

Análisis de sentimientos utilizando un modelo de IA basado en LSTM y CNN: comparación con modelos previos

AYLIN CAMACHO¹, LUIS CANO¹, VALERIA CONDE¹, ERICK GOMEZ¹, AXEL VILLANUEVA¹

¹Instituto Tecnológico y de Estudios Superiores de Monterrey, Monterrey, NL 64849 MX

Autor de correspondencia: Aylin Camacho (e-mail: a01379272@tec.mx).

Este trabajo fue elaborado como parte de la evaluación final para el curso TC3002B del Instituto Tecnológico y de Estudios Superiores de Monterrey.

ABSTRACT Este artículo científico propone el desarrollo de un modelo de inteligencia artificial utilizando Redes Neuronales Convolucionales (CNN) y Memoria a Corto y Largo Plazo (LSTM) para realizar análisis de sentimiento en críticas de películas. El estudio busca proporcionar una solución a la necesidad de análisis automatizado de grandes cantidades de datos de texto y de identificar la opinión específica de miles de reseñas. El artículo presenta el modelo propuesto y explica detalladamente sus características relevantes. El estudio también incluye el uso de una base de datos de 50,000 críticas de películas de IMDb, que se considera la fuente más popular y autorizada de contenido cinematográfico. La investigación busca contribuir a la mejora de los procesos de análisis y toma de decisiones, y proporcionar un análisis comparativo del modelo propuesto con modelos clásicos de análisis de sentimiento. Del artículo se obtiene que el modelo desarrollado en el mismo alcanza una exactitud del 87.91 % en el set de evaluación designado. El artículo concluye con una discusión de los resultados, una comparativa con modelos anteriores y propuestas para futuras mejoras del modelo.

INDEX TERMS Análisis de sentimientos, Convolutional Neural Networks, Inteligencia artificial, Long short term memory.

I. INTRODUCCIÓN

LA Inteligencia Artificial describe aquellos sistemas que tienen la capacidad de “razonar, aprender y actuar de una manera que normalmente requeriría inteligencia humana o que involucre datos cuya escala exceda lo que los humanos pueden analizar” [1], lo que la vuelve una herramienta imprescindible para empresas e individuos que buscan mejorar procesos de análisis y toma de decisiones, así como una herramienta eficaz capaz de automatizar funciones que normalmente requerirían de un ser humano, como chatbots o recomendadores de artículos.

Por otra parte, el análisis de sentimientos es definido como “un método para automáticamente clasificar grandes cantidades de texto como positivos o negativos” [2]. Dicha aplicación de la Inteligencia Artificial (IA) continúa ganando tracción debido a la cada vez más abundante cantidad de datos en internet.

Entre las aplicaciones que destacan del uso de dicho método de clasificación, las reseñas de productos, servicios y empresas son las más notorias. De ellas, la que concierne al proyecto es el análisis de sentimientos en reseñas de películas, debido a la necesidad de implementar una solución de software que permita identificar rápidamente de forma automática la opinión específica de miles de reseñas, y así posteriormente realizar análisis descriptivos de dicho contenido multimedia de ser necesario.

Con el fin de brindar una solución a la problemática, planteamos el desarrollo de un modelo de inteligencia artificial que utilizará los métodos de Convolutional Neural Networks y Long Short-Term Memory (LSTM) para realizar el análisis de sentimientos que nos compete, pues la literatura actual sustenta el uso de dichas técnicas [3].

LSTM es una técnica derivada de Recurrent Neural Networks que resulta especialmente útil para retener in-

formación relevante de las entradas por mayor duración y, al mismo tiempo, descartar información irrelevante, funcionando incluso para analizar granularmente por palabras y caracteres [4].

Por otra parte, CNN se ha utilizado previamente para resolver problemas relacionados con Natural Language Processing como Sentence Classification [5], Text Summarization [6], Text Classification [7], entre otros.

Para el entrenamiento y validación del modelo planeamos el uso de una base de datos con 50,000 reseñas obtenidas de IMDb, a través del sitio de recopilación de datasets Kaggle. IMDb se describe como la fuente más popular y autoritativa para contenido de películas, televisión y celebridades, así como un medio para ayudar a fanáticos a decidir qué mirar [8], por lo que resulta la opción más adecuada para completar este proyecto.

Las principales contribuciones de esta investigación son:

- Propuesta de un modelo de Co-LSTM para el análisis de sentimiento orientado a la clasificación de reseñas de películas en positivas y negativas.
- Estudio de resultados obtenidos a partir de variaciones a nuestro modelo. Se contempla principalmente el estudio de aplicar primero CNN y luego LSTM, y viceversa.
- Análisis de resultados obtenidos y discusión de efectos producidos en el modelo a partir del ajuste de los hiper parámetros.
- Comparación de resultados con modelos clásicos utilizados previamente para el análisis de sentimientos.
- Propuesta de mejoras futuras al modelo a partir de los resultados finales y las hipótesis que se puedan generar a partir de ellas.

