng cao và khả năng mở rộng dễ dàng.

### Cách hoạt động của MongoDB

MongoDB hoạt động dưới một tiến trình ngầm service, luôn mở một cổng (Cổng mặc định là 27017) để lắng nghe các yêu cầu truy vấn, thao tác từ các ứng dụng gửi vào sau đó mới tiến hành xử lý. Mỗi một bản ghi của MongoDB được tự động gắn thêm một field có tên **id** thuộc kiểu dữ liệu **ObjectId** mà nó quy định để xác định được tính duy nhất của bản ghi này so với bản ghi khác, cũng như phục vụ các thao tác tìm kiếm và truy vấn thông tin về sau. Trường dữ liệu **id** luôn được tự động đánh **index** (chỉ mục) để tốc độ truy vấn thông tin đạt hiệu suất cao nhất. Mỗi khi có một truy vấn dữ liệu, bản ghi được **cache** (ghi đệm) lên bộ nhớ Ram, để phục vụ lượt truy vấn sau diễn ra nhanh hơn mà không cần phải đọc từ ổ cứng.

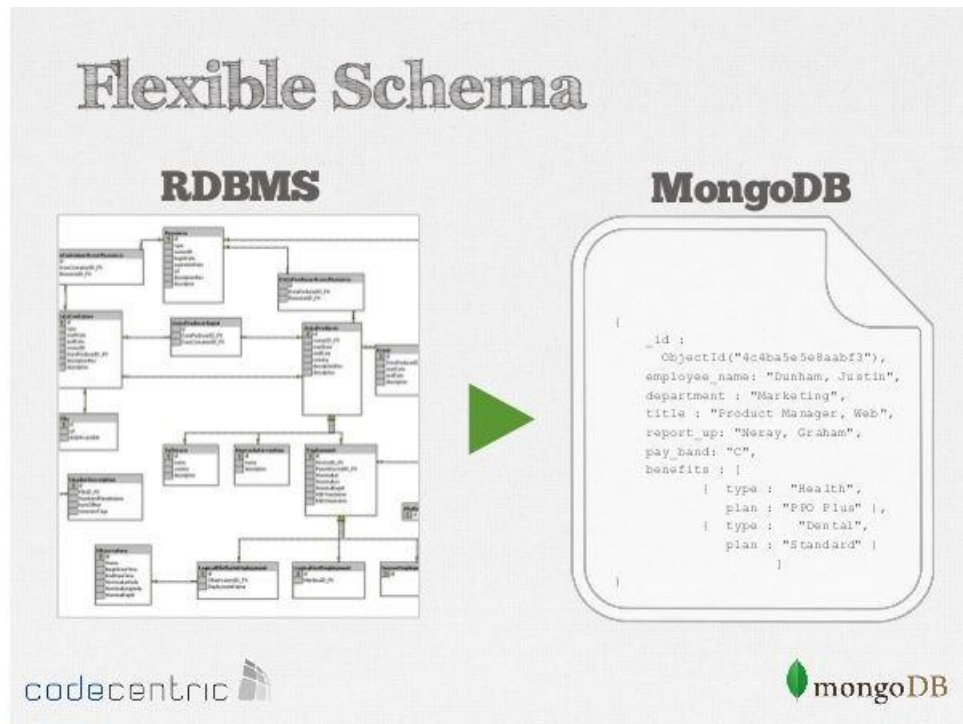
Khi có yêu cầu thêm/sửa/xóa bản ghi, để đảm bảo hiệu suất của ứng dụng mặc định MongoDB sẽ chưa cập nhật xuống ổ cứng ngay, mà sau 60 giây MongoDB mới thực hiện ghi toàn bộ dữ liệu thay đổi từ RAM xuống ổ cứng.



HÌNH 3.5: Cơ chế hoạt động của mongodb

### Lợi thế khi dùng Nodejs Backend

MongoDB là phần mềm mã nguồn mở miễn phí, có cộng đồng phát triển rất lớn, hiệu năng cao với tốc độ truy vấn nhanh hơn hẳn so với các hệ quản trị cơ sở dữ liệu quan hệ (RDBMS). Khả năng mở rộng cực lớn: việc mở rộng dữ liệu mà không phải lo đến các vấn đề như khóa ngoại, khóa chính, kiểm tra ràng buộc,... MongoDB cho phép thực hiện replication và sharding nên việc mở rộng cũng thuận lợi.

