



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS

Proyecto 3

Identificador de dígitos

Integrantes

Aldo Iván García Salman
418004788

Jorge Antonio Ruíz López
315119729

Asignatura

Inteligencia Artificial

Índice

1. Introducción	3
2. Delimitación y planteamiento del problema de investigación	4
3. Justificación	5
4. Objetivos	6
5. Marco Teorico	7
6. Metodología	8
7. Conclusiones	10
8. Referencias bibliográficas	11

1. Introducción

Como vimos, existen distintos tipos de agentes dentro de la inteligencia artificial, en este caso nos enfocaremos especialmente en los agentes inteligentes, los cuales son capaces de adquirir conocimiento y desempeñarse con una mayor eficacia en las siguientes iteraciones de la tarea a la que está dedicado o para la que fue diseñado.

La motivación del tema principal sobre este proyecto viene de tratar de entender un poco más sobre que es una red neuronal, sus capacidades y su funcionamiento.

A lo largo de este proyecto se intentará desarrollar un analizador/identificador de dígitos, mediante los datos obtenidos por una red neuronal entrenada con un conjunto de datos. Para continuar debemos recordar primero que una red neuronal es un modelo que nos permite, dado un cierto conjunto de parámetros y una permutación de este conjunto (una forma de combinarlos), predecir un posible resultado correcto.

2. Delimitación y planteamiento del problema de investigación

El programa intentará reconocer el dígito dibujado por el usuario. Para poder realizar esta acción, vamos a hacer uso de una red neuronal convolucional, la cual será entrenada previamente con un conjunto de datos sobre representación de dígitos mediante gráficas. Al ser números dibujados por el usuario el problema del reconocimiento es mayor pues cada número puede ser representado de una forma distinta por diferentes usuarios, además que el mismo usuario no siempre realizará los mismos dígitos de la misma forma.

Primero tenemos las Neuronas Convolucionales, las cuales se encargan de la extracción de datos de una imagen 2D en forma de matriz y la salida de cada una es la combinación lineal de la salida de cada una de las neuronas operadas con el operador de convolución, el cual se encarga de filtrar la imagen con un núcleo previamente entrenado. Al final sumaremos valor definido que representa un "premio" todo esto será pasado por la función de activación.

Se realizan distintas fases de extracción de características los datos recaudados llegarán a una fase de clasificación. Para poder llegar a esta sección los datos son depurados con una serie de características únicas para la imagen o la clase que se tiene como entrada y después las neuronas (de clasificación) intentan clasificar estas características en una clase u otra.

La salida de estas neuronas está determinada por la combinación lineal de las salidas de cada una de las neuronas de la capa anterior multiplicadas por un peso (visto como una gráfica este puede ser el costo de la conexión). A esto le sumaremos un parámetro que juega una especie de "premio" al final pasará por una función de activación definida dependiendo el problema.

No olvidemos que, mientras mayor entrenamiento tenga la red, mejor deberá ser su desempeño en la tarea dada.

Como hipótesis se tiene que el agente (el programa) intentará adivinar/reconocer el número dibujado con un 30 % mínimo de efectividad.

3. Justificación

El análisis de este problema es de alta importancia dentro del campo de la inteligencia artificial pues tiene una importancia social y academica muy relevante.

Primero, podemos verlo como un agente inteligente. El análisis de estos agentes es muy importante porque son capaces de adquirir conocimiento de las tareas previamente realizadas o de las características de su entorno.

La aplicación resulta muy importante pues a lo largo de la realización de este proyecto se intenta acertar en el número que ingresó el usuario y ese es su uso específico. Pero a este tipo de agentes podemos verles distintas aplicaciones muy importantes y muy relacionadas a el realizado en este, como lo pueden ser el reconocimiento facial, la búsqueda de distintos simbolos y más.

Como apoyo en la realización de este proyecto utilizaremos algunas bibliotecas previamente definidas en python como lo es keras, la cual nos brindara el primer conjunto de datos con el que entrenaremos la red para despues poder obtener un modelo a entregar y que sea dicho modelo el que carguemos en el programa para el reconocimiento de los digitos.

