

NGÀNH CÔNG NGHỆ THÔNG TIN



BÁO CÁO ĐỒ ÁN I

TENSORGRAM

PHẦN MỀM TẠO DIAGRAM

CHO MÔ HÌNH TENSORFLOW

|  |  |
| --- | --- |
| Nhóm sinh viên thực hiện: | |
| Huỳnh Quốc Hoàng Vương | 17110256 |
| Nguyễn Minh Trung  Việt | 17110xxx  17110xxx |

GVHD: TS. Huỳnh Xuân Phụng

Tp. Hồ Chí Minh, tháng 11 – 2019

ĐIỂM SỐ

|  |  |  |  |
| --- | --- | --- | --- |
| TIÊU CHÍ | NỘI DUNG | TRÌNH BÀY | TỔNG |
| ĐIỂM |  |  |  |

NHẬN XÉT CỦA GIÁO VIÊN HƯỚNG DẪN

Giáo viên hướng dẫn

(*ký và ghi họ tên*)

Huỳnh Xuân Phụng

ĐIỂM SỐ

|  |  |  |  |
| --- | --- | --- | --- |
| TIÊU CHÍ | NỘI DUNG | TRÌNH BÀY | TỔNG |
| ĐIỂM |  |  |  |

NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

Giáo viên phản biện

(*ký và ghi họ tên*)

LỜI CẢM ƠN

Để hoàn thành tốt đề tài và bài báo cáo này, chúng em xin gửi lời cảm ơn chân thành đến giảng viên, tiến sĩ Huỳnh Xuân Phụng, người đã trực tiếp hỗ trợ chúng em trong suốt quá trình làm đề tài. Chúng em cảm thầy đã đưa ra những lời khuyên từ kinh nghiệm thực tiễn của mình để định hướng cho chúng em đi đúng với yêu cầu của đề tài đã chọn, luôn giải đáp thắc mắc và đưa ra những góp ý, chỉnh sửa kịp thời giúp chúng em khắc phục nhược điểm và hoàn thành tốt cũng như đúng thời hạn Khoa đã đề ra.

Chúng em cũng xin gửi lời cảm ơn chân thành các quý thầy cô trong khoa Đào tạo Chất Lượng Cao nói chung và nghành Công Nghệ Thông Tin nói riêng đã tận tình truyền đạt những kiến thức cần thiết giúp chúng em có nền tảng để làm nên đề tài này, đã tạo điều kiện để chúng em có thể tìm hiểu và thực hiện tốt đề tài. Cùng với đó, chúng em xin được gửi cảm ơn đến các bạn cùng khóa đã cung cấp nhiều thông tin và kiến thức hữu ích giúp chúng em có thể hoàn thiện hơn đề tài của mình.

Đề tài và bài báo cáo được chúng em thực hiện trong khoảng thời gian ngắn, với những kiến thức còn hạn chế cùng nhiều hạn chế khác về mặt kĩ thuật và kinh nghiệm trong việc thực hiện một dự án phần mềm. Do đó, trong quá trình làm nên đề tài có những thiếu sót là điều không thể tránh khỏi nên chúng em rất mong nhận được những ý kiến đóng góp quý báo của các quý thầy cô để kiến thức của chúng em được hoàn thiện hơn và chúng em có thể làm tốt hơn nữa trong những lần sau. Chúng em xin chân thành cảm ơn.

Cuối lời, chúng em kính chúc quý thầy, quý cô luôn dồi dào sức khỏe và thành công hơn nữa trong sự nghiệp trồng người. Một lần nữa chúng em xin chân thành cảm ơn.

TP.HCM, ngày 10 tháng 12 năm 2018

Nhóm sinh viên thực hiện

MỤC LỤC

[DANH MỤC CÁC HÌNH 1](#_Toc22330169)

[DANH MỤC CÁC BẢNG 2](#_Toc22330170)

[CHƯƠNG 1: TỔNG QUAN CHƯƠNG TRÌNH 3](#_Toc22330171)

[*1.* *Giới thiệu chung* 3](#_Toc22330172)

[1.1. Machine Learning, Tensorflow và Layers API 3](#_Toc22330173)

[1.1.1. Lí thuyết Machine Learning cơ bản 3](#_Toc22330174)

[1.1.2. Thư viện ML Tensorflow 3](#_Toc22330175)

[1.1.3. Layers API của Tensorflow 3](#_Toc22330176)

[*2.* *Đặc tả phần mềm TensorGram* 3](#_Toc22330177)

[2.1. Phần mềm TensorGram 3](#_Toc22330178)

[2.1.1. Giới thiệu về phần mềm TensorGram 3](#_Toc22330179)

[2.1.2. Use Case Diagram 3](#_Toc22330180)

[2.1.3. Tính năng chính 3](#_Toc22330181)

