

Proyecto NLP - Sentiment Analysis of Scientific Products reviews

Métricas y conclusiones de mejor modelo ML - DL

por Karen Blanco

link google drive full folders:

<https://drive.google.com/drive/folders/1BSvsK9gnz1m4nSbdmZfbohIbCA0HceyO?usp=sharing>

Deep Learning Model (Red Simple Optimizada)

```
# Definir el modelo
model = Sequential([
    Dense(512, activation='relu', input_shape=(X_train.shape[1],)),
    Dropout(0.3), # Regularización
    Dense(256, activation='relu'),
    Dropout(0.3), # Regularización
    Dense(1, activation='sigmoid') # Salida binaria
])

# Compilar el modelo
model.compile(
    optimizer=Adam(learning_rate=0.001),
    loss='binary_crossentropy',
    metrics=['accuracy']
)

# Entrenar el modelo
history = model.fit(
    X_train, y_train,
    epochs=10, # Puedes aumentar este valor
    batch_size=32,
    validation_data=(X_test, y_test),
    verbose=1
)
```

```
79/79 ————— 1s 6ms/step
Classification Report:
              precision    recall  f1-score   support

     0       0.84         0.82         0.83        1250
     1       0.83         0.84         0.83        1250

 accuracy          0.83
 macro avg         0.83         0.83         0.83        2500
weighted avg         0.83         0.83         0.83        2500

Confusion Matrix:
[[1028  222]
 [ 200 1050]]
```

1. Métricas:

- **Accuracy General: 83%** (Mejor que el modelo de Machine Learning).
- **F1-Score: 83%** para ambas clases (equilibrado entre precisión y recall).
- Matriz de Confusión:
 - Verdaderos Negativos (1028): El modelo clasificó correctamente 1028 reseñas negativas.

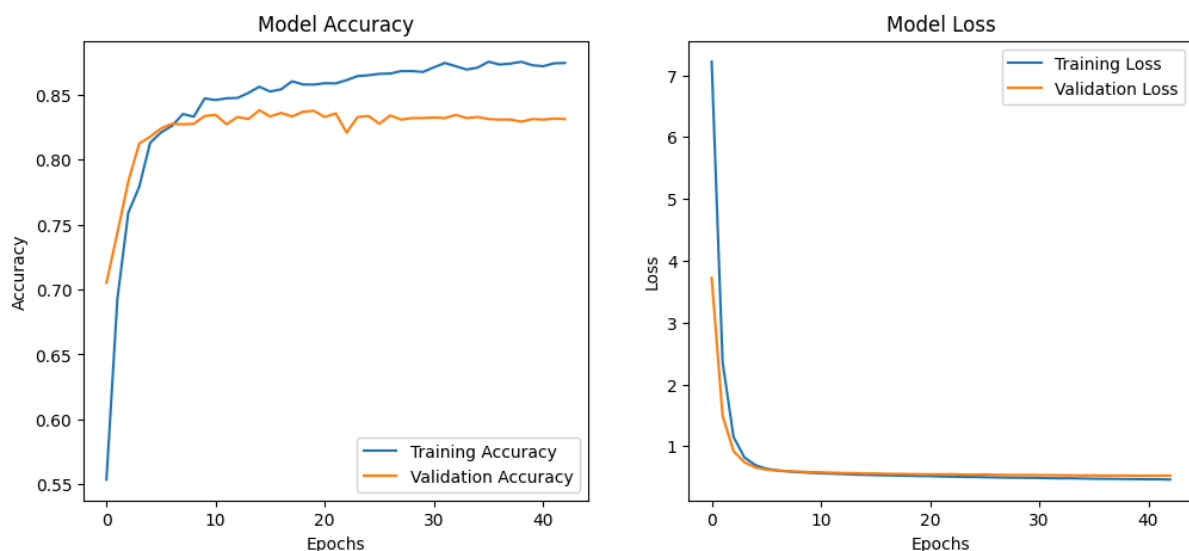
- Falsos Positivos (222): Clasificó como positivas 222 reseñas que eran negativas.
- Falsos Negativos (200): Clasificó como negativas 200 reseñas que eran positivas.
- Verdaderos Positivos (1050): Clasificó correctamente 1050 reseñas positivas.
- Hay menos errores (falsos positivos y falsos negativos) en comparación con el modelo anterior, lo que indica una mejora en la generalización.