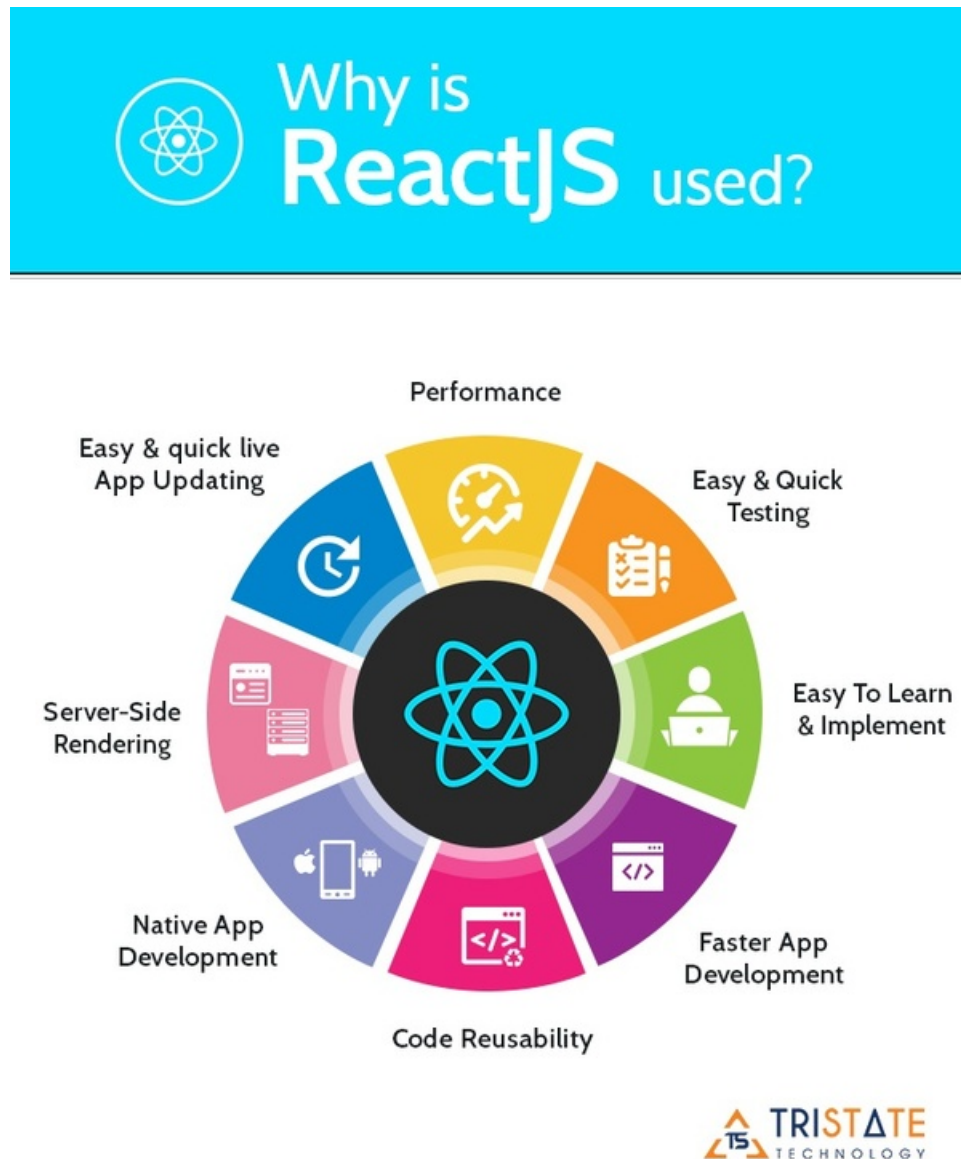


HÌNH 3.6: Số lượng schema của MONGODB ít hơn của RDBMS

## 3.6 ReactJS Platform

### Khái niệm

ReactJS (còn được gọi là React hoặc React.js) là một thư viện JavaScript để xây dựng giao diện (UI) người dùng. Nó được duy trì bởi Facebook và một cộng đồng gồm các lập trình viên mã nguồn mở. ReactJS có thể được sử dụng trong việc phát triển các ứng dụng web hoặc di động, vì nó là tối ưu để tìm nạp dữ liệu thay đổi nhanh chóng cần được ghi lại. Mục đích cốt lõi của ReactJS không chỉ khiến cho trang web phải thật mượt mà còn phải nhanh, khả năng mở rộng cao và đơn giản. Sức mạnh của nó xuất phát từ việc tập trung vào các thành phần riêng lẻ. Chính vì vậy, thay vì làm việc trên toàn bộ ứng dụng web, ReactJS cho phép một developer có thể phá vỡ giao diện người dùng phức tạp thành các thành phần đơn giản hơn.



