

# Clasificación de Reviews en Películas usando Redes Neuronales Recurrentes

1<sup>st</sup> Santiago Andrés Rocha, 2<sup>nd</sup> Sebastián Rojas Bueno

*Escuela Colombiana de Ingeniería Julio Garavito, Bogotá, Colombia*

santiago.rocha-c@mail.escuelaing.edu.co, sebastian.rojas-bu@mail.escuelaing.edu.co

**Resumen**—En este artículo se presenta un proyecto que abordó el desafío de clasificar reviews de películas en dos categorías, 'Review positiva' y 'Review negativa', mediante el uso de redes neuronales recurrentes (RNN). Se combinaron técnicas de procesamiento de lenguaje natural (NLP) con las capacidades de las RNN para lograr una alta exactitud de clasificación y tiempos de ejecución eficientes.

El proyecto siguió un enfoque en el que se tomaron reviews de películas previamente etiquetadas en un dataset conocido, las cuales fueron utilizadas para entrenar y evaluar el modelo. Se aplicaron técnicas de preprocesamiento de datos para ajustar el texto, así como para convertir las palabras en vectores numéricos utilizando técnicas de representación vectorial como Word Embeddings.

La construcción del modelo de RNN se realizó mediante el uso de capas recurrentes con ayuda del framework Keras, permitiendo el procesamiento secuencial de las reviews. Mediante experimentación y ajuste de los hiperparámetros del modelo, se logró alcanzar una **exactitud** de clasificación **superior al 85 %**.

Los **tiempos de ejecución** también fueron considerados durante la experimentación, y se logró mantenerlos por debajo de **1 hora**, lo que asegura una respuesta rápida en la clasificación de reviews de películas.

Los resultados obtenidos demuestran que el enfoque propuesto es efectivo para la clasificación de reviews de películas, proporcionando una solución precisa y eficiente gracias al cumplimiento de las métricas optimizadas y alcanzadas. Este proyecto y en particular el modelo creado tiene potenciales aplicaciones en la industria del entretenimiento y en plataformas de recomendación de películas, mejorando la experiencia de los usuarios al ofrecer análisis automatizados y rápidos sobre la polaridad de las reviews.

**Index Terms**—Problema de clasificación, redes neuronales recurrentes, reviews de películas, entrenamiento, validación, pruebas.

## I. INTRODUCCIÓN

En la era digital actual, la abundancia de información y la rápida expansión de las plataformas de reseñas en línea han llevado a un crecimiento exponencial en la cantidad de opiniones y evaluaciones de productos y servicios. En particular, las reviews de películas han adquirido una relevancia significativa, ya que los usuarios buscan orientación para tomar decisiones informadas sobre qué películas ver. La influencia de estas reviews en las decisiones de visualización es innegable, ya que muchas personas confían en las opiniones de otros para decidir si ver o no una película en particular.

Sin embargo, procesar y analizar manualmente un gran volumen de reseñas se vuelve cada vez más difícil y consume mucho tiempo. Además, la calidad de la clasificación de

estas reviews puede tener un impacto significativo tanto para los usuarios como para los productores de películas. Una clasificación precisa y confiable puede ayudar a los usuarios a identificar rápidamente películas de su interés y mejorar su experiencia de visualización, mientras que para los productores, conocer la percepción de los espectadores puede ser invaluable para tomar decisiones informadas sobre futuros proyectos y ajustar su estrategia de marketing.

Para abordar este desafío, se ha explorado el uso de técnicas de procesamiento de lenguaje natural (NLP) y aprendizaje automático para la clasificación automatizada de reviews de películas. En particular, el enfoque de las redes neuronales recurrentes (RNN) ha demostrado ser prometedor debido a su capacidad para modelar el contexto secuencial y capturar las relaciones a largo plazo en el texto.

En las siguientes secciones, describiremos en detalle nuestra metodología de trabajo, los experimentos realizados, los resultados obtenidos y discutiremos las implicaciones de nuestro enfoque en el contexto de la clasificación de reviews de películas. Nuestro objetivo final es proporcionar una solución precisa y eficiente que mejore la experiencia de los usuarios y brinde información valiosa a los productores en la industria del cine.

## II. ESTADO DEL ARTE

### II-A. Marco Teórico

La clasificación de reviews de películas es un campo de estudio que ha experimentado diversas metodologías a lo largo del tiempo. En sus inicios, la clasificación se realizaba de forma manual, donde los expertos en el dominio leían y analizaban individualmente cada review para asignarles una etiqueta de 'positiva' o 'negativa'. Este enfoque, aunque proporcionaba resultados precisos, era laborioso, lento y limitado en términos de escalabilidad.

Con el avance de la tecnología y el crecimiento explosivo de las plataformas en línea, se han desarrollado enfoques automatizados para la clasificación de reviews de películas. Estos enfoques se basan en técnicas de procesamiento de lenguaje natural (NLP) y aprendizaje automático, permitiendo el análisis rápido y eficiente de grandes volúmenes de reviews. Algunas de estas técnicas incluyen el uso de algoritmos de aprendizaje supervisado, como Naive Bayes o Support Vector Machines (SVM), que se entrenan utilizando características extraídas de las reviews, como la frecuencia de palabras clave o la presencia de sentimientos positivos o negativos.