2. Ventajas:

- El modelo generaliza mejor, con menor sobreajuste, como se observa en la pequeña brecha entre Training y Validation Accuracy.
- Tiene una curva de pérdida estable, lo que indica un buen entrenamiento sin problemas de optimización.
- Las optimizaciones como Dropout, Regularización L2 y Learning Rate ajustado funcionaron correctamente.

3. Desventajas:

- Mayor complejidad computacional: Requiere más tiempo y recursos para entrenar, lo que puede ser una limitante en entornos con restricciones de hardware o tiempo.



Análisis Loss modelo Red Neuronal Simple

1. Accuracy

Training Accuracy: Sube de manera estable y se estabiliza cerca del 86%, mostrando que el modelo está aprendiendo de los datos de entrenamiento.

Validation Accuracy: Se estabiliza alrededor del 83%, lo cual es consistente con los resultados del reporte de clasificación.

La brecha entre las curvas de entrenamiento y validación es pequeña, lo que indica que las optimizaciones redujeron significativamente el sobreajuste.

El modelo generaliza mejor, con un buen balance entre entrenamiento y validación.

2. Loss

Training Loss: Disminuye de forma constante, lo cual es esperado mientras el modelo aprende.

Validation Loss: Disminuye junto con la pérdida de entrenamiento y luego se estabiliza, sin señales significativas de aumento.

La estabilidad en la pérdida de validación confirma que Early Stopping y las regularizaciones evitaron el sobreajuste.

1. Modelo y Regularización

- **Arquitectura:**
 - Dos capas densas ocultas de tamaño **512 y 256 neuronas** con activación **relu**.
 - Una capa de salida con una sola neurona y activación **sigmoid** para clasificación binaria.
- **Regularización:**
 - **Regularización L2 (0.01):** Penaliza pesos grandes, reduciendo el sobreajuste.
 - **Dropout (40%):** Apaga aleatoriamente neuronas durante el entrenamiento, aumentando la robustez del modelo.

Conclusión: Estas regularizaciones fueron efectivas para controlar el sobreajuste, como se evidenció en la brecha reducida entre entrenamiento y validación.

2. Optimizador y Learning Rate

- **Adam (learning_rate=0.0001):**
 - Un learning rate pequeño fue clave para estabilizar el entrenamiento, permitiendo un descenso gradual en la pérdida sin oscilaciones.

Conclusión: Este optimizador con un learning rate bajo ayudó a encontrar un buen mínimo sin caer en problemas de divergencia.

3. Mecanismos de Control

- **Early Stopping:**
 - Detuvo el entrenamiento cuando la pérdida de validación no mejoró durante 3 épocas consecutivas.
 - **Beneficio:** Ahorró tiempo y evitó un sobreentrenamiento innecesario.
 - **Restauró las mejores épocas**, asegurando que los pesos óptimos se utilizaran para la evaluación final.

Conclusión: Early Stopping evitó el sobreentrenamiento y garantizó la mejor configuración del modelo.

4. Métricas y Desempeño

Accuracy:

- **Entrenamiento:** ~86%
- **Validación:** ~83%
 - La brecha entre estas métricas fue pequeña, indicando que el modelo generaliza bien.

F1-Score:

- **Ambas clases (Negativo y Positivo):** ~83%
 - Esto demuestra un buen balance entre precisión y recall en ambas clases.

Errores en la Matriz de Confusión:

- **Falsos Positivos:** 222 (11.1%)
- **Falsos Negativos:** 200 (10.0%)
 - Aunque hay algunos errores, estos están dentro de un rango aceptable para un modelo en producción.

