

Reporte de documentación de modelo de Red Neuronal Recurrente (RNN)

Diego Alfonso Ramírez Montes – A01707596

Data Set.

El data set utilizado tiene por nombre “*Complaints*”, el cual contiene información relativa a las quejas de los consumidores de ciertos productos ofrecidos por algunas compañías en los Estados Unidos. El data set cuenta con un total de 3,988,916 instancias.

```
1 [{"date_received": "2023-07-31", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
2 [{"date_received": "2023-07-31", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
3 [{"date_received": "2023-08-03", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
4 [{"date_received": "2023-08-16", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
5 [{"date_received": "2023-08-16", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
6 [{"date_received": "2023-08-16", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
7 [{"date_received": "2023-08-16", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
8 [{"date_received": "2023-08-16", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
9 [{"date_received": "2023-07-27", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
10 [{"date_received": "2017-04-30", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
11 [{"date_received": "2021-06-14", "product": "Credit card or prepaid card", "sub_product": "General-purpose credit card or charge card", "issue": "False statements or representation"},
12 [{"date_received": "2018-08-06", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
13 [{"date_received": "2017-07-30", "product": "Credit card or prepaid card", "sub_product": "General-purpose credit card or charge card", "issue": "False statements or representation"},
14 [{"date_received": "2017-07-29", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
15 [{"date_received": "2017-07-29", "product": "Debt collection", "sub_product": "Credit card debt", "issue": "False statements or representation"},
16 [{"date_received": "2022-10-06", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
17 [{"date_received": "2015-10-31", "product": "Credit card", "sub_product": "", "issue": "Identity theft / Fraud / Embezzlement", "sub_issue": ""},
18 [{"date_received": "2017-07-29", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
19 [{"date_received": "2023-01-10", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
20 [{"date_received": "2022-10-06", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
21 [{"date_received": "2021-06-14", "product": "Credit reporting, credit repair services, or other personal consumer reports", "sub_product": "Credit reporting, credit repair services, or other personal consumer reports", "issue": "False statements or representation"},
22 [{"date_received": "2017-07-29", "product": "Checking or savings account", "sub_product": "Checking account", "issue": "Closing an account", "sub_issue": ""},
23 [{"date_received": "2020-08-11", "product": "Mortgage", "sub_product": "FHA mortgage", "issue": "Applying for a mortgage or refinancing an existing mortgage"}]
```

Figura 1. Ejemplo del Dataset.

Modelo de Red Neuronal Recurrente

Para el desarrollo del modelo se tomó como el mejor candidato una Red Neuronal Recurrente, pues se pensaba trabajar con datos con formato de texto y realizar algo muy similar a un “*Sentiment Analysis*” por lo que una RNN fue la elección óptima basado en los ejemplos vistos en clase. El mismo fue definido como un modelo secuencial, con una capa de “*embedding*”, una capa de “*LSTM*”, una de “*Dropout*” y una capa de densa de activación con sigmoide.

```
# Define the model to use
model = Sequential()
model.add(Embedding(max_words, word2vec_model.vector_size, weights=[embedding_matrix], input_length=max_len, trainable=False))
model.add(LSTM(128))
model.add(Dropout(0.2))
model.add(Dense(1, activation='sigmoid'))
```

Figura 2. Modelo utilizado.

Evaluación del Modelo.

Para el caso particular del modelo se utilizó una sola métrica de evaluación, la cual sería el “*accuracy*” del mismo en los sets de entrenamiento y validación/test. Esto dado el pésimo rendimiento del modelo:

```
Epoch 1/10
1791/1791 [=====] - 784s 436ms/step - loss: 0.6895 - accuracy: 0.5376 - val_loss: 0.6896 - val_accuracy: 0.5351
Epoch 2/10
1791/1791 [=====] - 772s 431ms/step - loss: 0.6760 - accuracy: 0.5787 - val_loss: 0.6743 - val_accuracy: 0.5878
Epoch 3/10
1791/1791 [=====] - 809s 452ms/step - loss: 0.6655 - accuracy: 0.5994 - val_loss: 0.6732 - val_accuracy: 0.5771
Epoch 4/10
1791/1791 [=====] - 939s 524ms/step - loss: 0.6620 - accuracy: 0.6012 - val_loss: 0.6714 - val_accuracy: 0.5933
Epoch 5/10
1791/1791 [=====] - 2352s 1s/step - loss: 0.6580 - accuracy: 0.6100 - val_loss: 0.6694 - val_accuracy: 0.5925
Epoch 6/10
1791/1791 [=====] - 751s 419ms/step - loss: 0.6499 - accuracy: 0.6201 - val_loss: 0.6709 - val_accuracy: 0.5891
Epoch 7/10
1791/1791 [=====] - 765s 427ms/step - loss: 0.6422 - accuracy: 0.6288 - val_loss: 0.6723 - val_accuracy: 0.5956
Epoch 8/10
1791/1791 [=====] - 829s 463ms/step - loss: 0.6306 - accuracy: 0.6409 - val_loss: 0.6780 - val_accuracy: 0.5944
Epoch 9/10
1791/1791 [=====] - 996s 556ms/step - loss: 0.6136 - accuracy: 0.6568 - val_loss: 0.7009 - val_accuracy: 0.5916
Epoch 10/10
1791/1791 [=====] - 1028s 574ms/step - loss: 0.5909 - accuracy: 0.6776 - val_loss: 0.7049 - val_accuracy: 0.5831
```