4. Objetivos

El objetivo general de la realización de este trabajo es comprender más sobre los agentes inteligentes, su desempeño y sus aplicaciones reales. Para eso, éste agente busca dar como respuesta el número más parecido a la imagen que recibe como entrada (el número dibujado por el usuario) y lo hace con ayuda de redes neuronales convolucionales.

Específicamente el agente intentará clasificar la imagen que recibe como entrada (una matriz) en alguna de las 10 clases (los números 0,1,2,...9).

Se busca que el porcentaje mínimo de efectividad sea de 80% sobre los números acertados, pero una efectividad general de un mínimo del 30%.

5. Marco Teorico

Inicialmente obtendremos un conjunto de datos ya definidos por la biblioteca keras de python `mnist.load_data()` lo cual nos brinda un conjunto de datos de 60,000 imagentes en escala de grises de 28x28 píxeles, sobre los 10 dígitos, además de un conjunto extra de 10,000 imagenes que serán usadas para el entrenamiento de la red.

Esto es, porque una red neuronal necesita tener un conjunto de datos de entrenamiento para que el agente pueda "aprender" y después del entrenamiento se utilizará el conjunto de prueba para ver el desempeño de la red.

El entrenamiento de una red neuronal se realiza modificando los pesos de sus neuronas para extraer los resultados que buscamos. Para lograrlo lo que hacemos es introducir los datos del conjunto de entrenamiento en la red y en función del resultado que se obtenga, se modificaran los pesos de cada una de las neuronas de la red según el error obtenido y en función de cuanto haya contribuido cada neurona a dicho resultado.

Con ese procedimiento, el agente es capaz de aprender con los datos que reciba como entrada.

6. Metodología

Vamos a describir este agente inteligente mediante sus REAS:

- Tipo de agente: Agente Inteligente, pues trata de modelar las propiedades de un sistema del entendimiento humano. El cual consta de una entrada, la estructura y una salida.
- Rendimiento: Obtiene una respuesta mediante la clasificación de los datos de entrada.
- Entorno: El conjunto de datos de entrenamiento, el conjunto de datos de prueba y la entrada del usuario.
- Actuadores: La red neuronal completa, pues cada una de las neuronas con la que está formada son las encargadas de extraer las características de los datos y las mismas neuronas se encargan de clasificar los datos de la entrada para obtener una salida.
- Sensores: Las mismas neuronas son aquellas que actúan como sensores dentro de la red

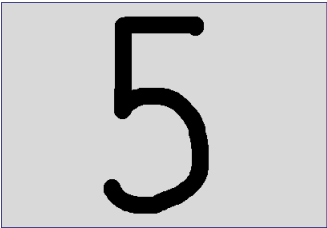
Propiedades del entorno de trabajo:

- Entorno de trabajo: Conjunto de datos de entrenamiento, prueba y la entrada del usuario.
- Observable: Es un entorno Completamente Observable pues podemos acceder al conjunto completo de datos de entrenamiento y prueba, además de la entrada del usuario
- Estocástico: Pues el estado siguiente no está determinado por estado actual ni la acción ejecutada por el agente en este estado.
- Episódico: Pues la experiencia del agente si está dividida en episodios atómicos, pues cumple que cada episodio consiste en la percepción del agente (el dígito de la entrada actual) y la realización de una única acción posterior (el dígito predecido).
- Discreto: Pues jamás se trabaja con un número infinito de elementos en el conjunto de datos.
- Estático: Pues el ambiente no cambia mientras el agente está deliberando..
- Multiagente: Pues hay varios agentes (las neuronas) cooperando o compitiendo para realizar una tarea en común.

