

Mejora de algoritmo de reconocimiento de emociones



Cecilia Jael Aguilar Aranda[†],
Alexander Espronceda Gómez[‡]
Satu Elisa Schaeffer



[†]Ingeniería en Administración de Sistemas, [‡]Ingeniería en Tecnologías de Software, Posgrado en Ingeniería de Sistemas

[†]cecilia.aguilarand@uanl.edu.mx, [‡]alexeg98@gmail.com

Resumen

En este proyecto se busca incrementar el funcionamiento de un algoritmo de análisis de sentimiento en base a texto utilizando redes neuronales. Esto conlleva realizar modificaciones al código, así como el aumento del dataset con el que se trabaja. El objetivo es realizar un chatbot capaz de reconocer patrones, determinar el sentimiento mostrado por el usuario y actuar acorde a ello.

1. Introducción

El análisis de sentimiento se refiere a la aplicación de procesamiento de lenguaje natural, lingüística computacional y análisis de texto para identificar la emoción del autor. [1]. En el presente documento se considera *Análisis de sentimiento y Reconocimiento de emociones* como un mismo concepto.

El objetivo principal de este proyecto es llevar a cabo diversos procesos que ayuden a que un algoritmo de análisis de sentimiento por medio de texto tenga un porcentaje de acierto más alto. Esto se logrará mediante búsqueda, análisis y filtrado de datos relevantes, así como también la actualización del algoritmo que está siendo utilizado actualmente.

2. Antecedentes

El algoritmo de análisis de sentimiento por medio de texto que se utiliza en este proyecto le corresponde a Alexander Espronceda Gómez, quien actualmente está escribiendo un paper al respecto [2]. Por lo tanto, este proyecto solamente se enfoca en el mejoramiento de éste, y no en el proceso completo.