El resto del documento se divide como se describe a continuación: la sección Fundamentos describe los antecedentes de la investigación, así como los conceptos clave involucrados. El apartado Método propuesta explica el modelo de IA a utilizar, mencionando detalladamente cada una de sus características relevantes. La Metodología describe la información relevante para replicar los resultados de esta investigación, a través de pruebas coherentes y suficientes. A continuación, la sección de Resultados expone a detalle los resultados obtenidos de aplicar las pruebas de la metodología descrita anteriormente. Finalmente, en Conclusiones se explica el análisis de los resultados, las implicaciones de esta investigación y las propuestas de mejora a futuro.

II. FUNDAMENTOS

En una investigación propuesta por [9] sobre los modelos Co-LSTM (de memoria convolucional a largo y corto plazo) para el análisis de sentimientos en Big Data social, se describió una arquitectura que utiliza capas convolucionales (CNN) para la extracción de características y capas LSTM para el modelado secuencial. Los

autores comparan su modelo propuesto con otros modelos existentes en datasets de datos de redes sociales de referencia, y los resultados muestran que el modelo Co-LSTM supera a otros modelos en términos de precisión, puntuación F1 y puntuación AUC, así como contó con mejores resultados en casos de negación y sarcasmo.

Este antecedente lleva a la propuesta de investigación de nuestra interpretación de la arquitectura Co-LSTM con propias variaciones experimentales, para el campo de análisis de sentimientos en reseñas de películas, siendo un tipo específico de comentario social elaborado por humanos y con un sentimiento implícito.

Los conceptos clave involucrados en este escrito son las dos estructuras de Aprendizaje Profundo utilizadas en la propuesta.

II-A. DEEP LEARNING

Deep learning (conocido como Aprendizaje Profundo) es un método de entrenamiento utilizado en inteligencia artificial que permite a los modelos computacionales “aprender representaciones de información con múltiples niveles de abstracción” ???. Utiliza redes neuronales que simulan el comportamiento de las neuronas humanas para procesar grandes cantidades de información. La forma en que logra funcionar para procesar dicha información es a través de ligeros cambios entre capas a partir de la información proveída por las capas anteriores.

II-B. CNN

Una Convolutional Neural Network (CNN) es un modelo de machine learning que consiste en capas de convulsión, pooling y capas densamente conectadas. Se usa normalmente para reconocimiento de imágenes y procesamiento de señales. “La característica más importante de una red neuronal convolucional es su capacidad para aprender automáticamente jerarquías espaciales de características a partir de los datos de entrada” [11]. Podemos usarla para análisis de sentimientos, de manera que reciba texto y el modelo extraiga las características útiles del texto usando filtros convolucionales para analizar si el sentimiento es positivo, negativo o neutral.

II-C. LSTM

Esta red neuronal se utiliza regularmente en tareas de procesamiento de lenguaje natural y de aprendizaje automático secuencial. “La innovación clave de LSTM es su mecanismo de activación, que le permite actualizar u olvidar selectivamente información en la celda de memoria, brindándole la capacidad de recordar o descartar información según sea necesario” [12]. Cada unidad tiene una celda de memoria, la cual puede guardar información durante periodos de tiempo largos y tres puertas (entrada, olvido y salida) las cuales regulan el flujo de información dentro y fuera de la celda.

III. MÉTODO PROPUESTO

III-A. ENTENDIMIENTO DE LOS DATOS

El objetivo de esta sección del modelo es explicar cómo se visualizan los datos para entender mejor el problema que estamos abordando. Se presentan tres gráficos que nos ayudan a comprender la distribución de críticas positivas y negativas, de palabras utilizadas en las críticas y las palabras más comunes utilizadas en las críticas positivas y negativas.

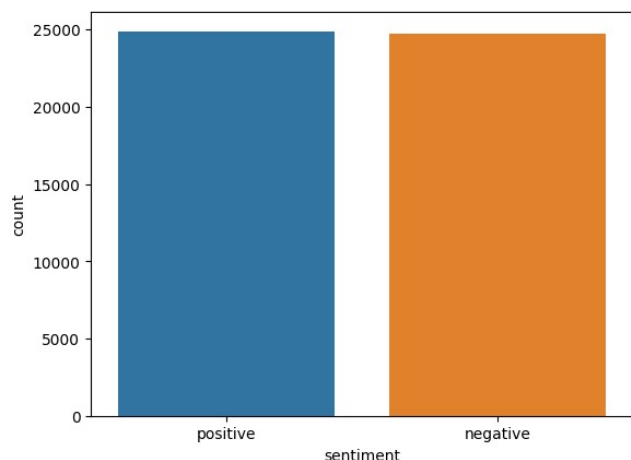


Figura 1. Distribución de críticas positivas y negativas.