[2.1.4. Ứng dụng 3](#_Toc22330182)

[2.2. Giao diện dự kiến 3](#_Toc22330183)

[2.3. Yêu cầu kĩ thuật 3](#_Toc22330184)

[2.4. Mục tiêu thực hiện 3](#_Toc22330185)

[CHƯƠNG 2: KẾ HOẠCH THỰC HIỆN 4](#_Toc22330186)

[*1.* *Kế hoạch* 4](#_Toc22330187)

[*2.* *Phân công công việc* 4](#_Toc22330188)

[CHƯƠNG 3: THIẾT KẾ PHẦN MỀM 5](#_Toc22330189)

[*1.* *Thiết kế lớp* 5](#_Toc22330190)

[1.1. Thiết kế lớp cho các Layer của TensorFlow Layers API 5](#_Toc22330191)

[1.1.1. Tổng quan 5](#_Toc22330192)

[1.1.2. Thiết kế 6](#_Toc22330193)

[1.1.3. Đặc tả lớp 7](#_Toc22330194)

[1.1.4. Đặc tả các phương thức trong lớp 8](#_Toc22330195)

[*1.1.4.1.* *Lớp Layer* 8](#_Toc22330196)

[*2.* *Thuật toán* 8](#_Toc22330197)

[*3.* *Thiết kế giao diện* 9](#_Toc22330198)

[CHƯƠNG 4: CÀI ĐẶT VÀ KIỂM THỬ 9](#_Toc22330199)

[CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN 10](#_Toc22330200)

[*1.* *Kết luận* 10](#_Toc22330201)

[*2.* *Hướng phát triển* 10](#_Toc22330202)

[TÀI LIỆU THAM KHẢO 11](#_Toc22330203)

# DANH MỤC CÁC HÌNH

# DANH MỤC CÁC BẢNG

CHƯƠNG 1: TỔNG QUAN CHƯƠNG TRÌNH

1. *Giới thiệu chung*
   1. Machine Learning, Tensorflow và Layers API
      1. Lí thuyết Machine Learning cơ bản
      2. Thư viện ML Tensorflow
      3. Layers API của Tensorflow
2. *Đặc tả phần mềm TensorGram*
   1. Phần mềm TensorGram
      1. Giới thiệu về phần mềm TensorGram
      2. Use Case Diagram
      3. Tính năng chính
      4. Ứng dụng
   2. Giao diện dự kiến
   3. Yêu cầu kĩ thuật
   4. Mục tiêu thực hiện

CHƯƠNG 2: KẾ HOẠCH THỰC HIỆN

1. *Kế hoạch*

|  |  |
| --- | --- |
| Tuần | Công việc |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |
| 9 |  |
| 10 |  |
| 11 |  |

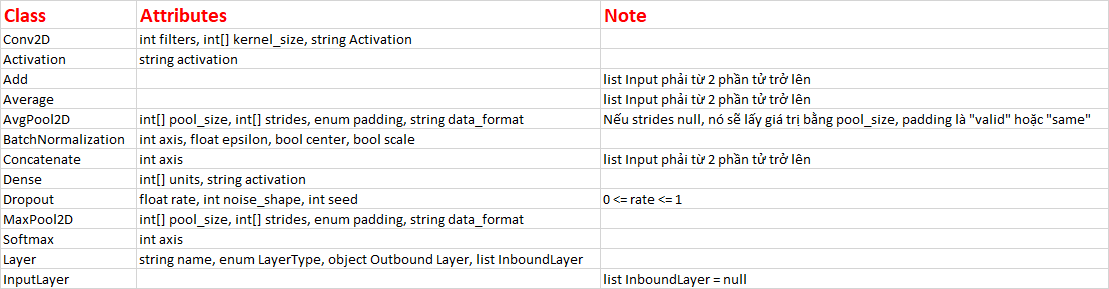
1. *Phân công công việc*

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên sinh viên | Miêu tả công việc | Đóng góp |
| 1 | Huỳnh Q.H. Vương |  |  |
| 2 | Minh Trung |  |  |
| 3 | Việt |  |  |

CHƯƠNG 3: THIẾT KẾ PHẦN MỀM

1. *Thiết kế lớp*
   1. Thiết kế lớp cho các Layer của TensorFlow Layers API
      1. Tổng quan

Xem xét trong phạm vi phần mềm sẽ xây dựng, ngoài các class đặc thù phục vụ cho các chức năng và sự vận hành của chương trình, do sự hạng chế về thời gian và kiến thức của chúng em, phần mềm này chỉ cài đặt và hiện thực hoá 13 trong tổng số 203 Layer trong Layers API, cũng như chúng em sẽ chỉ cài đặt cho các layer này các thuộc tính quan trọng chứ không cài đặt toàn bộ tất cả thuộc tính cho chúng.