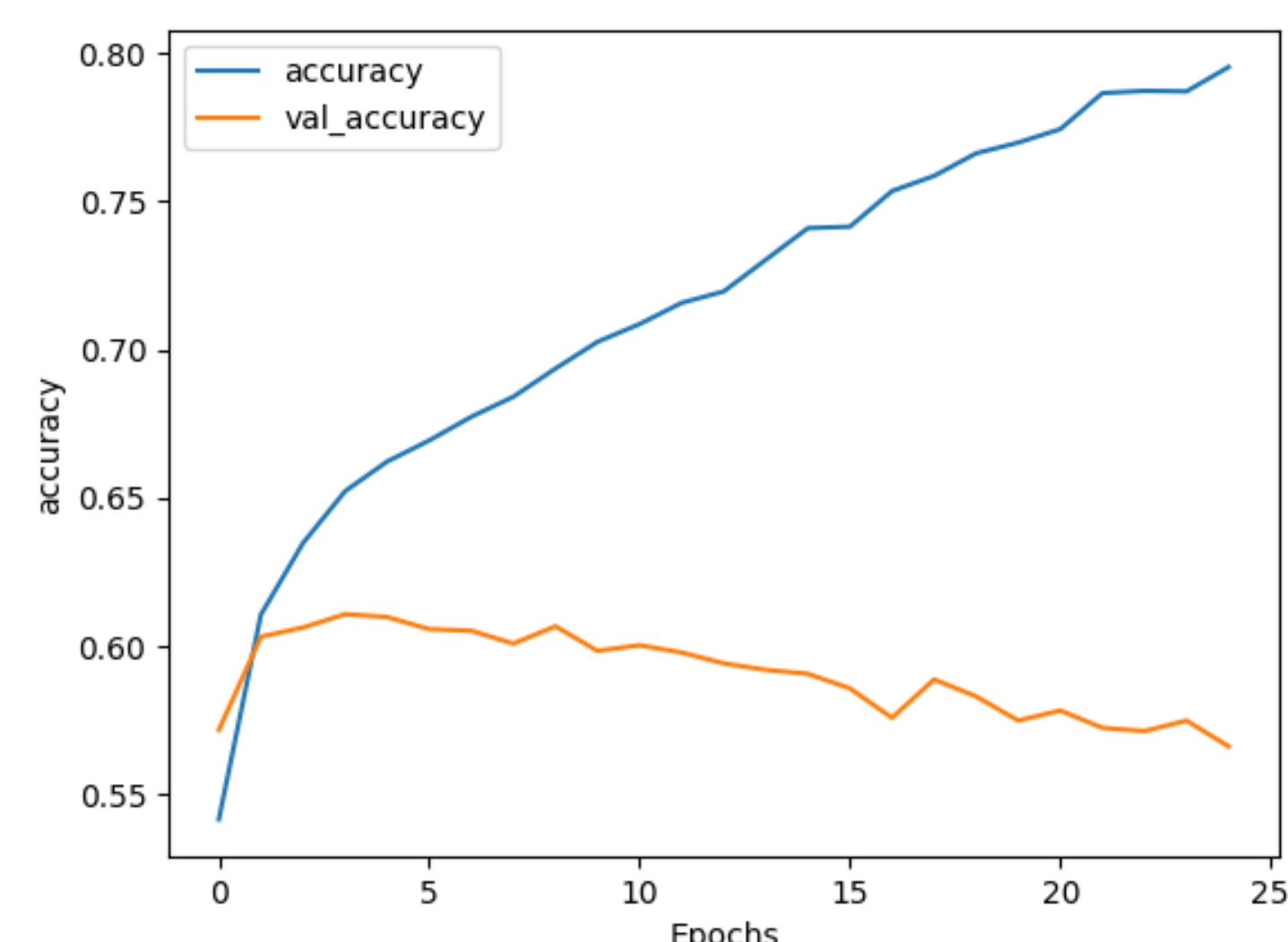


Figura 1: Gráfica representativa del porcentaje de acierto del algoritmo, antes de aplicársele alguna mejora

3. Estado de arte

En la actualidad existen muchos algoritmos complejos de análisis de sentimiento que funcionan sin necesidad de introducirle tantos datos, tales como GPT-3 (Generative Pre-trained Transformer 3). En donde gracias a su uso eficiente de transformadores y también gracias a que ha sido entrenado extensamente con datasets enormes (por ejemplo: Wikipedia Corpus, Common Crawl, WebText2, entre otros), no hay necesidad de encontrar datos tan específicos para entrenarlos [3].

Las opiniones son simples de entender para los humanos, pero, según Liu [4], para una computadora es necesario una noción con los siguientes elementos:

La gran diferencia entre el algoritmo propuesto en este proyecto y algo más elaborado como lo es GPT-3, es el hecho de que este último no funciona localmente, sino que en realidad es una API por la cual tienes que pagar para poder usar, además de que OpenAI tiene completo control y regulación de los usos de su algoritmo [5].

Lo que se intenta hacer al momento de utilizar TensorFlow es darle el control completo al usuario en caso de que quiera darle sesgo a los datos o que lo adapte a las necesidades que tenga, sin necesidad de que pase a ser una caja negra. Éste proceso es bastante complicado y probablemente no se alcanzará el grado de porcentaje de acierto con el que cuenta GPT-3, pero se espera que sea lo suficiente como para poder ser una alternativa open-source viable.

4. Solución propuesta

La implementación se hizo en Python 3.7 [6], además de otras herramientas relacionadas con redes neuronales como Tensorflow 2.0.0 [7].

Metodología Se comienza por una búsqueda de datasets, utilizándose la librería NLTK [8] para filtrar palabras que no aportan mucho significado al texto.

Se considera mejor un enfoque generalista debido a que las diferentes etiquetas de sentimiento a analizar no estaban distribuidas de manera uniforme; por lo que los datos se clasifican únicamente en "Bueno", "Neutral" y "Malo".

Se utiliza un algoritmo bidireccional STM con una activación de softmax de última capa. El léxico está limitado a 5000 elementos, y el largo máximo de cualquier frase después de ser filtrada es de 30 caracteres.

Ya después de ser actualizado, el dataset utilizado cuenta con alrededor de 60,000 tweets filtrados y clasificados.

Durante la fase de entrenamiento, se utilizaron 25 epochs con un 75 % del dataset en un orden arbitrario, usando el 25 % restante como validación.

