

Trabajo Final - Machine Learning

1st Braulio Sebastián Baldeón Albornoz
Ciencias de la Computacion
Lima, Perú
U201615681@upc.edu.pe

2nd Daniel Nuñez Robinson
Ciencias de la Computacion
Lima, Perú
U201622489@upc.edu.pe

3rd Renzo Paolo Alegre Flores
Ciencias de la Computacion
Lima, Perú
U201412379@upc.edu.pe

I. RESUMEN

El presente trabajo tiene como objetivo comparar el rendimiento de diferentes algoritmos de Machine Learning al momento de clasificar imágenes. Para lo cual, se implementó siete modelos base disponibles en las librerías TensorFlow y sklearn. Con la finalidad de establecer un punto de comparación se emplearon las mismas particiones del conjuntos de datos CIFAR10 y se evaluaron los resultados en base a la métrica de desempeño accuracy.

II. INTRODUCCIÓN

La clasificación o reconocimiento de imágenes es la tarea de identificar lo que representa una imagen. En otras palabras, un modelo entrenado para clasificar imágenes es capaz de reconocer a que clase o grupo pertenece una imagen de entrada [1]. El presente trabajo tiene como objetivo emplear distintos algoritmos de Machine Learning para clasificar imágenes y comparar sus resultados. Para lo cual, se utilizó el conjunto de datos CIFAR10 provisto por la librería TensorFlow. Este conjunto de datos consta de 60000 imágenes en color de 32x32 en 10 clases.

III. FUNDAMENTO TEÓRICO

Los algoritmos de Machine Learning implementados en este trabajo son detallados a continuación:

- Red Neuronal Recurrente (RNN), son un tipo de redes neuronales diseñada para reconocer patrones en secuencias de datos. Normalmente empleadas para analizar textos, genomas, escritura, palabra hablada o series temporales numéricas. No obstante, también pueden ser aplicadas en imágenes mediante su descomposición en trozos y tratándolos como secuencias [2].
- Long short-term memory (LSTM), como afirman Chung et al. en [3], las LSTM y GRU (Gated Recurrent Units) son capas de redes neuronales recurrentes más especializadas para mantener un aprendizaje de las dependencias a largo plazo al mantener la información en su memoria por tiempos más prolongados.
- Red Neuronal Convolutiva (CNN), es una de los principales modelos para hacer reconocimiento de imágenes debido a que eliminan la necesidad de una

extracción manual de características. Es decir, las CNN aprende directamente las características de las imágenes mediante capas convolucionales.

- Perceptron Multicapa (MLP), es la estructura de red neuronal más utilizada; especialmente, la estructura de 2 capas en la que las unidades de entrada y la capa de salida están interconectadas con una capa oculta intermedia [6].
- Naive Bayes, son un conjunto de algoritmos de aprendizaje supervisados basados en la aplicación del teorema de Bayes con la suposición “ingenua” de independencia condicional entre cada par de características dado el valor de la variable de clase [7].
- Árbol de decisión, los árboles de decisión pueden ser usados para representar tanto modelos de clasificación como modelos de regresión. Cuando son empleados para clasificar tienen como propósito predecir la clase de una instancia en base a sus atributos de entrada [8].
- KMeans, el proceso que realiza, básicamente, consiste en dada una determinada agrupación inicial, reubicar cada punto en su nuevo centro más cercano, actualizar los centros de agrupación calculando la media de los puntos miembros y repita el proceso de reubicación y actualización hasta los criterios de convergencia sean satisfechos [9].

IV. MÉTODO

Como primer paso, se carga el conjunto de datos directamente desde TensorFlow. Graficar un conjunto aleatorio de imágenes para comprobar si se cargó correctamente. Para los dos primeros modelos se transformó las tres matrices de píxeles en tres vectores de 1024 píxeles cada uno; estos vectores son enviados como datos de entrada. En cambio, para el modelo CNN se emplean las mismas imágenes como datos de entrada. Por otro lado, para el resto de modelos se convirtió los tres vectores en uno solo vector de 3072 píxeles.