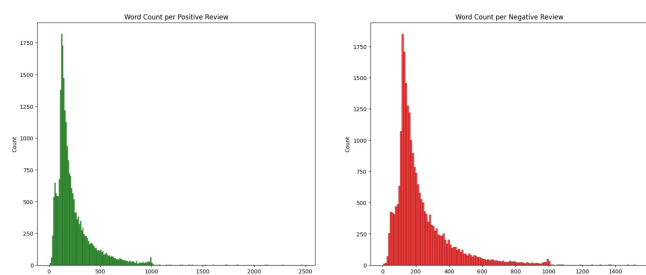


Figura 2. Distribución de palabras utilizadas en las críticas.

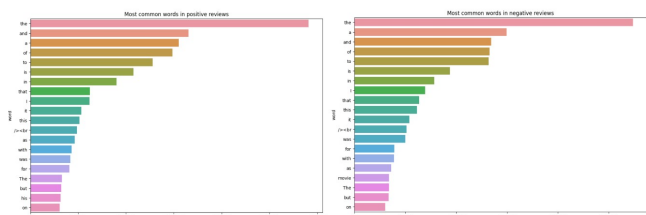


Figura 3. Distribución de palabras más comunes utilizadas en las críticas.

Las visualizaciones presentadas en las Figuras 1-3, nos ayudaron a entender mejor los datos y nos permitieron identificar patrones en las críticas que podrían ser útiles para entrenar un modelo de aprendizaje automático.

III-B. PREPROCESAMIENTO DE LOS DATOS

Esta etapa previa al modelo sirvió para convertir el texto original en un formato procesable y estructurado, eliminando información redundante, ruido y estandarizar el texto.

En primer lugar, se eliminan las abreviaciones y se reemplazan por sus formas completas utilizando un diccionario que contiene las abreviaciones y sus correspondientes formas completas. También, se eliminaron las palabras vacías o stop words, que son palabras comunes en un idioma que no aportan significado al texto, como “el”, “la”, “de”, etc. Esto se realiza para reducir el tamaño del vocabulario y mejorar la eficiencia del análisis, mientras que la eliminación de palabras vacías para evitar problemas de ambigüedad y mejorar la coherencia del texto.

	review	sentiment
0	One reviewers mentioned watching 1 Oz episode hooked. They right, exactly happened me. ...	positive
1	A wonderful little production. The filming technique unassuming- old-time-BBC fashio...	positive
2	I thought wonderful way spend time hot summer weekend, sitting air conditioned theater watching ...	positive
3	Basically family little boy (lake) thinks zombie closet & parents fighting time. This...	negative
4	Petter Mattei's "Love Time Money" visually stunning film watch. Mr. Mattei offers us vivid portr...	positive

Figura 4. Muestra de los datos después de eliminar abreviaciones y stop words.

Luego, se realizan varios procesos de limpieza de texto, incluyendo la eliminación de signos de puntuación, números, caracteres especiales y etiquetas HTML. Este proceso es fundamental para reducir el ruido, asegurar que el análisis se centre en las palabras relevantes y su contexto.

	review	sentiment
0	One reviewers mentioned watching 1 Oz episode hooked They right exactly happened me The first ...	positive
1	A wonderful little production The filming technique unassuming old time BBC fashion gives comf...	positive
2	I thought wonderful way spend time hot summer weekend sitting air conditioned theater watching ...	positive
3	Basically family little boy Jake thinks zombie closet parents fighting time This movie slowe...	negative
4	Petter Mattei's Love Time Money visually stunning film watch Mr Mattei offers us vivid portr...	positive

Figura 5. Muestra de los datos después de la limpieza.

Por último, se agregan tokens <start> y <end> a cada review y se aplica la lematización, que consiste en reducir las palabras a su forma base o raíz, para reducir el número de palabras distintas que aparecen en el texto. Para esto se utiliza el lematizador de WordNet, que identifica la forma base de una palabra y la devuelve en su forma lematizada.

III-C. TRANSFORMACIÓN DE DATOS

En la etapa de transformación de datos mapeamos los valores de ‘sentiment’ para que el modelo pudiera trabajar con números en lugar de texto, donde positivo es 1 y negativo es 0. Luego realizamos la tokenización, donde hacemos un mapeo del texto de palabras a números únicos, para crear el vocabulario para el entrenamiento.

	review	sentiment
0	<start> One reviewer mentioned watching 1 Oz episode hooked They right exactly happened me The f...	positive
1	<start> A wonderful little production The filming technique unassuming old time BBC fashion give...	positive
2	<start> I thought wonderful way spend time hot summer weekend sitting air conditioned theater wa...	positive
3	<start> Basically family little boy Jake think zombie closet parent fighting time This movie slo...	negative
4	<start> Petter Mattei s Love Time Money visually stunning film watch Mr Mattei offer u vivid por...	positive

Figura 6. Muestra de los datos al final de la etapa de preprocesamiento.