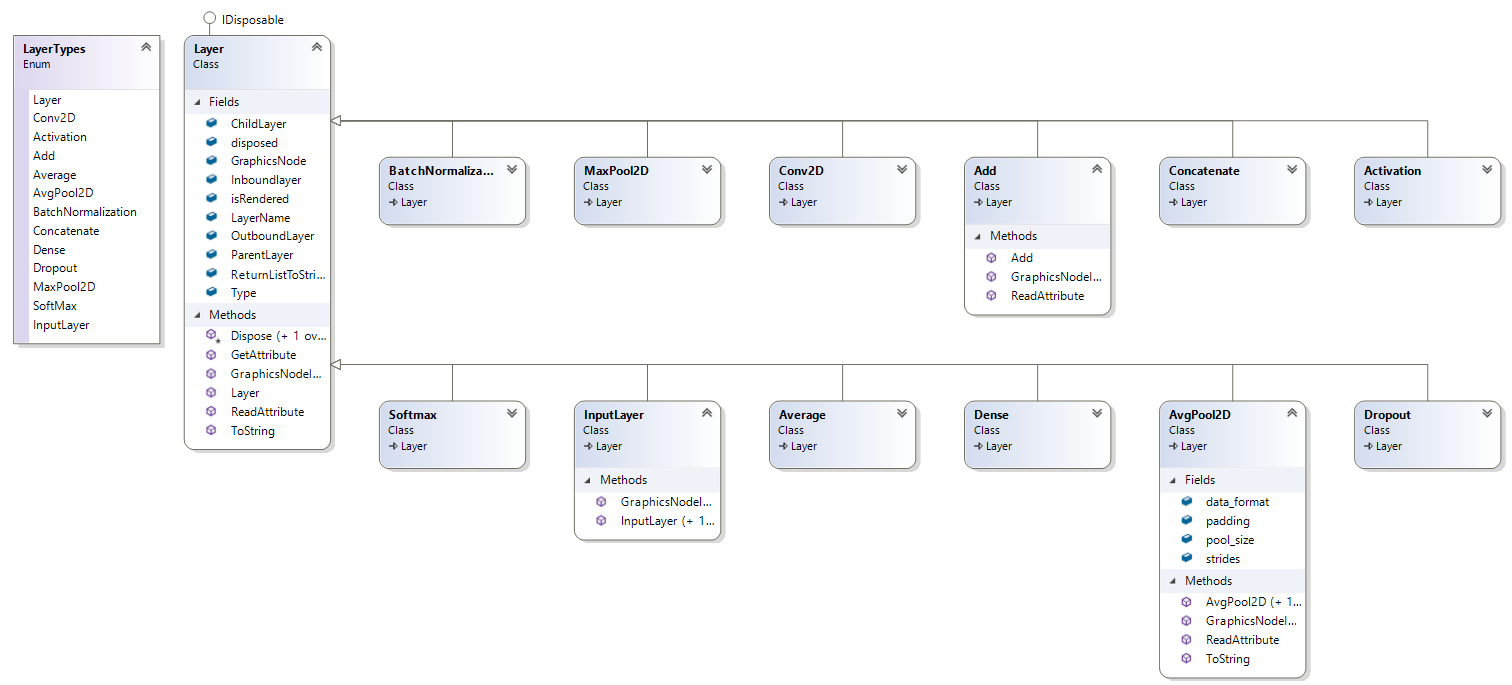


*Hình 3: Các Class và Attributes ứng với các layer.*

|  |  |  |
| --- | --- | --- |
| TT | Tên lớp | Chức năng |
| 1 | Layer | Base layer, là dạng chung của tất cả các class của Keras Layers trong mô hình TensorFlow. |
| 2 | InputLayer | Dùng làm điểm vào cho mạng (Biểu đồ các Layer). |
| 3 | Conv2D | Tạo ra một *convolution kernel,* kết hợp với Input của Layer, tạo ra một Tensor đầu ra. |
| 4 | Activation | Áp dụng hàm activation vào một Output. |
| 5 | Add | Thực hiện phép cộng tất cả các phần tử đồng dạng trong Input, cho ra Output duy nhất 1 phần tử(Đồng dạng với Input) |
| 6 | Average | Thực hiện phép trung bình tất cả các phần tử đồng dạng trong Input, cho ra Output duy nhất 1 phần tử(Đồng dạng với Input) |
| 7 | AvgPool2D | Thực hiện phép trung bình gộp lên dữ liệu Spatial |
| 8 | BatchNormalization |  |
| 9 | Concatenate |  |
| 10 | Dense |  |
| 11 | Propout |  |
| 12 | MaxPool2D |  |
| 13 | Softmax |  |

*Bảng 2: Chi tiết chức năng các layer trong TensorFlow Model*

* + 1. Thiết kế



*Hình 4 : UML Diagram biểu diễn các lớp sẽ xây dựng trong phần mềm.*

Xét thấy giữa các layer có các tính chất chung, bao gồm:

* + Tên layer.
  + Loại layer.
  + Input layer.
  + Output layer.