La implementación y validación de todos los modelos se realizaron de la siguiente manera:

1. Se crean el modelo en cuestión.
2. Se realiza el entrenamiento y validación del modelo.
3. Se grafica el accuracy de cada época.

V. EXPERIMENTOS

El desarrollo del proyecto se llevó acabo integramente en Google Colab y se utilizó el servicio de repositorio GitHub. En cuanto a la partición del conjunto de datos, se decidio emplear las particiones propuestas por TensorFlow. Es decir, se utilizó 50,000 instancias para el entrenamiento y 10,000 para la validación de los modelos. Por otro lado, las métricas de desempeño utilizadas para evaluar el rendimiento de los modelos se detallan a continuación:

- Accuracy, es larelación entre las instancias predichas correctamente y el total de instancias.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

- Precision, es la relación entre las instancias positivas predichas correctamente y el total de instancias positivas predichas.

$$Precision = \frac{TP}{TP + FP} \quad (2)$$

- Recall, es la relación entre las instancias positivas predichas correctamente y todas las instancias de la clase positiva real.

$$Recall = \frac{TP}{TP + FN} \quad (3)$$

- F1 score, es el promedio ponderado de Precision y Recall. Por lo tanto, esta puntuación tiene en cuenta tanto los falsos positivos como los falsos negativos.

$$F1score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (4)$$

VI. RESULTADO

VI-A. Comparación entre las redes neuronales

Como se puede observar en las Figuras 1, 2, 3 y 4, el modelo que obtuvo mejor accuracy tanto para el entranamiento y la validación fue la red neuronal convolucional. El cual obtuvo 76 % y 65 % de accuracy surante el entrenamiento y la validación respectivamente.