HÌNH 3.7: Lợi thế của việc sử dụng ReactJS trong thiết kế UI

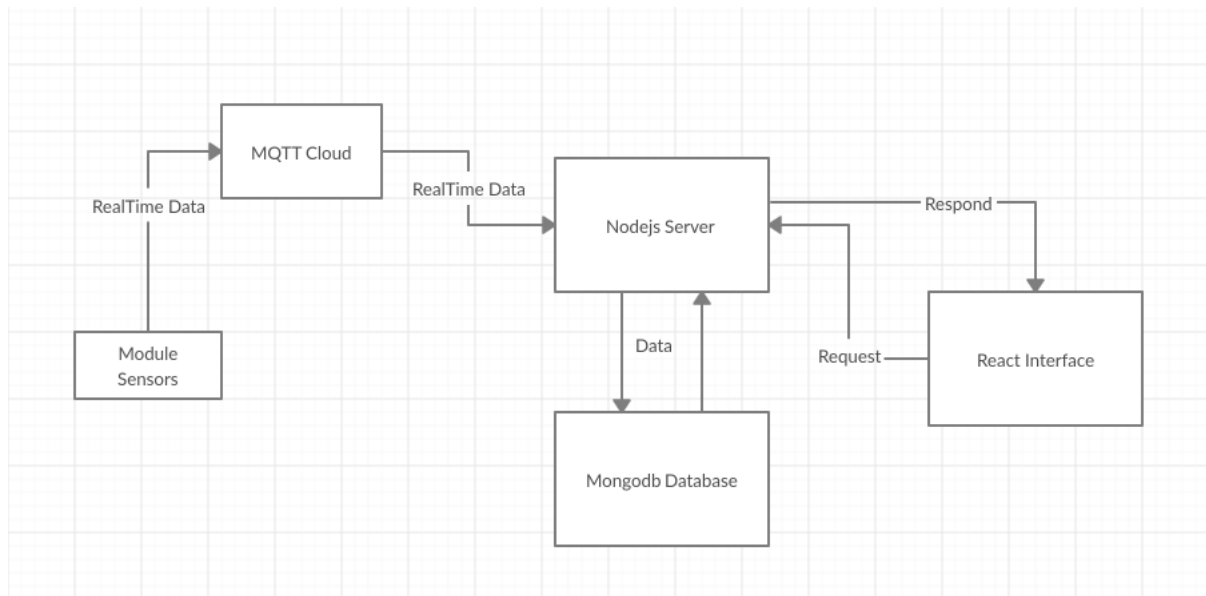
## Chương 4

# Phân tích và Thiết kế hệ thống

### 4.1 Tổng quan

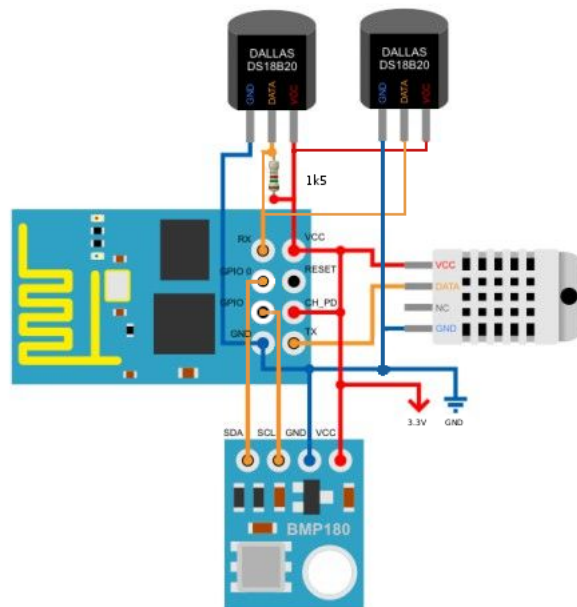
Trong phạm vi đề tài, bài toán đặt ra là giám sát sự thay đổi các chỉ số môi trường và cảnh báo tình trạng biến động của các chỉ số đó trên ứng dụng di động và ứng dụng web của người dùng. Như vậy, đầu vào của hệ thống là dữ liệu do các sensor cung cấp thông qua MQTT, dữ liệu do gửi lên sẽ được xử lý trên máy chủ, sau đó trả về được xử lý, lưu trữ dưới cơ sở dữ liệu và hiển thị thông qua giao diện ứng. Từ đó, xác định được hệ thống với những thành phần chính. Hệ thống được gồm có 4 phần chính như sau:

- Hệ thống thiết bị sensor tại các điểm khảo sát
- Back-End Server (ứng dụng trên máy chủ)
- Cơ sở dữ liệu lưu trữ (Database)
- Front-End Client (ứng dụng trên trang web)



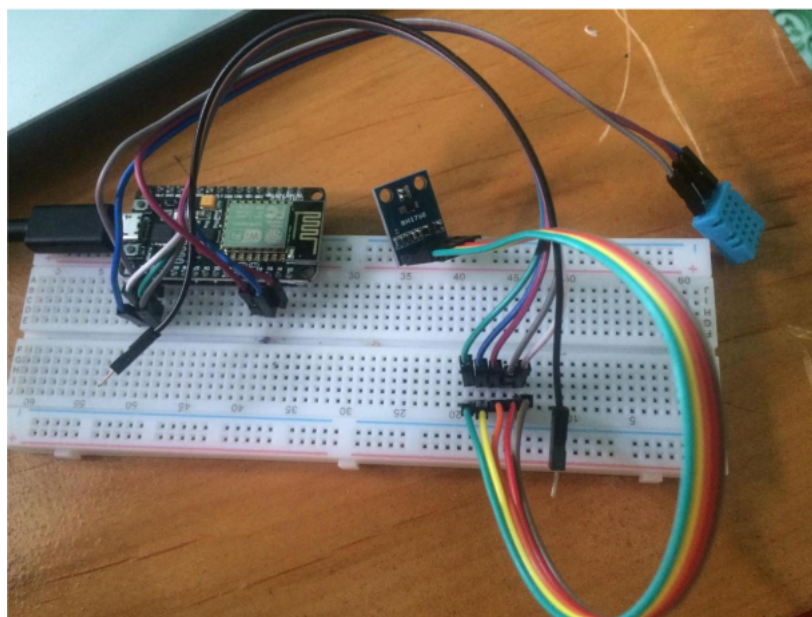
HÌNH 4.1: Mô hình tổng quan của hệ thống

### 4.1.1 Hệ thống thiết bị sensor



HÌNH 4.2: Sơ đồ thiết kế phần cứng

Trong sơ đồ nguyên lý này gồm có 3 linh kiện chính đó là ESP8266, DHT11 và BH1750.