Sin embargo, a pesar de los avances logrados con estas técnicas automatizadas, la capacidad de capturar el contexto y las relaciones a largo plazo en el texto de las reviews se ha convertido en un desafío. Aquí es donde entran en juego las redes neuronales recurrentes (RNN). Las RNN son un tipo de modelo de aprendizaje automático que se especializa en el procesamiento de secuencias y datos con dependencias temporales. A diferencia de los enfoques tradicionales, las RNN tienen la capacidad de aprender y recordar información a lo largo de la secuencia de las reviews de películas, capturando así las relaciones y matices sutiles que influyen en la polaridad de las mismas.

Las RNN se han utilizado con éxito en la clasificación de reviews de películas, especialmente mediante el uso de capas de unidades LSTM (Long Short-Term Memory) o GRU (Gated Recurrent Unit). Estas capas permiten el procesamiento secuencial de las reviews y la captura de las dependencias temporales. Mediante el entrenamiento de las RNN con grandes conjuntos de datos etiquetados, es posible lograr altos niveles de exactitud en la clasificación de las reviews.

Además de la precisión en la clasificación, las RNN también ofrecen la ventaja de la escalabilidad y la eficiencia en términos de tiempo de ejecución. Esto es especialmente importante en el contexto de las plataformas en línea, donde las reviews de películas se generan constantemente y es necesario procesarlas rápidamente para brindar información relevante a los usuarios y productores.

La estructura básica de una RNN consta de unidades recurrentes que procesan secuencialmente la entrada y mantienen una memoria interna o estado oculto. Esta memoria permite a la red retener información contextual sobre las palabras anteriores y utilizarla para influir en la predicción actual. Al incorporar la retroalimentación recurrente, las RNN pueden capturar relaciones a largo plazo y comprender el contexto global de una secuencia [1].

Las RNN han sido ampliamente utilizadas en el campo del procesamiento de lenguaje natural para tareas como la traducción automática, la generación de texto y la clasificación de sentimientos [2]. En particular, en la clasificación de reviews de películas, las RNN pueden capturar las sutilezas del lenguaje y comprender el tono y la intención detrás de las opiniones expresadas.

Históricamente, las RNN se remontan a la década de 1980, con el desarrollo del modelo de red neuronal recurrente estándar, conocido como Elman network. Sin embargo, las RNN tradicionales tenían dificultades para capturar relaciones a largo plazo debido al problema del desvanecimiento o explosión del gradiente. Esto llevó al desarrollo de variantes más sofisticadas de las RNN, como las previamente mencionadas LSTM y las GRU, que han demostrado ser más efectivas en el procesamiento de secuencias largas [3].

Para implementar redes neuronales recurrentes (RNN) en el procesamiento de lenguaje natural y la clasificación de reviews de películas, un framework ampliamente utilizado es Keras. Keras es una biblioteca de código abierto escrita en Python que proporciona una interfaz de alto nivel para construir y entrenar redes neuronales de manera eficiente. Además, proporciona una variedad de capas de procesamiento

de texto, como `.Embedding` para representar palabras en forma vectorial para su mejor procesamiento [4].

Es relevante mencionar el optimizador 'RMSprop' y la función de pérdida 'Binary Crossentropy', dos componentes esenciales en el entrenamiento de redes neuronales recurrentes (RNN) en Keras, y componentes importantes en la implementación de los modelos que posteriormente explicaremos.

- Optimizador RMSprop: RMSprop (Root Mean Square Propagation) es otro algoritmo popular de optimización utilizado en el entrenamiento de redes neuronales. Al igual que Adam, RMSprop también se adapta a los cambios en la tasa de aprendizaje durante el proceso de entrenamiento. RMSprop utiliza una estimación adaptativa de los momentos del gradiente para actualizar los pesos del modelo. A diferencia de Adam, RMSprop solo utiliza el momento de primer orden (gradiente), lo que lo hace especialmente adecuado para problemas con gradientes ruidosos o no estacionarios. Ajusta automáticamente la tasa de aprendizaje para cada parámetro individual en función del historial de los gradientes pasados [5].
- Pérdida Binary Crossentropy: La pérdida `binary_crossentropy`, también conocida como entropía cruzada binaria, es una función de pérdida comúnmente utilizada en problemas de clasificación binaria, como la clasificación de reviews de películas en positivas o negativas. Esta función de pérdida mide la discrepancia entre las etiquetas verdaderas y las predicciones del modelo. Cuanto menor sea el valor de la pérdida `binary_crossentropy`, más cercanas serán las predicciones del modelo a las etiquetas verdaderas [6].