Después de la tokenización hacemos una vectorización del texto, donde representamos cada opinión con una secuencia de números con el vocabulario que generamos. Usamos padding y truncamiento para que todas las opiniones tengan el mismo tamaño.

Finalmente usamos embedding para asignar a cada palabra un vector de números que representan su significado. Con ello creamos la matriz de embeddings con los vectores de cada palabra y asignamos un valor aleatorio a las palabras que no sean parte de los embeddings. Esto ayuda a mejorar el modelo pues contextualiza las palabras.

III-D. CREACIÓN DE MODELO

Para el modelo usamos una red neuronal recurrente que contiene capas de convolución y capas densas; utilizamos una capa de embedding con pesos pre-entrenados para convertir las palabras en vectores numéricos y añadimos una capa de normalización. La entrada procesada entra entonces a una capa Bidirectional LSTM con 128 neuronas y aplicamos una tasa de dropout de 0.3.

Seguidamente se aplican 3 capas de convolución, con 128 filtros y kernel de tamaño 7, con 256 filtros y kernel de tamaño 5 y con 512 filtros y kernel de tamaño 3, cada una con su respectivo Max Pooling para reducir la muestra.

Luego pasa a dos capas adicionales de Bidirectional LSTM, ambas con 128 neuronas y la segunda de ellas con un dropout recurrente de 0.2.

Finalmente agregamos dos capas densas con 128 y 1 neurona respectivamente y una función de activación sigmoide en la capa de salida que genera 0s y 1s, con la que obtenemos la predicción de sentimiento positivo o negativo.

Para compilar el modelo usamos el optimizador Adam con un learning rate de 0.0005, la función de pérdida binary cross entropy, y la métrica de evaluación es la exactitud de clasificación.

IV. METODOLOGÍA

Para evaluar y seleccionar al mejor modelo de Co-LSTM en su tarea de predicción de sentimiento en reseñas de películas, llevamos a cabo una metodología dividida en experimentación preliminar y experimentación final.

IV-A. PRUEBAS PRELIMINARES

En la primera etapa probamos diferentes configuraciones de nuestro modelo, utilizando capas de CNN y capas de LSTM de manera variada y en distintas secuencias.

Probamos las siguientes variaciones en la secuencia de las capas:

- Capas de CNN antes de las capas de LSTM.
- Capas de LSTM antes de las capas CNN.
- Capas CNN entre capas de LSTM.

También exploramos la inclusión de una o varias capas de cada tipo. Además, para mejorar el rendimiento y la capacidad de capturar características contextuales, aplicamos capas de LSTM bidireccionales en los modelos.

Para la evaluación de los modelos generados y el rendimiento de cada estructura propuesta, utilizamos las métricas proporcionadas por Keras de Loss (pérdida) y Accuracy (precisión), las cuales se evaluaron en cada caso sobre los datos de entrenamiento, los datos de validación y los datos de prueba.

En las pruebas preliminares se generaron modelos listados en la Figura 7.

Nombre	Estructura (cantidades de neuronas cambian)	Accum Dropout	LR	Epoche	Callbacks
model1.h5	Embed/BatchNorm/BiLSTM/CNN(3)/BiLSTM/LSTM/Dense/Out	0.8	0.0001	9 / 15	EarlyStop
model2.h5	Embed/BatchNorm/BiLSTM/CNN(3)/BiLSTM(2)/Dense/Out	0.8	0.0001	7 / 15	EarlyStop
model3.h5	Embed/BatchNorm/CNN/BiLSTM/LSTM/Dense/Out	0.3	0.001	7 / 15	EarlyStop
model4.h5	Embed/BiLSTM/CNN(2)/Flatten/Dense/Out	0.3	0.001	9 / 15	EarlyStop, Checkpoint
model5.h5	Embed/BiLSTM/CNN/LSTM/Out	0.1	0.001	8 / 15	EarlyStop, Checkpoint

Figura 7. Muestra de los datos al final de la etapa de preprocesamiento.

Todos fueron entrenados con el optimizador Adam bajo las condiciones de 15 epochs, utilizando EarlyStopping de dos epochs de paciencia y contando con el mismo embedding de GloVe. Todos los modelos generados a partir de las pruebas fueron guardados en una carpeta para su reutilización.

Una vez realizadas las pruebas preliminares, evaluamos la precisión de cada modelo utilizando un conjunto de datos de prueba. Tras un análisis detallado, involucrando la visualización de las curvas de pérdida y precisión, seleccionamos al Modelo 2, el cual obtuvo los resultados presentados en la Figura 8 y 9.