HÌNH 4.3: Hiện thực phần cứng sensor

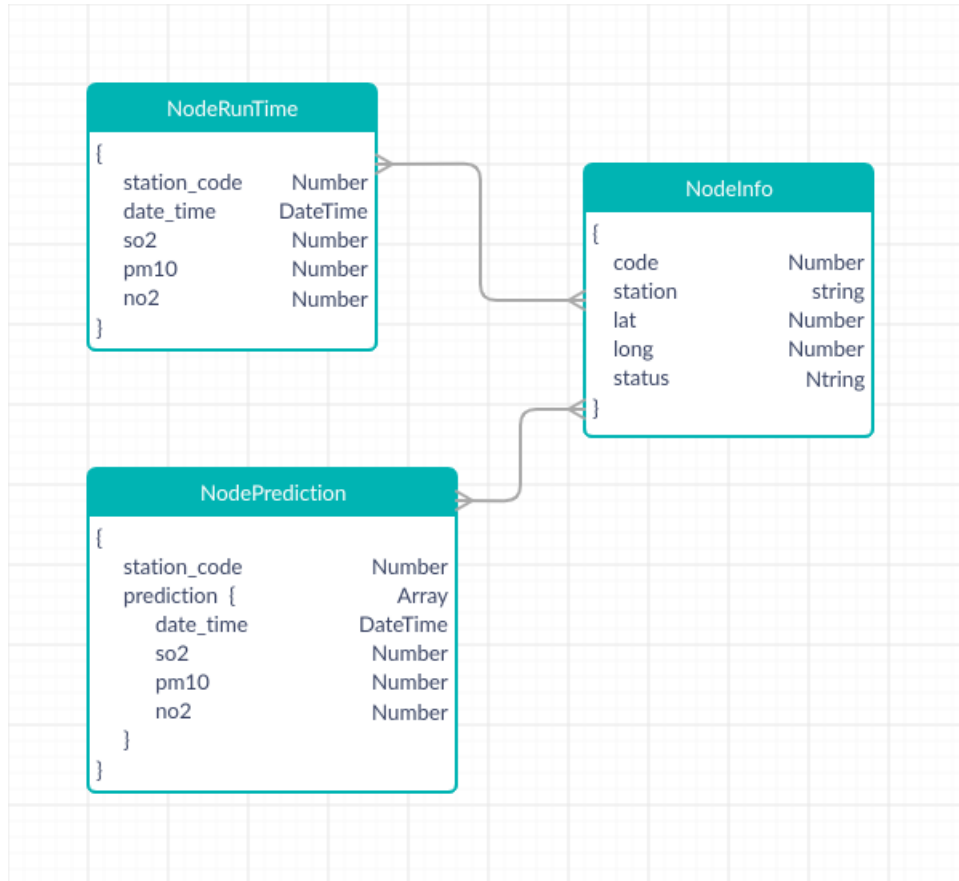
### 4.1.2 Server

Server đóng vai trò là phần lõi của cả hệ thống, là server cung cấp các API để ứng dụng có thể giao tiếp với Front-End, nhận dữ liệu MQTT để xử lý dữ liệu tiền lưu trữ và lưu trong cơ sở dữ liệu đồng thời còn truy xuất và trả về dữ liệu mà Front-End yêu cầu từ Cơ sở dữ liệu. Server sẽ chính là cầu nối giữa ứng dụng với dữ liệu của hệ thống, cung cấp môi trường giao tiếp giữa các phần trong hệ thống.

### 4.1.3 Cơ sở dữ liệu

Cơ sở dữ liệu có nhiệm vụ lưu trữ dữ liệu của toàn bộ hệ thống, giao tiếp với Server lưu và cung cấp dữ liệu. Đây là nơi lưu trữ các thông tin của các trạm quan sát, những thông tin do sensor gửi lên cho hệ thống được xử lý và lưu trữ.





HÌNH 4.4: Lược đồ cơ sở dữ liệu của ứng dụng

- Thực thể **NodeInfo** là nơi lưu trữ các thông tin cơ bản của các trạm được quan sát, gồm các trường:
  - code: mã định danh của mỗi trạm
  - station: tên của trạm tương ứng
  - lat: thông tin kinh độ
  - long: thông tin vĩ độ
- Thực thể **NodeRuntime** lưu trữ toàn bộ dữ liệu được gửi lên của các trạm:
  - stationcode: mã định danh của trạm
  - datetime: thời gian gửi lên
  - so2: thông tin về chỉ số  $SO_2$
  - pm10: thông tin về chỉ số PM10

- no2: thông tin về chỉ số  $NO_2$
- Thực thể **NodePrediction** thông tin dự đoán về các chỉ số sẽ được lưu trữ ở đây
  - stationcode: mã định danh của trạm
  - prediction: lưu trữ dưới dạng mảng các thông tin dự đoán các phần tử được tổ chức giống như ở NodeRuntime.

#### 4.1.4 Front-End Client

Front-End trên ứng dụng Web thể hiện phải thể hiện tình trạng không khí tại các trạm đó ở thời điểm hiện tại. Thông tin chi tiết về mỗi trạm đo và thông tin dự đoán trong vòng các giờ tiếp theo.