## II-B. Trabajo Relacionado

En cuanto a trabajo relacionado, es importante hablar tanto de datasets y conjuntos de datos relacionados y usados en este tipo de problemas, así como de soluciones tecnológicas que se han dado y se encuentran de forma pública. Hablaremos de ambos aspectos a continuación:

- Datasets: El conjunto de datos IMDB de Keras es un popular conjunto de datos utilizado para la clasificación de textos en el campo del procesamiento de lenguaje natural. Este conjunto de datos se basa en el conjunto de datos de la Internet Movie Database (IMDB) y contiene opiniones (reviews) de películas etiquetadas como positivas o negativas [7]. El conjunto de datos IMDB está equilibrado, con un total de 50,000 reviews, de las cuales 25,000 son etiquetadas como positivas y 25,000 como negativas. Está dividido en un conjunto de entrenamiento y un conjunto de prueba, con 25,000 reviews en cada uno. Cada review en el conjunto de datos IMDB se representa como una secuencia de palabras, donde cada palabra se codifica como un número entero que corresponde a su índice en un diccionario predefinido de palabras. Las reviews tienen una longitud variable, lo que implica que se debe realizar un procesamiento adicional, como el ajuste de la longitud o el truncamiento de las secuencias

de palabras, para poder alimentarlas a un modelo de aprendizaje automático.

El objetivo de utilizar el conjunto de datos IMDB en la clasificación de reviews de películas es construir un modelo de aprendizaje automático capaz de predecir con precisión si una review es positiva o negativa en función del texto proporcionado.

El conjunto de datos IMDB de Keras es ampliamente utilizado como punto de referencia en investigaciones y experimentos relacionados con el procesamiento de lenguaje natural y la clasificación de textos.

**Este dataset de Keras será el que usaremos y con el que trabajamos .**

Además del conjunto de datos IMDB, existen otros conjuntos de datos destacados en el ámbito de las reviews de películas. Algunos de ellos incluyen:

1. Rotten Tomatoes [8]: Rotten Tomatoes es un sitio web popular que recopila opiniones de críticos profesionales sobre películas. Proporciona un conjunto de datos de reviews de películas etiquetadas como positivas o negativas. Estos datos se han utilizado en la clasificación de sentimientos y la predicción de calificaciones de películas.
2. Amazon Reviews [9]: Amazon es uno de los principales minoristas en línea y también recopila opiniones de usuarios sobre productos, incluyendo películas. Los datos de reviews de películas de Amazon proporcionan una fuente rica de información para tareas de clasificación y análisis de sentimientos.
3. MovieLens [10]: MovieLens es una plataforma de recomendación de películas que recopila datos de evaluación y opiniones de usuarios. Además de las calificaciones, MovieLens también puede proporcionar información textual en forma de reseñas o comentarios de los usuarios, lo que permite la construcción de conjuntos de datos de reviews de películas.

- Soluciones Relacionadas: En cuanto a soluciones relacionadas, hablaremos sobre 2 trabajos particulares que destacan por su acercamiento a la solución haciendo uso de redes neuronales no necesariamente recurrentes:

1. Deeply Moving: Deep Learning for Sentiment Analysis (2013) [11]

Es un artículo de investigación que se centra en la aplicación de técnicas de aprendizaje profundo para el análisis de sentimientos, específicamente en el ámbito de las críticas de películas. El artículo explora el uso de Redes Neuronales Convolucionales (CNN) y Redes Neuronales Recurrentes (RNN) para clasificar las críticas de películas como positivas o negativas según su sentimiento.

Los autores comienzan discutiendo la importancia del análisis de sentimientos en el contexto de las críticas de películas, ya que muchas personas confían en estas opiniones para tomar decisiones sobre qué películas ver. Luego, presentan los fundamentos teóricos del aprendizaje profundo y describen cómo las CNN y las RNN pueden ser utilizadas para capturar características

relevantes en las secuencias de texto y realizar la clasificación de sentimientos. Se informó una precisión de clasificación de aproximadamente el 85 % en la tarea de distinguir críticas positivas de negativas.

2. Aspect Based Sentiment Analysis of Movie Reviews using Convolutional Neural Network (2017) [12]

Es un artículo que se centra en la aplicación del análisis de sentimiento basado en aspectos en las críticas de películas utilizando redes neuronales convolucionales (CNN). El objetivo del estudio es analizar las opiniones de los usuarios sobre diferentes aspectos de las películas, como el guion, la actuación o la cinematografía, y determinar si son positivas o negativas.

En el artículo se destaca la importancia de comprender los aspectos específicos que influyen en las opiniones de las películas, ya que los usuarios a menudo expresan diferentes sentimientos hacia diferentes elementos de una película. Los autores proponen el uso de una CNN para identificar y clasificar automáticamente los aspectos en las críticas de películas, lo que permite un análisis más detallado y preciso de los sentimientos asociados con cada aspecto. Se informó una precisión de clasificación de aproximadamente el 90.83 % en la tarea de distinguir críticas positivas de negativas.

**Nota:** Aunque estos trabajos relacionados se encargan de clasificar reviews de películas, es importante aclarar que entre ellos, y en comparación con nosotros, no se usa en ningún caso el mismo dataset, sino que cada uno utiliza uno diferente e independiente con distintas características e información.