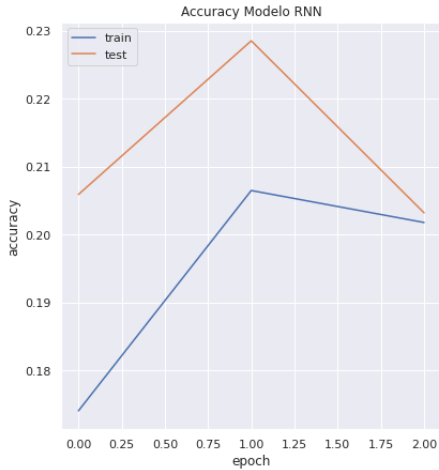


Figura 1. Accuracy por época modelo RNN

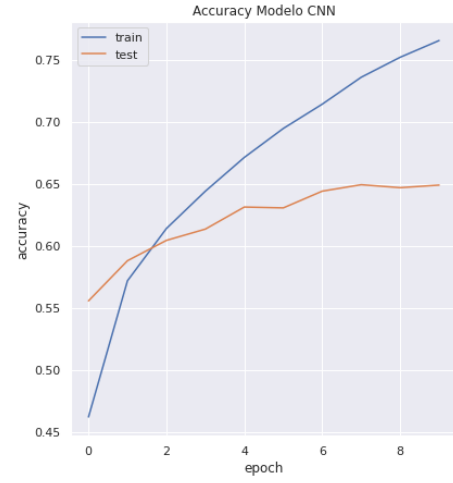


Figura 2. Accuracy por época modelo CNN

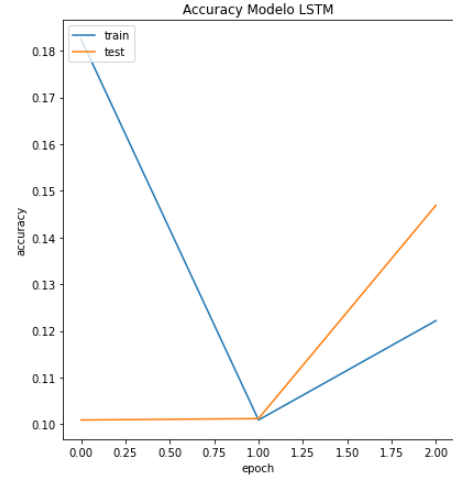


Figura 3. Accuracy por época modelo LSTM

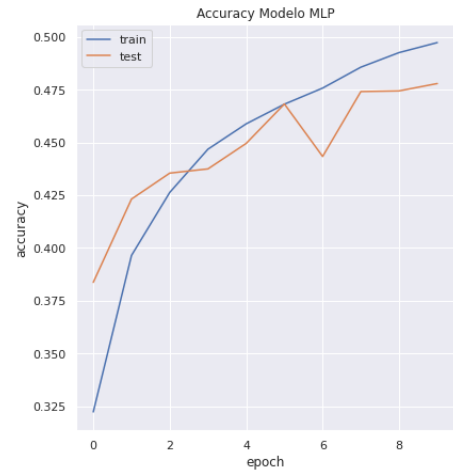


Figura 4. Accuracy por época modelo MLP

VI-B. Comparación entre los modelos

El accuracy obtenido durante la validación de cada modelos se detalla en la Tabla I.

Modelo	Accuracy Validación
RNN	20 %
LSTM	15 %
CNN	65 %
MLP	48 %
Naive Bayes	29 %
Árbol de Decisión	27 %
KMeans	22 %

Tabla I
ACURRACY DE VALIDACIÓN POR MODELO

Como se puede visualizar el modelo que obtuvo el mejor accuracy durante la validación fue la red neuronal convolucional.

VII. CONCLUSIONES

- Los modelos de machine learning ofrecen variadas técnicas de aprendizaje. Es importante comprender cuando es conveniente usar cada uno de estos.
- El algoritmo de Machine Learning que se utilice debe ser adecuado para el conjunto de datos. Es decir, debe ser capaz de manejar ese tipo de señal y la cantidad de instancias de conjunto de datos.
- El pre-procesamiento de la información es una etapa muy importante. Procesos como el de normalización o la extracción de características permiten que los modelos aprendan mejor.
- Los modelos de aprendizaje no supervisados son muy versátiles al momento de resolver un problema de cual no se posee mucha información.
- Entre las redes neuronales, los modelos con mejores resultados fueron CNN y MLP. Consideramos que esto es debido a la cantidad de instancias del conjunto de datos. Por otro lado, los modelos RNN y LSTM obtuvieron bajos resultados, probablemente, debido a que se empleó los mismos píxeles como datos de entrada.
- El mejor modelo fue CNN debido a que extrae los vectores de característica por su cuenta. Este hecho plantea un posible trabajo futuro en el cual se utilicen vectores de características como entrada de los otros modelos para observar que tanto mejora el rendimiento.

REFERENCIAS

- [1] TensorFlow. Clasificación de Imágenes. https://www.tensorflow.org/lite/models/image_classification/overview. Accedido 26-11-2020.
- [2] Pau Agustín Granell. Redes Neuronales Recurrentes: Una aplicación para los mercados bursátiles. http://diposit.ub.edu/dspace/bitstream/2445/124249/1/tfg_Pau_Agustin_memoria.pdf, 2017.
- [3] Chung J, Gulcehre C, Cho K, Bengio Y. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. <http://arxiv.org/abs/1412.3555>, 2014.
- [4] Kido S, Hirano Y, Hashimoto N. Detection and Classification of Lung Abnormalities by Use of Convolutional Neural Network (CNN) and Regions with CNN Features (R-CNN), 2018.
- [5] H Xian-Hua H, y Sun and Y Chen. Computer-aided classification of pulmonary diseases: RESIDUAL COMPONENT ESTIMATING CNN FOR IMAGE SUPER-RESOLUTION, 2019.
- [6] Matthieu Sainleza Georges Heyen. Recurrent neural network prediction of steam production in a Kraft recovery boiler. Computer Aided Chemical Engineering, 29, págs. 1784-1788, 2011.
- [7] Scikitlearn. Naive Bayes. https://scikit-learn.org/stable/modules/naive_bayes.html. Accedido 26-11-2020.
- [8] Rokach L. Maimon O. Introduction to Decision Trees. In Data Mining With Decision Trees: Theory And Applications, 2014.
- [9] Xin Jin Jiawei Han. K-Means Clustering. Encyclopedia of Machine Learning, 2010.