Este modelo está conformado por las siguientes capas:

```
embedding_54 (Embedding)
batch_normalization_52 (BatchNormalization)
bidirectional_63 (Bidirectional)
dropout_149 (Dropout)
conv1d_152 (Conv1D)
max_pooling1d_144 (MaxPooling1D)
conv1d_153 (Conv1D)
max_pooling1d_145 (MaxPooling1D)
conv1d_154 (Conv1D)
dropout_150 (Dropout)
max_pooling1d_146 (MaxPooling1D)
bidirectional_64 (Bidirectional)
bidirectional_65 (Bidirectional)
```

dense_85 (Dense)
 dropout_151 (Dropout)
 dense_86 (Dense)

Como se puede notar, este modelo implementa la variación de contar con capas CNN rodeadas por capas LSTM bidireccionales. Esta secuencia de la estructura Co-LSTM es una variación a las más comunes, sobre la

cual decidimos profundizar dado su rendimiento superior a las otras en el presente caso.

En su instancia original este modelo se entrenó con embedding de GloVe, un Learning Rate de 0.0001, con pérdida de “binary_crossentropy” y “métrica de accuracy”.

IV-B. PRUEBAS FINALES

Posteriormente llevamos a cabo una experimentación final más específica para refinar y mejorar aún más el rendimiento del modelo seleccionado. Realizamos cambios en diferentes aspectos del modelo:

- Cambios en Learning Rate con el objetivo de convergencia y de suavizar y reducir las fluctuaciones en la curva de pérdida.
- Modificaciones en el Dropout, para evitar los sobre ajustes y buscar que la precisión de entrenamiento no se encuentre muy lejana a aquella de validación y prueba.
- Cantidad de neuronas en las capas, aumentando la cantidad de operaciones y posibilidad de que el modelo aprenda mejor.

Adicionalmente, intentamos agregar una capa adicional de CNN para evaluar su impacto al rendimiento del modelo, pero esta no trajo mejoras.

Durante esta fase de experimentación final, medimos la precisión de cada modelo resultante utilizando el mismo conjunto de datos de prueba. Se probaron los siguientes casos sobre el modelo 2:

- Modelo 6: Aumento de Learning Rate a 0.001, modificación en los tamaños de los kernels de las capas CNN y mayor cantidad de neuronas en las capas de LSTM bidireccionales.
- Modelo 7: Los cambios anteriores, ahora con Learning Rate de 0.0005 y reducción de capas de Dropout pero incremento en los valores de las restantes.
- Modelo 8: Los cambios anteriores, ahora con Learning Rate de 0.0002 y aumento de Dropout en las capas.

A partir de los resultados obtenidos, se generó una tabla que resume la precisión de cada modelo y nos permite realizar una comparación detallada:

A partir del análisis de estos resultados, se encontró que el Modelo 7 obtuvo la precisión más alta en los datos de prueba, con un 88 % de precisión en sus predicciones. Este modelo fue basado en la estructura del Modelo 2, incorporando las modificaciones mencionadas. Estos cambios demostraron ser beneficiosos. Se trató aumentar aún más el Dropout y modificar el Learning Rate en el Modelo 8, pero esto trajo un peor rendimiento, por lo cual decidimos al Modelo 7 como nuestra mejor representación de la arquitectura Co-LSTM para la predicción de sentimiento en reseñas de películas.

V. RESULTADOS

El modelo final de Co-LSTM fue el siguiente:

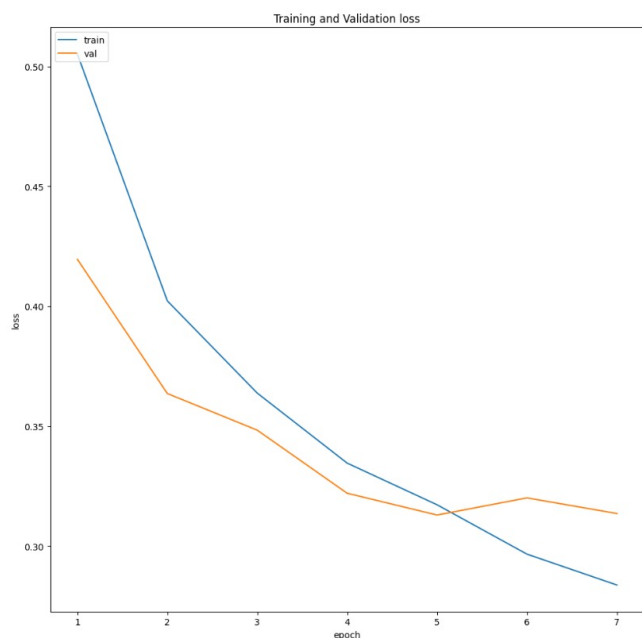


Figura 8. Historia de pérdida a lo largo de las epochs para el Modelo 2.