### III. METODOLOGÍA

La metodología para el desarrollo de la solución se planteó de la siguiente forma:

1. Seleccionar el dataset a trabajar.
2. Pensar en el tipo de problema.
3. Analizar el tipo de dato a utilizar como fuente.
4. Considerar una solución con redes neuronales.
5. Considerar qué tipo de red neuronal convendría más.
6. Buscar trabajo relacionado para ayudarnos a comenzar.
7. Empezar a desarrollar la solución.

A continuación, describiremos nuestro desarrollo para cada una de estas etapas en nuestra metodología:

1. Seleccionar el dataset a trabajar:  
En esta etapa, se eligió el conjunto de datos que se utilizaría para entrenar el modelo de clasificación de objetos de moda. Se seleccionó un conjunto de datos lo suficientemente grande y diverso para permitir que el modelo tuviese la capacidad de entrenarse correctamente.
2. Pensar en el tipo de problema:  
Se definió el problema de clasificación de reviews de películas, es decir, el objetivo del proyecto era desarrollar un modelo de clasificación que pueda tomar las reseñas y determinar si son positivas o negativas con respecto al material audiovisual al que van dirigidas.
3. Analizar el tipo de dato a utilizar como fuente:

Se analizó el tipo de datos que se utilizarían como fuente para el modelo de clasificación. En este caso, se utilizaron cadenas de texto, reseñas extensas y detalladas, lenguaje natural.

4. Considerar una solución con redes neuronales:  
Se decidió utilizar una solución basada en redes neuronales para abordar el problema de clasificación de objetos de moda. Esto se debió a que las redes neuronales han demostrado ser eficaces en problemas de clasificación.
5. Considerar qué tipo de red neuronal convendría más:  
Se analizaron diferentes tipos de redes neuronales y se decidió utilizar una red neuronal recurrente en combinación con una capa de red neuronal simple (SNN). Esto se debió a que las redes neuronales recurrentes han demostrado ser eficaces para el procesamiento de lenguaje natural, y las simples eficaces para la clasificación.
6. Buscar trabajo relacionado para ayudarnos a comenzar:  
Se investigaron trabajos relacionados para obtener ideas y conocimientos sobre el desarrollo de un modelo de clasificación de reviews utilizando redes neuronales recurrentes, encontrando la ayuda de la documentación de Keras [7]. De esta investigación determinamos la **métrica a optimizar de Exactitud**, que debe ser **superior al 85 %**, y la **métrica a Satisfacer de Tiempo** de ejecución **inferior a 1 hora**.
7. Empezar a desarrollar la solución:  
Finalmente, se comenzó a desarrollar la solución utilizando una RNN. Para esto y con ayuda tecnológica, se crearía un modelo de clasificación en Python, principalmente con la ayuda del framework Keras y otras herramientas para su análisis y visualización.

Una vez se terminen estos pasos, y a modo de experimentación, se plantean diferentes modelos similares con cambios sutiles para validar el mejor de entre los mismos. Más adelante analizaremos y explicaremos cada uno de los modelos planteados, y los resultados obtenidos, determinando cuál resulta ser mejor con base en nuestras métricas de Exactitud y Tiempos de Ejecución.

Nuestra idea de solución finamente sería la implementación de una Red Neuronal Recurrente, combinada con una Red Neuronal Simple, para la clasificación de Reviews de Películas, partiendo de texto y llegando a decir si las reseñas son Positivas o Negativas, haciendo uso de Python y en particular, el framework Keras.

Posteriormente, ya teniendo planteada la idea, se procedería con los siguientes pasos, que nos permitirían ya a nivel práctico jugar con el(los) modelos de solución, ajustarlos, experimentar y demás.

1. Análisis de los datos.
2. Diseñar el modelo de red neuronal.
3. Pruebas y análisis de resultados.
4. Optimización / Regularización.

Con el camino claro, procedemos entonces con la construcción de nuestra idea de solución.

#### IV. SOLUCIÓN

Nuestra solución consiste en la implementación de nuestra idea de solución en Python, con ayuda de librerías como

Numpy, Matplotlib y, por su puesto, Keras. Siguiendo con los pasos previamente dados, analizaremos el proceso que llevamos a la hora de llevarlos a cabo.

1. Análisis de los datos: Primero, y usando el Dataset 'IMDB movie review sentiment classification dataset' realizamos un preprocesamiento de los datos donde:

- Analizamos las entradas y distribución de los datos, verificando que de las 50000 críticas presentes, 25000 de ellas están etiquetadas como 'positivas' y las otras 25000 como 'negativas'. Adicionalmente, están distribuidas en dos grupos grandes, Entrenamiento y Pruebas con 12500 reseñas cada uno.