Ejemplo de ejecuciones

Analizador de Dígitos

Dibuja un Dígito



El dígito predicho es: 5

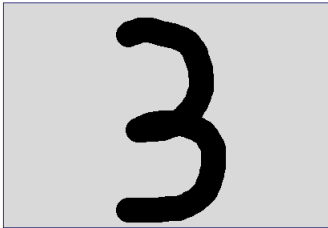
Porcentaje de exactitud: 98%

Predecir

Borrar

Analizador de Dígitos

Dibuja un Dígito



El dígito predicho es: 3

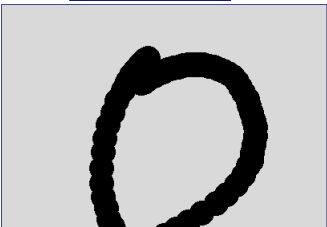
Porcentaje de exactitud: 40%

Predecir

Borrar

Analizador de Dígitos

Dibuja un Dígito



El dígito predicho es: 0

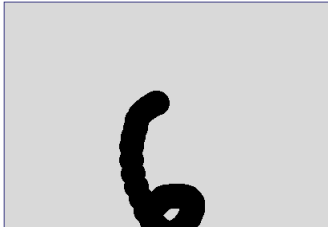
Porcentaje de exactitud: 45%

Predecir

Borrar

Analizador de Dígitos

Dibuja un Dígito



El dígito predicho es: 1

Porcentaje de exactitud: 56%

Predecir

Borrar

7. Conclusiones

Como se vio a lo largo del documento y del trabajo el uso de las redes neuronales es importante pues son agentes que no solo pueden dar una determinada respuesta en el momento como un programa común, sino que son capaces de aprender de su entorno y mejorar su rendimiento en posibles tareas futuras (claramente de su tipo específico para el que fueron diseñadas)

Aunque son agentes capaces de aprender de su entorno, también debemos tener en cuenta que se está tratando de reconocer la entrada para intentar clasificarla, lo cual no nos da una respuesta exacta del todo pues lo que se está haciendo es una especie de predicción, por lo cual podemos encontrar fallos y por más grande que sea el porcentaje de exactitud siempre debemos contemplar un rango de error.

En conclusión, aunque no son agentes 100 % eficaces podemos afirmar que son la mejor opción pues estos son capaces de aprender de su entorno y específicamente de la entrada recibida y mejorar para decisiones futuras.

8. Referencias bibliográficas

- TechVidvan. (2020). Handwritten Digit Recognition with Python & CNN. 24/01/2021, de TechVidvan Sitio web: <https://techvidvan.com/tutorials/handwritten-digit-recognition-with-python-cnn/>
- Orhan Yalçın. (2018). Image Classification in 10 Minutes with MNIST Dataset. 11/01/2021, de towards data science Sitio web: <https://towardsdatascience.com/image-classification-in-10-minutes-with-mnist-dataset-54c35b77a38d>
- Christian Versloot. (2020). How to predict new samples with your TensorFlow / Keras model?. 25/01/2021, de MachineCurve Sitio web: <https://www.machinecurve.com/index.php/2020/02/21/how-to-predict-new-samples-with-your-keras-model/>
- Abhishek Ezhava. (2019). ProgrammingKnowledge-Tkinter-Series. 20/01/2021, de ProgrammingKnowledge (YouTube) Sitio web: [https://github.com/abhishek305/ProgrammingKnowledge-Tkinter-Series/blob/master/10th/Slider %20and %20color %20choos.py](https://github.com/abhishek305/ProgrammingKnowledge-Tkinter-Series/blob/master/10th/Slider%20and%20color%20choos.py)
- <https://stackoverflow.com/questions/9886274/how-can-i-convert-canvas-content-to-an-image>
- liupeixin. (2012). How can I convert canvas content to an image?. 27/01/2021, de Stack Overflow Sitio web: <https://stackoverflow.com/questions/9886274/how-can-i-convert-canvas-content-to-an-image>
- IBM. El modelo de redes neuronales. Recuperado el 2 de febrero de https://www.ibm.com/support/knowledgecenter/es/SS3RA7_sub/modeler_mainhelp_client_ddita/components/neuralnet/neuralnet_model.html