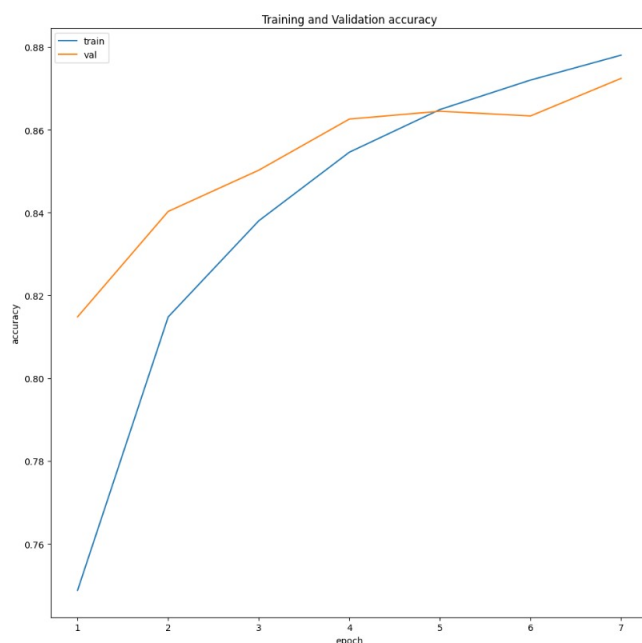


Figura 9. Historia de precisión a lo largo de las epochs para el Modelo 2.

Nombre	Accuracy		
	train	val	test
model1.h5	0.9343565106	0.8686499596	0.8593324423
model2.h5	0.8949010372	0.8724316359	0.8603408337
model3.h5	0.9210261106	0.8605823517	0.8550972939
model4.h5	0.9338207245	0.8719273806	0.8684077859
model5.h5	0.8919702768	0.8651204109	0.8597357869
model7.h5	0.9519727826	0.8878104091	0.879096508
model8.h5	0.9355855584	0.8889449239	0.871735394

Figura 10. Tabla de Resultados de modelos generados.

```

Capa, (tamaño de output), # parámetros
=====
Embedding, (None, 189, 100), 8900800
BatchNormalization, (None, 189, 100), 400
Bidirectional, (None, 189, 256), 234496
Dropout, (None, 189, 256), 0
Conv1D, (None, 183, 128), 229504
MaxPooling1D, (None, 91, 128), 0
Conv1D, (None, 87, 256), 164096
MaxPooling1D, (None, 43, 256), 0
Conv1D, (None, 41, 512), 393728
Dropout, (None, 41, 512), 0
MaxPooling1D, (None, 20, 512), 0
Bidirectional, (None, 20, 256), 656384
Bidirectional, (None, 256), 394240
Dense, (None, 128), 32896
Dropout, (None, 128), 0
Dense, (None, 1), 129
=====

```

Con un total de 11,006,673 de parámetros, de los cuales 2,105,673 son entrenables y 8,901,000 no son entrenables.

Este cuenta con una capa bidireccional de LSTM, seguida por tres capas de CNN, seguidas por 2 capas bidireccionales de LSTM y finalmente una capa densa y una de salida.

En las Figuras 11 y 12 podemos observar sus resultados en pérdida y precisión en los datos de entrenamiento y validación.

Se puede ver cómo a pesar de las ligeras fluctuaciones en la precisión de validación, esta tiende a constante aumento.

Finalmente, vemos los resultados de las predicciones sobre datos de prueba utilizando al Modelo 7 en la Figura 13.

VI. CONCLUSIONES

Seleccionamos al Modelo 7 debido a su capacidad para capturar características relevantes y contextuales, así como su alto rendimiento en la clasificación de sentimientos en las reseñas de películas.

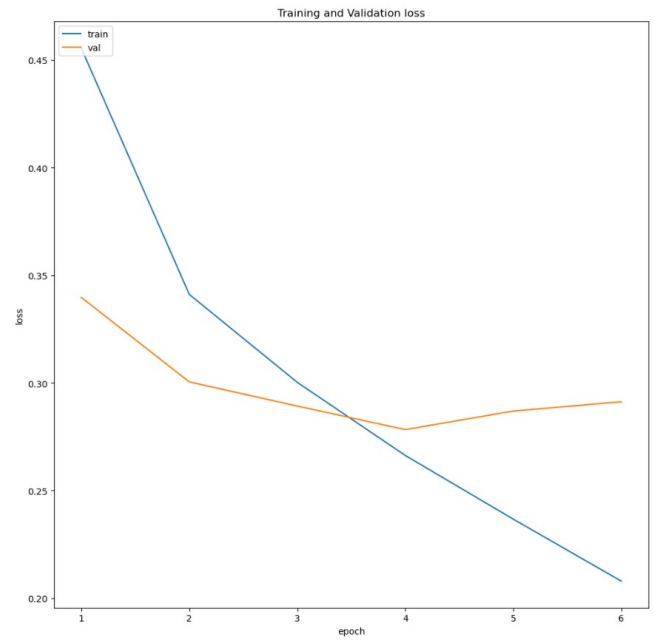


Figura 11. Historia de pérdida a lo largo de las épocas para el Modelo 7.