Cuadro I  
DIVISIÓN DE DATOS

Etapas	% de Datos	No. de Datos	Positivas	Negativas
Train	50 %	25000	12500	12500
Test	50 %	25000	12500	12500

- No es necesario reorganizar las entradas del dataset, pues se encuentran ya en 'desorden'.
- No es necesario ajustar los datos para que puedan ser tomados como entradas válidas para la red neuronal a implementar, pues estas se encuentran codificadas y listas para los modelos de keras.  
Decodificando la primera de las reseñas, y a modo de ejemplificación, enseñaremos la primera de las reseñas donde se puede apreciar las cadenas de caracteres que componen cada una de las entradas:  
'? this film was just brilliant casting location scenery story direction everyone's really suited the part they played (...)'
- Nuevo agrupamiento de datos, necesarios para el conjunto de Validación: Del 50 % de datos de pruebas, se toma el 50 % para Validación y el resto permanece en Pruebas, de forma que los datos quedarían distribuidos de la siguiente forma:

Cuadro II  
DIVISIÓN DE DATOS CONJUNTO DE PRUEBAS

Etapas	% de Datos	No. de Datos	Positivas	Negativas
Valid	50 %	12500	6184	6316
Test	50 %	12500	6316	6184

Con ayuda de Mathplotlib, podemos verificar esta distribución de los datos a nivel de código en la figura 1

2. Diseñar el modelo de red neuronal: Puestos ya en la implementación del modelo, tendremos en cuenta las características dadas en la página del dataset de Keras. El modelo a destacar tiene las siguientes configuraciones":

##### Modelo 1:

- Tipo de Modelo: Secuencial
- Entradas: Texto codificado en Dicionarios
- Capa Embedding: Máximo número de características, tamaño 32
- Capa Recurrente: Función de Activación Interna: Tanh, Función de Activación Salida: Softmax, 32 Neuronas

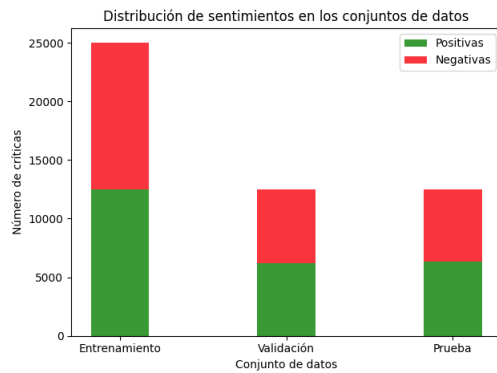


Figura 1. Distribución de datos

- Capa Densa de Salida: 1 neurona - Función de Activación: Sigmoid
- Épocas: 10
- Batch: 128
- Costo: Entropía cruzada Binaria
- Optimizador: rmsprop
- Métricas: Exactitud 85 %
- Métricas: Tiempo de Ejecución: 1 hora

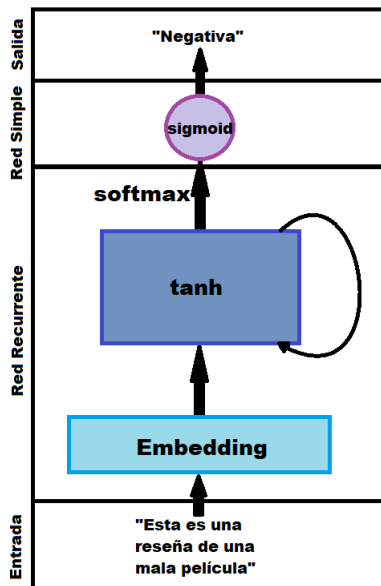


Figura 2. Arquitectura de la Red

Para tener modelos contrastantes, se planteó otros 2 modelos con variaciones sobre el original. Las variaciones actúan sobre el tamaño de la capa de Embedding y sobre el número de neuronas de la capa Recurrente de la siguiente forma, verificable también en las figuras 3, 4 y 5:

El objetivo de estos cambios fue verificar la eficacia del modelo suministrado y determinar si podemos hacer mejoras al mismo en cuanto a tiempos de ejecución, exactitud o pérdidas.

- Entrenamiento de los modelos: Para entrenar los modelos, fueron usados los datos destinados a Entrenamiento, y con

Cuadro III  
DIFERENCIAS ENTRE LOS MODELOS

Modelos	Tam. Embedding	No. Neuronas Capa Recurrente
Modelo 1	32	32
Modelo 2	64	64
Modelo 3	32	16

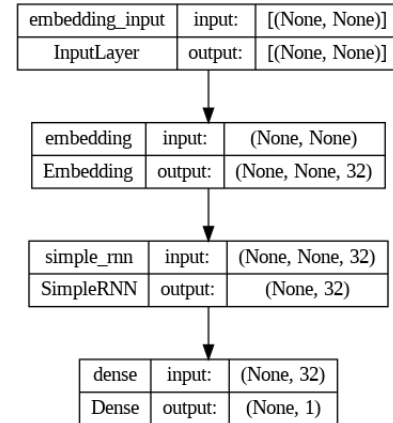


Figura 3. Arquitectura del modelo 1

las variaciones previamente descritas, se obtuvo distintos resultados a tener en cuenta.

- Evaluación del modelo: De acuerdo con los siguientes parámetros propuestos:

- Función de pérdida/costo: Entropía Cruzada Binaria
- Métrica: Precisión superior al **85 %**.
- Métrica: Tiempo de ejecución no mayor a **1 hora**.