## 4.2 Áp dụng Recurrent Neural Network vào bài toán giám sát

### 4.2.1 Chuẩn bị dữ liệu

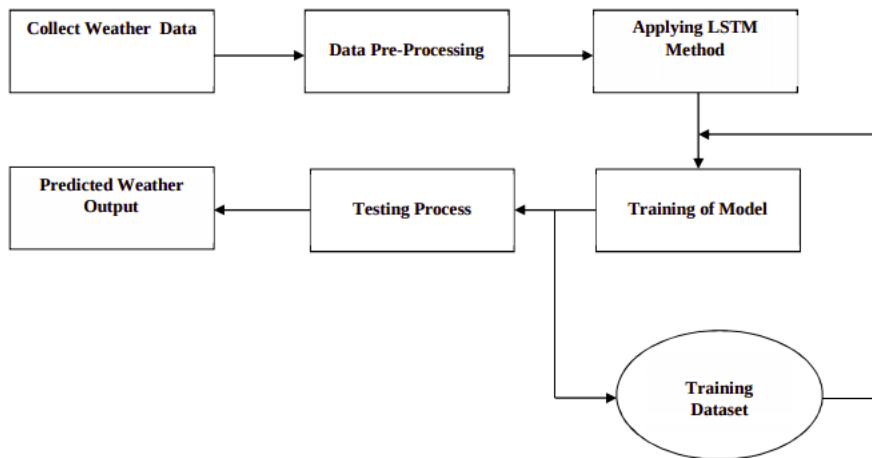
Dữ liệu là một yếu tố bắt buộc trong tất cả các bài toán Machine Learning. Để thực hiện xây dựng model giám sát ta cần dữ liệu tin cậy để cho máy có thể học được, ở đây em dùng tập dữ liệu về chỉ số ô nhiễm không khí của các chất ở Tây Ban Nha, từ năm 2001 đến 2005

| date                | CO                  | NO_2               | O_3              | PM10              | SO_2               | station  |
|---------------------|---------------------|--------------------|------------------|-------------------|--------------------|----------|
| 2003-03-01 01:00:00 | 1.72000002861023    | 73.9000015258789#  | 10.5500001907349 | 55.2099990844727  | 24.2999992370605   | 28079001 |
| 2003-03-01 01:00:00 | 1.45000004768372    | 72.1100006103516#1 | 6.71999979019165 | 52.3899993896484  | 14.2299995422363#  | 28079035 |
| 2003-03-01 01:00:00 | 1.57000005245209    | 80.5599975585938#  | 21.0499992370605 | 63.2400016784668  | 17.8799991607666   | 28079003 |
| 2003-03-01 01:00:00 | 2.45000004768372    | 78.370002746582#   | 4.21999979019165 | 67.8399963378906  | 24.8999996185303   | 28079004 |
| 2003-03-01 01:00:00 | 3.25999999046326    | 96.25#             | 8.46000003814697 | 95.7799987792969  | 18.75              | 28079039 |
| 2003-03-01 01:00:00 | 1.94000005722046##0 | 90.3000030517578## | 9.94999980926514 | 95.1500015258789# | 29.2700004577637## | 28079006 |
| 2003-03-01 01:00:00 | 1.37999999523163    | 89.5800018310547#  | 7.19999980926514 | 54                | 23.7099990844727#  | 28079007 |
| 2003-03-01 01:00:00 | 1.58000004291534    | 93.6399993896484#  | 4.19000005722046 | 26.6200008392334  | 17.7399997711182#  | 28079008 |
| 2003-03-01 01:00:00 |                     |                    |                  |                   |                    | 28079009 |
| 2003-03-01 01:00:00 | 1.91999995708466    | 71.8399963378906#  | 5.32999992370606 | 39.3600006103516  | 21.6399993896484   | 28079038 |
| 2003-03-01 01:00:00 | 1.33000004291534    | 87.9199981689453#  | 4.25             | 52.2700004577637  | 24.7000007629395#  | 28079011 |
| 2003-03-01 01:00:00 | 2.1800000667572     | 89.8499984741211#  | 8.40999984741211 | 78.0599975585938  | 34                 | 28079012 |
| 2003-03-01 01:00:00 | 1.13999998569489    | 73.870002746582#   | 7.21999979019165 | 63.2700004577637  | 7.8899998664856#   | 28079040 |
| 2003-03-01 01:00:00 | 4.67999982833862    | 42.189998626709#   | 7.57999992370606 | 103.5             | 16.5               | 28079014 |
| 2003-03-01 01:00:00 | 1.44000005722046#   | 91.0100021362305#  | 1.19000005722046 | 50.5400009155273  | 20.0400009155273## | 28079015 |
| 2003-03-01 01:00:00 | 1.25                | 70.5699996948242#  | 9.68000030517578 | 67.9499969482422  | 14.8299999237061   | 28079016 |
| 2003-03-01 01:00:00 | 1.63999998569489    | 45.4700012207031#  | 6.19000005722046 | 81.9700012207031  | 11.9799995422363   | 28079017 |
| 2003-03-01 01:00:00 | 1.85000002384186    | 94.5100021362305#  | 7.51000022888184 | 52                | 16.8500003814697#  | 28079018 |
| 2003-03-01 01:00:00 | 1.74000000953674    | 62.2599983215332#  | 2.64000010490417 | 58.0299987792969  | 17.9099998474121   | 28079019 |
| 2003-03-01 01:00:00 | 1.53999996185303    | 73.2399978637695#  | 2.67000007629395 | 67.6900024414063  | 20.7900009155273   | 28079036 |
| 2003-03-01 01:00:00 | 1.73000001907349    | 6.90000009536743#  | 7.76999998092651 | 72.3099975585938  | 26.1900005340576   | 28079021 |
| 2003-03-01 01:00:00 | 1.75                | 88.4800033569336#  | 7.53000020980835 | 63.5999984741211  | 23.3899993896484#  | 28079022 |
| 2003-03-01 01:00:00 | 1.32000005245209#   | 84.8499984741211#  | 5.42999982833862 | 75.0100021362305  | 13.1999998092651## | 28079023 |
| 2003-03-01 01:00:00 | 1.26999998092651##0 | 54.25##            | 6.53999996185303 | 53.0099983215332# | 8.80000019073486## | 28079024 |
| 2003-03-01 01:00:00 | 0.959999978542328## | 89.3300018310547## | 6.63000011444092 | 69.629997253418#  | 13.5299997329712#  | 28079025 |
| 2003-03-01 01:00:00 |                     | 95.120002746582#   | 7.71999979019165 | 61.4799995422363  | 9.6899995803833#   | 28079026 |