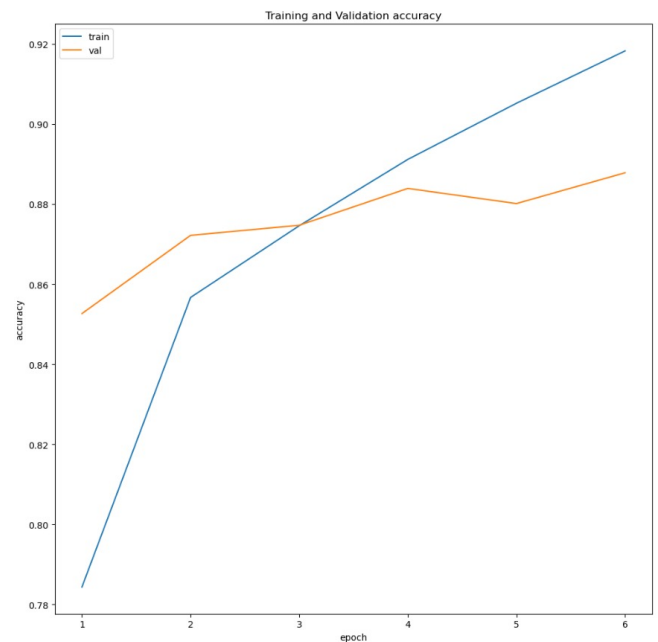


Figura 12. Historia de precisión a lo largo de las épocas para el Modelo 7.

Tras preparar la data con el embedding de GloVe, este modelo utiliza una capa LSTM para procesar la secuencia de entrada, aprendiendo y representando características relevantes de las palabras en el contexto de la secuencia. Seguidamente el modelo implementa tres capas de CNN con diferentes cantidades de filtros y tamaños de kernels, para buscar patrones locales y características a diferentes niveles de profundidad en

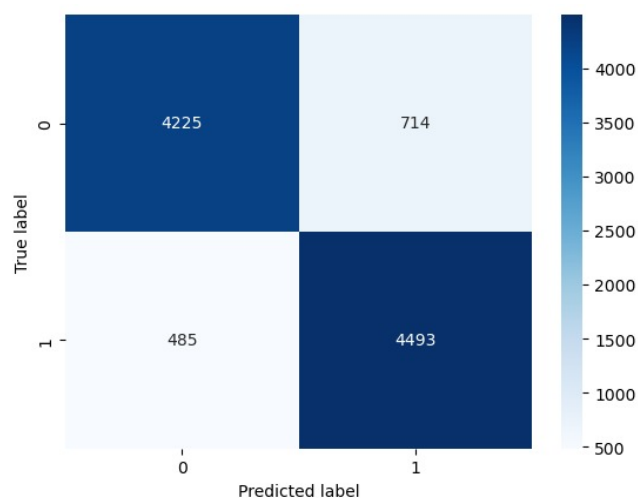


Figura 13. Resultados de predicciones para el Modelo 7.

las reseñas. Las consiguientes capas de LSTM ayudan a capturar aún más información contextual, reforzando la capacidad del modelo para modelar relaciones a largo plazo en la secuencia. Y al final el modelo toma a sus capas densas, la primera con múltiples neuronas actuando como un clasificador intermedio en el modelo y la segunda de una neurona siendo la capa final, que con la función sigmoide produce como salida una predicción de sentimiento en 0s y 1s.

Modelos de Machine learning utilizados en el pasado han utilizado Naive Bayes-based Classification models con un 87% en exactitud [3]. En contraste, nuestro modelo desarrollado en este artículo consigue una exactitud del 87.91%, lo que representa un ligero incremento respecto al modelo propuesto por [3]. Dicha diferencia de resultados indica áreas de mejora dentro del entrenamiento, o bien del diseño del modelo de machine learning utilizado. Es posible, además, que se deba a los datos utilizados para el entrenamiento y testing del modelo, que, por lo tanto, cambian por completo las técnicas de preprocesamiento y, por consiguiente, la calidad del input.

Entre las posibles áreas de mejora de nuestro modelo desarrollado aquí, destacan:

- La experimentación con diferentes parámetros de EarlyStopping y un mayor número de épocas, junto a la variación en un rango cercano del actual Learning Rate y cantidad de Dropout. A partir de la gráfica de historia de pérdida del modelo se notó un constante aumento en la pérdida hasta frenar por el EarlyStopping de “val_acc” (precisión de validación) en la época número seis. Optimizar estas modificaciones podría permitir al modelo romper esa barrera y continuar aumentando su precisión.
- Experimentación con embeddings estáticos avanza-

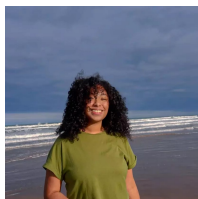
dos, como GPT-2 a GPT-4, y con embeddings dinámicos, particularmente con transformers, ya que estos contienen un gran potencial y actualmente siguen en mejora. Podrían proporcionar asimismo mejoras al modelo propuesto.