Đó là chưa kể giữa chúng còn có các methods và chứa các đối tượng giống nhau (ví dụ như method đọc input, đối tượng đồ hoạ, overdrive method .ToString(), …). Vì thế, để tận dụng sức mạnh của lập trình hướng đối tượng, chúng em quyết định thiết kế class Layer, class này là base cho các class khác, chứa tất cả những phương thức, thuộc tính chung của các layer. Đồng thời, các class của các layer khác kế thừa nó, bổ sung các Method, đối tượng của riêng nó để đáp ứng chức năng, nhiệm vụ của layer mà class đó hiện thực.

* + 1. Đặc tả lớp

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Tên lớp | Mục đích | SV phụ trách |
| 1 | Layer  Kế thừa từ: None | Base class, đối tượng hoá Layer *Layer* trong TensorFlow Model. Chứa tất cả những phương thức, thuộc tính chung của các Layer, cho các class khác sử dụng. | Hoàng Vương |
| 2 | InputLayer  Kế thừa từ: Layer | Đối tượng hoá Layer *InputLayer* trong TensorFlow Model. | Hoàng Vương |
| 3 | Conv2D  Kế thừa từ: Layer | Đối tượng hoá Layer *Conv2D* trong TensorFlow Model. | Hoàng Vương |
| 4 | Activation  Kế thừa từ: Layer | Đối tượng hoá Layer *Activation* trong TensorFlow Model. |  |
| 5 | Add  Kế thừa từ: Layer | Đối tượng hoá Layer *Add* trong TensorFlow Model. | Hoàng Vương |
| 6 | Average  Kế thừa từ: Layer | Đối tượng hoá Layer *Average* trong TensorFlow Model. |  |
| 7 | AvgPool2D  Kế thừa từ: Layer | Đối tượng hoá Layer *AveragePooling2D* trong TensorFlow Model. | Hoàng Vương |
| 8 | BatchNormalization  Kế thừa từ: Layer | Đối tượng hoá Layer *BatchNormalization* trong TensorFlow Model. |  |
| 9 | Concatenate  Kế thừa từ: Layer | Đối tượng hoá Layer *Concatenate* trong TensorFlow Model. |  |
| 10 | Dense  Kế thừa từ: Layer | Đối tượng hoá Layer *Dense* trong TensorFlow Model. |  |
| 11 | Propout  Kế thừa từ: Layer | Đối tượng hoá Core *Layer* Propout trong TensorFlow Model. |  |
| 12 | MaxPool2D  Kế thừa từ: Layer | Đối tượng hoá Layer *MaxPool2D* trong TensorFlow Model. |  |
| 13 | Softmax  Kế thừa từ: Layer | Đối tượng hoá Layer *Softmax* trong TensorFlow Model. |  |

*Bảng 3: Danh mục các lớp*

* + 1. Đặc tả các phương thức trong lớp
       1. *Lớp Layer*

|  |  |  |  |
| --- | --- | --- | --- |
| TT | Phương thức | Mục đích | Tên file, stt dòng khai báo |
| 1 | virtual ReadAttribute(string \_input)  input: \_input  output: None | Phương thức virtual, dùng để đọc plain text đã qua xử lí từ input của user thành dữ liệu, đưa vào các attributes của layer. | Layers/Topology/Layer.cs (73) |
| 2 | GetAttribute()  Input: None  output: List<string> | Phương thức virtual, dùng để xuất ra tất cả các thông tin về attributes của layer dưới dạng List<string> , mỗi phần tử trong list chưa tên và giá trị của nó. | Layers/Topology/Layer.cs (84) |
| 3 | ToString()  Input: None  output: List<string> | Phương thức virtual, dùng để xuất ra tất cả các thông tin về layer dưới dạng List<string> , mỗi phần tử trong list chưa tên và giá trị của nó. | Layers/Topology/Layer.cs (89) |
| 4 | GraphicsNodeInitialize()  Input: None  output: None | Phương thức virtual, khởi tạo đối tượng đồ hoạ cho layer với tên của đối tượng bằng với tên lớp đọc từ input người dùng | Layers/Topology/Layer.cs (79) |
| 5 | Layer()  Input: None  output: None | Khới tạo một class Layer mới, tất cả các attribute và properties cơ bản được khởi tạo về Null | Layers/Topology/Layer.cs (40) |

* 1. Thiết kế lớp chức năng

1. *Thuật toán*
2. *Thiết kế giao diện*

CHƯƠNG 4: CÀI ĐẶT VÀ KIỂM THỬ

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

1. *Kết luận*
2. *Hướng phát triển*

TÀI LIỆU THAM KHẢO