Modelo Seleccionado:

Tercer Modelo:

Cuadro IV  
CARACTERÍSTICAS DEL MODELO SELECCIONADO

Modelos	Tam. Embedding	No. Neuronas Capa Recurrente
Modelo 3	32	16

Con los siguientes resultados:

Modelo 3, Entrenamiento:

Cuadro V  
DIFERENCIAS ENTRE LOS MODELOS

Etapas	Pérdida	Exactitud	Tiempo
Train	0.1442	95.06 %	5 min aprox
Valid	0.3975	85.75 %	5 min aprox

Más adelante enseñaremos todos los resultados obtenidos, y podremos justificar finalmente la selección de este modelo, pero adelantamos que cumple con las métricas establecidas, teniendo mejores resultados que los demás, y tiene comportamientos adecuados a lo que se esperaría de una red neuronal recurrente para un problema de clasificación.

- Pruebas y análisis de resultados: Enseñaremos a continuación los resultados que obtuvimos, teniendo como



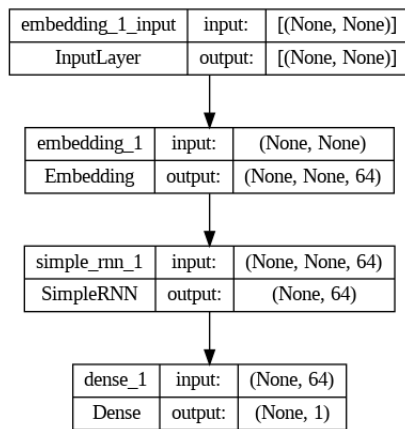


Figura 4. Arquitectura del modelo 2

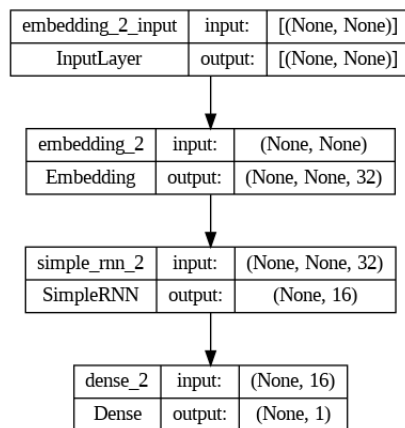


Figura 5. Arquitectura del modelo 3

prioridad enseñar la validez de la construcción y su eficacia para la entrada de los resultados.

## V. RESULTADOS

Los resultados obtenidos fueron los siguientes: Se evaluaron los modelos y se compararon sus resultados de Pérdida y se pueden verificar en la gráfica 6.

**Nota:** Para los gráficos enseñados, se tiene como valores en el eje X las épocas de ejecución de los modelos, mientras en el eje Y tenemos los valores de pérdida, para cada uno de los modelos.

Ahora, al momento de evaluar tanto exactitudes como tiempos, los resultados obtenidos fueron los siguientes:

Cuadro VI  
COMPARACIÓN RESULTADOS ENTRENAMIENTO

Modelos	Train Loss	Train Acc
Modelo 1	0.0608	98.00 %
Modelo 2	0.0116	99.65 %
Modelo 3	0.1442	95.06 %

Los mejores resultados en entrenamiento los posee el Modelo 2, con una exactitud del 99.65 % mientras que el Modelo queda en 'último lugar' con una exactitud del 95.06 %.

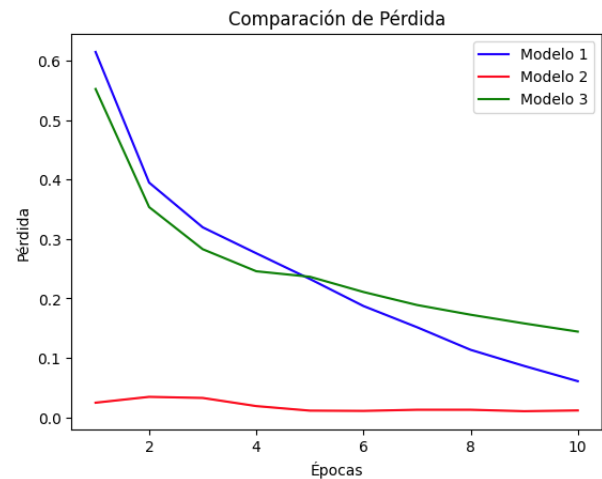


Figura 6. Comparación de Pérdidas

Cuadro VII  
COMPARACIÓN RESULTADOS VALIDACIÓN

Modelos	Val Loss	Val Acc
Modelo 1	0.5713	83.06 %
Modelo 2	1.1635	78.82 %
Modelo 3	0.3975	85.75 %

Los mejores resultados en validación los posee el Modelo 3, con una exactitud del 85.75 % mientras que el Modelo queda en 'último lugar' con una exactitud del 78.82 %. Esto es un indicativo de que el Modelo 2 tal como se presenta está sobre entrenado, teniendo problemas de exactitud en datos que no son los de su propio entrenamiento.