Dicha interpretación de resultados nos permite concluir que, si bien los datos del modelo podrían mejorar, fungen como base para continuar refinando las técnicas de machine learning en el futuro y mejorar la toma de decisiones respecto a qué métodos de entrenamiento deben utilizarse para el análisis de sentimientos.

Finalmente, ser capaces de interpretar la opinión general dada una base de datos de reseñas de películas permitirá a las partes interesadas determinar con mayor facilidad la opinión general del público espectador de películas (en este caso) pero bien podrían aplicarse técnicas similares para permitir la identificación de reseñas de otro tipo, por ejemplo: de productos o servicios.

Referencias

- [1] Google Cloud. (2023). What is Artificial Intelligence (AI)?. Accessed: May 1, 2023. [Online]. Available: <https://cloud.google.com/learn/what-is-artificial-intelligence>
- [2] K. Dashtipour, M. Gogate, A. Adeel, H. Larijani, and A. Hussain, “Sentiment Analysis of Persian Movie Reviews Using Deep Learning”, *Entropy*, vol. 23, no. 5, 2021.
- [3] M. A. Jassim, D. H. Abd, and M. N. Omri, “Machine learning-based new approach to films review”, *Social Network Analysis and Mining*, vol. 13, no. 1, p. 40, Mar. 2023.
- [4] D. Khurana, A. Koli, K. Khatter, and S. Singh, “Natural language processing: state of the art, current trends and challenges”, *Multimedia Tools and Applications*, vol. 82, no. 3, pp. 3713–3744, Jan. 2023.
- [5] R. Socher et al., “Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank”, in *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, 2013, pp. 1631–1642.
- [6] A. Santoro et al., “Relational recurrent neural networks”, in *Advances in Neural Information Processing Systems*, 2018, vol. 31.
- [7] S. Yu, S. R. Indurthi, S. Back, and H. Lee, “A Multi-Stage Memory Augmented Neural Network for Machine Reading Comprehension”, in *Proceedings of the Workshop on Machine Reading for Question Answering*, 2018, pp. 21–30.
- [8] IMDb. (2023). What is IMDb?. Accessed: May 1, 2023. [Online]. Available: <https://help.imdb.com/article/imdb/general-information/what-is-imdb/G836CY29Z4SGNMK5#>
- [9] R. K. Behera, M. Jena, S. K. Rath, and S. Misra, “Co-LSTM: Convolutional LSTM model for sentiment analysis in social big data”, *Information Processing & Management*, vol. 58, no. 1, p. 102435, 2021.
- [10] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning”, *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015.
- [11] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.
- [12] A. Graves, *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer Berlin Heidelberg, 2012.



instituciones.

AYLIN CAMACHO estudiante de último semestre en Ingeniería en Tecnologías Computacionales en el Tecnológico de Monterrey. Durante sus estudios universitarios ha participado en proyectos de diferentes áreas de TI, enfocados principalmente en administración de proyectos como project manager, ideación y prototipado de software, diseño de experiencia e interfaces de usuario y desarrollo de software para empresas e



de compiladores. Similarmente cuenta con experiencia previa en computación cuántica, con una publicación pendiente en el área de QNLP.

AXEL VILLANUEVA estudia su último semestre del plan de estudios Ingeniería en Tecnologías Computacionales en el Instituto Tecnológico y de Estudios Superiores de Monterrey. Durante su carrera ha tenido la oportunidad de trabajar como practicante en Facebook, Google y Microsoft, en áreas de la ingeniería de software como aplicaciones móviles, ingeniería de fiabilidad del sitio, desarrollo de software e infraestructura

...



LUIS CANO es Analista de Datos SEO en Envato y estudiante de Ingeniería en Tecnologías Computacionales en el Tecnológico de Monterrey con concentración en Ciencia de Datos. Sus intereses de investigación incluyen ciencia de datos, inteligencia artificial, redes neuronales, SEO, marketing digital y programación.



VALERIA CONDE es una futura practicante en ingeniería de software en Lyft. Durante su trayectoria profesional ha estudiado las carreras de Ingeniería en Tecnologías Computacionales y Licenciatura en Derecho. Ha trabajado previamente en Microsoft como practicante en dos ocasiones, contribuyendo al famoso entorno de desarrollo integrado Visual Studio y al compilador de C++ usado en la misma empresa.



experiencia en programación y su conocimiento de la teoría detrás del machine learning, el autor ha contribuido significativamente en la creación y mejora de aplicaciones inteligentes.

ERICK GOMEZ es un Ingeniero en Tecnologías Computacionales graduado del Instituto Tecnológico de Estudios Superiores de Monterrey. Como estudiante de tiempo completo, ha enfocado gran parte de su experiencia en el desarrollo de aplicaciones móviles y en el manejo de bases de datos. Recientemente, ha incursionado en el desarrollo de aplicaciones avanzadas que utilizan algoritmos de machine learning. Con su