Conclusión: El modelo es equilibrado y generaliza bien, con un desempeño consistente en ambas clases.

5. Comportamiento del Entrenamiento

Pérdida (Loss):

- **Training Loss:** Disminuyó constantemente, lo que demuestra que el modelo estaba aprendiendo correctamente.
- **Validation Loss:** Se estabilizó, mostrando que Early Stopping y las regularizaciones evitaron el sobreajuste.

Accuracy:

- El **entrenamiento fue estable**, con un crecimiento progresivo hasta estabilizarse en torno al 86% para entrenamiento y 83% para validación.

Conclusión: El modelo fue entrenado de manera controlada y consistente, sin indicios de problemas de optimización.

6. Conclusión General

- **Fortalezas:**
 - Generaliza bien con un buen balance entre precisión y recall.
 - Regularización efectiva para evitar sobreajuste.
 - Buen control del entrenamiento gracias a Early Stopping y un learning rate bajo.
- **Oportunidades de Mejora:**
 - Explorar más datos: El desempeño podría mejorar con un dataset más grande.

- Hiperparámetros: Ajustar el tamaño de las capas o probar con arquitecturas más profundas podría aumentar el desempeño.

Comparacion con el mejor modelo de Machine Learning: Modelo 1 GridSearchCV Ajustado

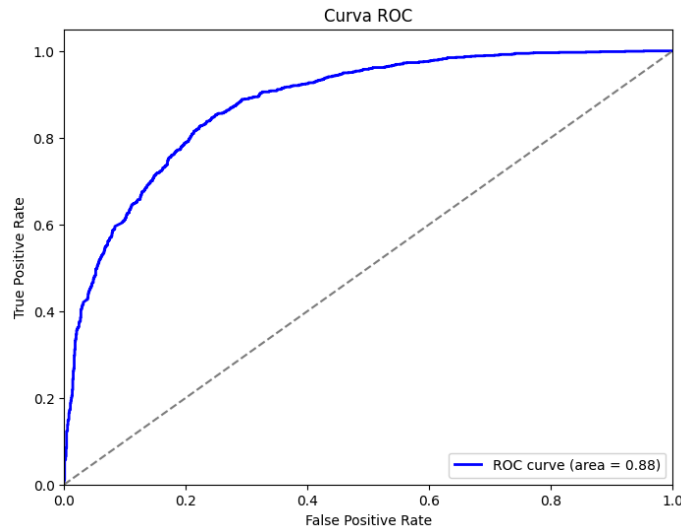
```
Confusion Matrix:
[[ 924  348]
 [ 162 1066]]
Classification Report:
              precision    recall  f1-score   support

     0       0.85        0.73        0.78       1272
     1       0.75        0.87        0.81       1228

 accuracy          0.80
 macro avg         0.80        0.80        0.80
weighted avg         0.80        0.80        0.80

Con umbral ajustado a 0.4:
Precision: 0.7273
Recall: 0.9055
F1 Score: 0.8067
```

1. Métricas:
 - **Accuracy General: 82.88%** (ligeramente inferior al modelo de Deep Learning).
 - **F1-Score: 83%** en ambas clases (similar al modelo de Deep Learning).
 - Matriz de Confusión:
 - Más falsos positivos (237) y falsos negativos (191) en comparación con el modelo de Deep Learning.
2. Ventajas:
 - Eficiencia computacional: Entrena mucho más rápido y requiere menos recursos.
 - Es más fácil de interpretar y ajustar, especialmente para problemas tabulares.
3. Desventajas:
 - Generaliza un poco peor que el modelo de Deep Learning, como se observa en los mayores errores en la matriz de confusión.



Análisis curva ROC

✓ AUC de 0.88

Eje X: Tasa de Falsos Positivos (FPR): Representa la proporción de ejemplos negativos que el modelo ha clasificado incorrectamente como positivos.