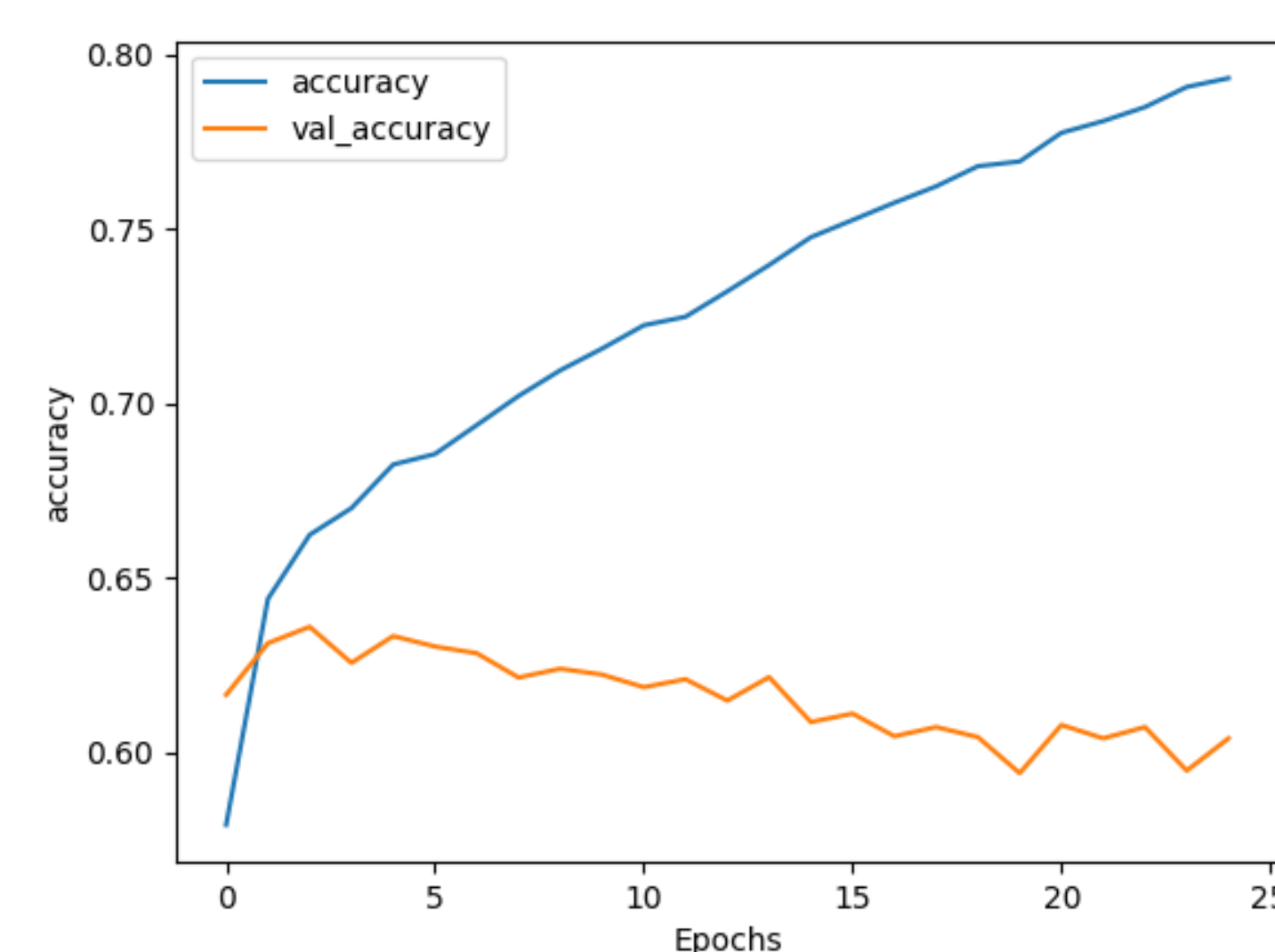


Figura 2: Gráfica representativa del porcentaje de acierto del algoritmo, después de aplicársele las mejoras propuestas

5. Experimentos

Para determinar que este enfoque iba a funcionar, lo que hicimos fue correr varias veces el algoritmo con diferentes parámetros dentro de él, identificando áreas de oportunidad en donde podrían ir mejoras, y finalmente comparando resultados.