Figura 3. Entrenamiento del modelo.

```
448/448 [=====] - 150s 334ms/step - loss: 0.7049 - accuracy: 0.5831
Test loss: 0.7048846483230591
Test accuracy: 0.5831180810928345
```

Figura 4. Evaluación del modelo.

Dado que el modelo es una implementación de regresión logística, no hace falta especificar que una precisión tan cercana al 50% representa básicamente que el resultado se está adivinando.

Mejoras.

En este apartado es necesario especificar algo primeramente y es que no guarde evidencia de todos los cambios que realicé en el modelo por la cantidad de tiempo que pasaba entre cada entrenamiento.

```
# Train the model
model.fit(X_train_pad, y_train, batch_size=32, epochs=10, validation_data=(X_test_pad, y_test))
✓ 167m 6.6s
```

Figura 5. Tiempos grandes por entrenamiento.

Así como diversos problemas que encontré al realizar el modelo, sin embargo, si puedo listar todas las mejoras que se realizaron y los resultados obtenidos.

1. Capa de regularización l2: Al primer modelo se le agregó una capa de regularización de l2 a fin de intentar mejorar la precisión de este (56% aproximadamente), esto no tuvo ningún resultado positivo significativo.
2. Modificación de Learning Rate: Se intentó modificar el learning rate del modelo de 0.001 a 0.01 mientras se mantenía la capa de regularización de l2. No tuvo ningún resultado favorable y terminó por disparar la pérdida en los últimos epochs de entrenamiento.
3. Uso de un modelo pre entrenado de embedding: En un inicio se había decidido por utilizar un modelo propio para el embedding, sin embargo se sospechó que el mismo podría ser insuficiente para el modelo, por lo que se decidió utilizar el modelo Word2Vec de Google "GoogleNews-vectors-negative300.bin". Dicha propuesta tampoco tuvo un efecto notorio en mejorar el modelo.
4. Uso de más columnas de la data set: Se pensó que utilizar más columnas de la data set podría ayudar al modelo a detectar de mejor forma si un consumidor disputó la queja o no, por lo que se le agregaron columnas extra al modelo con el fin de realizar "*Feature Engineering*". Esto resultó en una implementación mucho más compleja que tampoco llevó la precisión del modelo más allá del 56%.

5. Uso de más datos: En un inicio se utilizaron aproximadamente 8000 datos, repartidos de forma equitativa en 50% "Sí" para la disputa y 50% "No". Tras esto se usaron aproximadamente 12000 datos, luego 24000 y finalmente todos los 71614 datos. Esto tampoco trajo consigo ninguna mejora y fue la última implementación de una "mejora".

He de decir que todas las implementaciones incluyeron obviamente la limpieza de los datos a fin de no introducir comentarios vacíos ni valores nulos, así como todos los modelos contaban con la implementación del optimizador "Adam". Aún así ningún modelo fue capaz de superar la marca del 60% de acuraccy.

Conclusiones.

Trabajar con un modelo de RNN y, en general, un modelo de Deep Learning es bastante complicado, en el pasado ya me había enfrentado a un problema de este tipo al tener que realizar un detector de imágenes por medio de una CNN, sin embargo, creí que este problema no sería tan complicado como aquel, debo de decir que, tras todos los intentos de implementar una mejora, sigo sin comprender del todo que es lo que podría estar afectando tan negativamente a mi modelo y como es que pareciera que el 60% de precisión representa una barrera al momento de entrenar mi modelo.