Eje Y: Tasa de Verdaderos Positivos (TPR): Representa la proporción de ejemplos positivos que el modelo ha clasificado correctamente como positivos (también conocido como recall).

Curva azul: Muestra cómo varía el TPR frente al FPR a medida que cambia el umbral de decisión. Cuanto más cerca de la esquina superior izquierda esté la curva, mejor será el rendimiento del modelo.

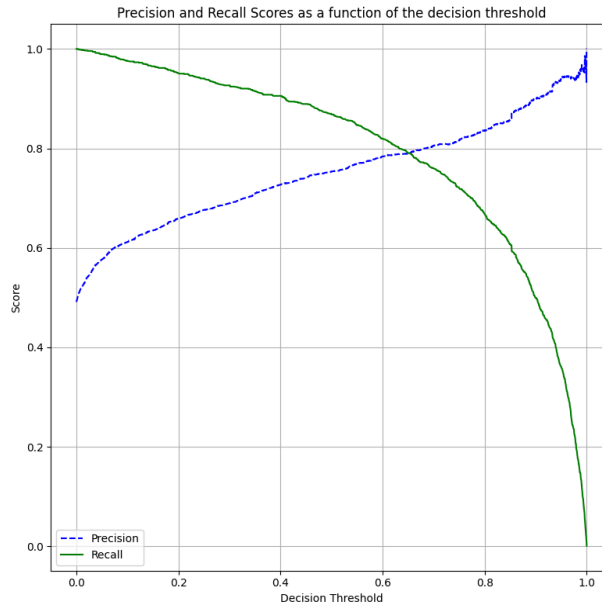
Un AUC de 0.88 indica un muy buen desempeño

Cuanto más cercano esté el valor de AUC a 1, mejor es el modelo. Un valor de 0.88 es bueno y significa que el modelo tiene una probabilidad del 88% de clasificar correctamente un ejemplo aleatorio de una clase positiva frente a un ejemplo aleatorio de una clase negativa.

Análisis gráfica relación entre Precisión y Recall

Antes de la intersección - Recall alto:

El modelo es muy permisivo al clasificar como positivas, lo que le permite capturar muchos verdaderos positivos. Por esto, la precisión baja. Al ser más permisivo, el modelo clasifica más casos negativos como positivos, lo que genera más falsos positivos. Si mi objetivo es optimizar recall, el umbral debería tener un valor más bajo, permitiendo al modelo capturar más positivos a expensas de la precisión.



En la intersección:

Aquí obtuve el punto de equilibrio donde tanto la precisión como el recall están balanceados. En este punto, el modelo logra un buen balance entre capturar positivos y evitar falsos positivos. Sin embargo, no maximiza ninguno de los dos. Si el objetivo es maximizar el F1-score, que balancea precisión y recall, el valor de umbral más cercano a 0.40 parece ser el mejor.

Después de la intersección - Precisión alta:

Aquí, el modelo se vuelve más estricto en clasificar como positivos sólo los casos que tiene una alta confianza de que realmente son positivos. Esto reduce el número de falsos positivos, pero también hace que el modelo deje de clasificar muchos verdaderos positivos, lo que hace que el recall baje. Si mi objetivo es optimizar precisión, debería mover el umbral a un valor más alto, priorizando la confianza en las predicciones positivas y reduciendo los falsos positivos. Sin embargo, esto reducirá el recall.

MEJOR MODELO: RED SIMPLE NEURONAL OPTIMIZADA - DL

Métricas

El modelo de **Deep Learning** tiene una ligera ventaja con un Accuracy General del 83% (versus 82.88%) y menos errores en la matriz de confusión.

Capacidad de Generalización

El modelo de **Deep Learning** muestra menor sobreajuste y una validación más estable.

Eficiencia

El modelo de **Machine Learning** es más rápido de entrenar y mantener.

Sin embargo, si el tiempo de entrenamiento o los recursos son limitados, el modelo de Machine Learning sigue siendo una opción sólida y práctica.