Cuadro VIII  
COMPARACIÓN TIEMPOS DE EJECUCIÓN

Modelos	Tiempo
Modelo 1	12 min aprox
Modelo 2	14 min aprox
Modelo 3	10 min aprox

Finalmente, el Modelo 3 destaca por su eficiencia en cuanto a tiempos de ejecución, cuando el Modelo 2 es el más demorado de los tres por un margen de 2 min aprox de diferencia.

En síntesis, aunque el Modelo 3 no posea los mejores resultados en cuanto a pérdidas, tiene el menor tiempo de ejecución y tiene el mejor porcentaje de Exactitud de entre todos los modelos, siendo además el único capaz de cumplir con la métrica de Exactitud superior al 85 %. Estos son los motivos por los cuales es el modelo seleccionado tal como se mencionó en la sección anterior.

Una vez entendida y justificada la decisión, podemos entonces probar el Modelo 3 con el conjunto de Pruebas, y verificar su desempeño por medio de una Matriz de confusión en la que de forma evidente, veremos cómo predominan los Verdaderos Positivos (reviews positivas clasificadas correctamente como positivas) y los Verdaderos Negativos (reviews negativas clasificadas correctamente como negativas). Ver figura 7.

Cuadro IX  
RESULTADOS MODELO 3 EN PRUEBAS

Modelos	Test Loss	Test Acc
Modelo 3	0.38	86.24 %

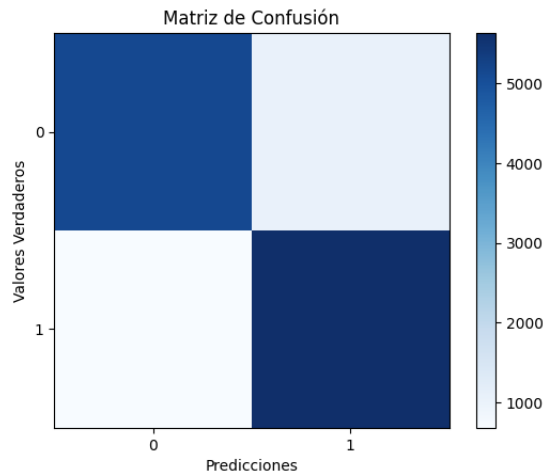


Figura 7. Matriz de Confusión para conjunto de Pruebas, Modelo 3

La figura 7 por la intensidad de colores, de azul claro a azul oscuro, nos muestra cómo se encuentran distribuidos los datos clasificados por la red implementada vs dónde realmente deberían estar. Normalmente en una matriz de confusión, la diagonal principal es donde se encuentran los Verdaderos Positivos y los Verdaderos Negativos, no siendo esta la excepción. Vemos que más de 5500 de las reseñas están correctamente clasificadas como positivas, y más de 5500 como negativas.

De esta forma, se construyó y evaluó varios modelos de redes neuronales recurrentes aplicados a problemas de clasificación, llegando uno a cumplir con las métricas establecidas y arrojar resultados favorables para el desarrollo de sistemas inteligentes, y resultados favorables para el posible uso de herramientas de ayuda para profesionales en la clasificación reseñas de películas.

## VI. CONCLUSIONES

Específicamente, la red neuronal recurrente definitiva posee la siguiente configuración:

### Tercer modelo:

- Tipo de Modelo: Secuencial
- Entradas: Texto codificado en Diccionesarios
- Capa Embedding: Máximo número de características, tamaño 32
- Capa Recurrente: Función de Activación Interna: Tanh, Función de Activación Salida: Softmax, 16 Neuronas
- Capa Densa de Salida: 1 neurona - Función de Activación: Sigmoid
- Épocas: 10
- Batch: 128
- Costo: Entropía cruzada Binaria
- Optimizador: rmsprop
- Métricas: Exactitud 85 %

- Métricas: Tiempo de Ejecución: 1 hora

Con los resultados vistos en el Cuadro X.

Cuadro X  
RESULTADOS GENERALES MODELO 3

Etapas	Loss	Accuracy
Training	0.1442	95.06 %
Validation	0.3975	85.75 %
Test	0.38	86.24 %

A través de la metodología propuesta, el desarrollo del modelo de RNN y la experimentación llevada a cabo, se obtuvieron los siguientes hallazgos y conclusiones:

1. Metodología eficaz: La metodología propuesta, el orden y la rigurosidad con la que planteamos y desarrollamos la idea de solución resultó crucial para los resultados aquí obtenidos, y el buen entendimiento del marco teórico de las redes recurrentes nos permitió crear modelos capaces de comprender el sentimiento expresado en las reviews.
2. Desarrollo del modelo: El desarrollo del modelo de RNN fue un proceso clave en el estudio. El diseño de la estructura de la red, la configuración de los hiperparámetros y la elección adecuada del optimizador y la función de pérdida permitieron obtener un modelo robusto y preciso para la clasificación de sentimientos en las reviews de películas.
3. Experimentación rigurosa: Se llevó a cabo una experimentación rigurosa utilizando conjuntos de datos de entrenamiento y prueba adecuados. Esto garantizó la validez y confiabilidad de los resultados obtenidos. Se realizaron pruebas exhaustivas para ajustar los hiperparámetros y evaluar el rendimiento del modelo en diferentes escenarios.
4. Resultados prometedores: Los resultados obtenidos fueron altamente prometedores. El modelo de RNN logró una precisión superior al 85 %, lo cual demuestra su eficacia en la clasificación de reviews de películas en categorías de sentimientos. Además, cumple con el tiempo de ejecución propuesto, lo que indica un equilibrio entre la capacidad de identificar correctamente las reviews positivas y negativas.
5. Posibles aplicaciones: El enfoque propuesto en este estudio tiene múltiples aplicaciones potenciales. Además de la clasificación de reviews de películas, este modelo podría adaptarse y aplicarse en otras áreas del procesamiento de lenguaje natural, como el análisis de sentimientos en reseñas de productos, comentarios en redes sociales o análisis de opiniones en encuestas y cuestionarios.

## VII. TRABAJO FUTURO

Como trabajo futuro, nos gustaría abordar los siguientes tópicos que consideramos altamente relevantes:

1. Mejora del rendimiento del modelo: A pesar de los resultados prometedores obtenidos, aún existe margen para mejorar el rendimiento del modelo de clasificación de sentimientos. Se pueden explorar técnicas de ajuste de hiperparámetros más avanzadas, como la optimización

bayesiana, o considerar arquitecturas de redes neuronales más complejas, como las redes neuronales transformer, para obtener un rendimiento aún mejor.

2. Análisis de sentimientos en múltiples idiomas: Este artículo se centró en la clasificación de reviews de películas en un solo idioma. Un área de investigación futura podría ser la adaptación del modelo propuesto para realizar análisis de sentimientos en reviews de películas en diferentes idiomas. Esto requeriría la recopilación y anotación de conjuntos de datos en varios idiomas y la exploración de enfoques de transferencia de aprendizaje.
3. Incorporación de información contextual adicional: Además de las palabras utilizadas en las reviews de películas, hay información contextual adicional que puede ser relevante para el análisis de sentimientos, como la puntuación asignada a la película o la fecha de lanzamiento. Integrar esta información contextual adicional en el modelo de clasificación de sentimientos podría mejorar la precisión y la comprensión del sentimiento expresado en las reviews.
4. Enfoque en aspectos específicos: Si bien este artículo se centró en la clasificación general de sentimientos en las reviews de películas, un enfoque futuro interesante podría ser realizar un análisis de sentimiento basado en aspectos específicos. Esto implicaría identificar y clasificar los sentimientos asociados con aspectos particulares de una película, como la actuación, la dirección o los efectos visuales.
5. Incorporación de datos multimodales: Además del texto de las reviews, existen otros tipos de datos, como imágenes de escenas de películas o audio de diálogos, que pueden proporcionar información adicional para el análisis de sentimientos. Integrar datos multimodales en el modelo de clasificación de sentimientos podría enriquecer la comprensión del sentimiento expresado en las reviews y llevar a un análisis más completo.

Estas cinco direcciones de trabajo futuro ofrecen oportunidades para ampliar y mejorar la investigación realizada en este artículo, abriendo nuevas posibilidades en el campo del análisis de sentimientos en las críticas de películas utilizando redes neuronales recurrentes.

## REFERENCIAS

- [1] Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. arXiv preprint arXiv:1412.3555.
- [2] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. arXiv preprint arXiv:1406.1078.
- [3] Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8), 1735-1780.
- [4] Keras: biblioteca de código abierto para crear redes neuronales. (s/f). IONOS Digital Guide., de <https://www.ionos.es/digitalguide/online-marketing/marketing-para-motores-de-busqueda/que-es-keras/>
- [5] RMSprop. (s/f). Keras.Io., de <https://keras.io/api/optimizers/rmsprop/>
- [6] Probabilistic losses. (s/f). Keras.Io., de [https://keras.io/api/losses/probabilistic\\_losses/](https://keras.io/api/losses/probabilistic_losses/)
- [7] IMDB movie review sentiment classification dataset. (s/f). Keras.Io., de <https://keras.io/api/datasets/imdb/>
- [8] Leone, S. (2020). Rotten Tomatoes movies and critic reviews dataset [Data set].

- [9] Bittlingmayer, A. (2019). Amazon Reviews for Sentiment Analysis [Data set].
- [10] GroupLens. (2018). MovieLens 20M Dataset [Data set].
- [11] Deeply moving: Deep learning for sentiment analysis. (s/f). Ycombinator.com., de <https://news.ycombinator.com/item?id=6333664>
- [12] Onalaja, S., Romero, E., Yun, B., & Javed, F. (s/f). Aspect-based sentiment analysis of movie reviews aspect-based sentiment analysis of movie reviews. Smu.edu., de <https://scholar.smu.edu/cgi/viewcontent.cgi?article=1205&context=datasciencereview>