HÌNH 4.5: Mẫu dữ liệu tại Madrid, Tây Ban Nha

## 4.2.2 Xử lý dữ liệu

Việc vẽ đồ thị rất quan trọng đối với các bài toán thuộc dạng Time Series Analysis như thế này. Vì dĩ nhiên việc đoán trước không thể trả về kết quả chính xác 100



HÌNH 4.6: Quy trình áp dụng học máy

## Chương 5

# Kết quả và đánh giá

Đánh giá đạt được là một khâu vô cùng quan trọng. Việc đánh giá phải khách quan và phải dựa vào những tiêu chuẩn đánh giá phù hợp với hệ thống đang xây dựng. Trong chương này, báo cáo sẽ đưa ra những tiêu chuẩn đánh giá, đồng thời thực hiện đánh giá mô hình và phân tích kết quả của những thử nghiệm mà đã thực hiện.

## Chương 6

# Tổng kết và hướng phát triển

Quá trình thực hiện đề tài trải qua nhiều giai đoạn. Việc triển khai thực hiện cơ bản đúng với nội dung nguyên cứu. Kết quả thu được một hệ thống giám sát và cảnh báo chất lượng không khí nhưng vẫn còn ở mức đơ sơ giản. Do không có đủ thời gian để nguyên cứu hệ thống với dữ liệu lớn, đa dạng, chưa đi sâu vào giải thuật dự đoán các chỉ số và độ chính xác của việc chạy giải thuật chưa được cao. Trong tương lai, cần mở rộng đề tài với việc giám sát các chỉ số cấp thiết mang tính thời sự hơn, nguyên cứu sự tương tác, tác động lẫn nhau giữa các chỉ số để hệ thống cảnh báo tốt hơn. Hệ thống cần được xây dựng để đáp ứng với số lượng sensor lớn và việc dự đoán ở các chỉ số ở tương lai chuẩn xác hơn.

## Tài liệu tham khảo

- [1] Tổng quan về Recurrent Neural Network  
<https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-recurrent-neural-networks>
- [2] Advanced Weather Forecasting Prediction using Deep Learning  
<https://www.ijraset.com/files/serve.php?FID=24817>
- [3] Quản lý chất lượng không khí.  
[https://www.sutp.org/files/contents/documents/resources/A\\_Sourcebook/SB5\\_Environment%20and%20Health/GIZ\\_SUTP\\_SB5a\\_Air-Quality-Management\\_VN.pdf](https://www.sutp.org/files/contents/documents/resources/A_Sourcebook/SB5_Environment%20and%20Health/GIZ_SUTP_SB5a_Air-Quality-Management_VN.pdf)