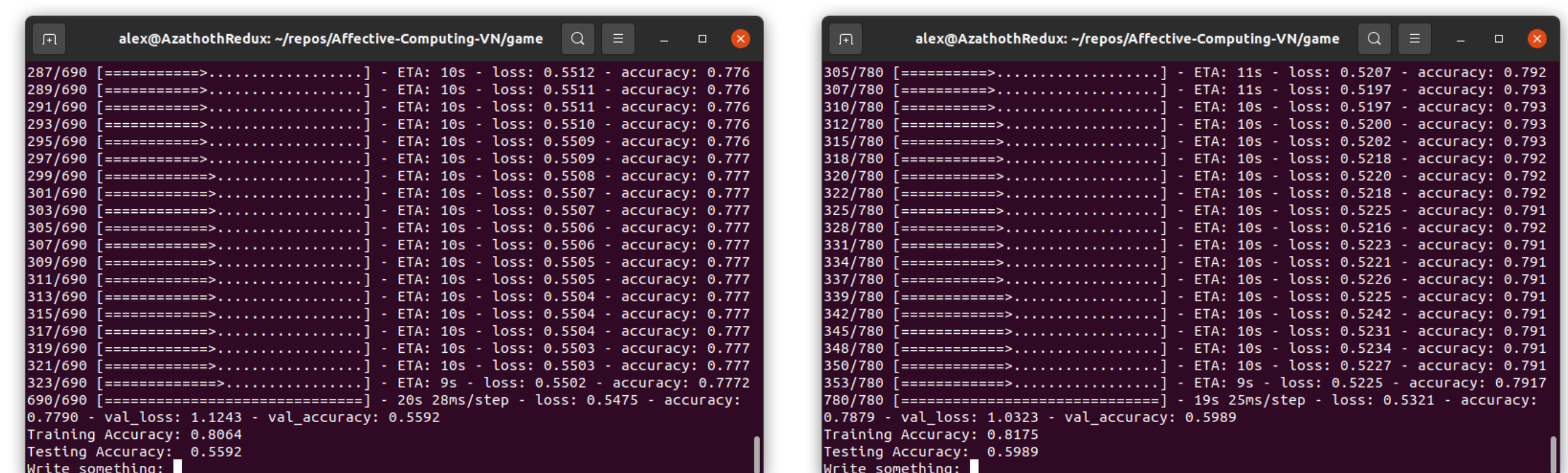


Figura 3: Parámetros antes y después de la mejora

6. Conclusiones

Se ha logrado un aumento en la precisión de reconocimiento de emociones, se espera realizar más modificaciones para mejorar el algoritmo y en un futuro implementarlo para su uso.

Agradecimientos Agradecemos a la dra. Elisa Schaeffer por su apoyo en la realización del póster, además de el PROVERICYT al dar la oportunidad de participar en el Verano de Investigación Científica y Tecnológica 2021 y proporcionar una beca. El póster se preparó con <https://www.overleaf.com/>.

Referencias

- [1] Tiejian Luo, Su Chen, Guandong Xu, and Jia Zhou. *Sentiment Analysis*, pages 53–68. 06 2013. ISBN 978-1-4614-7201-8. doi:10.1007/978-1-4614-7202-5_4.
- [2] Alexander Espronceda Gómez. Sentiment analysis through a chatbot. URL <https://github.com/Alex-Ego/Affective-Computing-VN>. A la fecha de redacción de este poster, éste paper sigue en proceso.
- [3] Amol Mavuduru. What is gpt-3 and why is it so powerful? *Towards Data Science*, 02 2021. URL <https://towardsdatascience.com/what-is-gpt-3-and-why-is-it-so-powerful-21ealba59811>.
- [4] Bing Liu. Sentiment analysis: A multi-faceted problem. *IEEE Intell Syst*, 25, 01 2010.
- [5] OpenAI. Openai api, 2020. URL <https://openai.com/blog/openai-api/>.
- [6] Python Software Foundation. python, 2021. URL <https://www.python.org/>.
- [7] Google Brain Team. tensorflow, 2019. URL <https://www.tensorflow.org/>.
- [8] Team NLTK. Natural language processor toolkit, 2021. URL <https://www.nltk.org/>.

