



FACULTAD DE INGENIERÍA
ESCUELA DE INGENIERÍA DE SISTEMAS

Easymoji: Sugerencias de emojis al escribir un mensaje.

Autores:

Richard Ascanio.

Manuel Ojeda.

Tutor Académico: Nicolás Araque Volk

Caracas, Febrero 2020

DERECHO DE AUTOR

Quienes suscriben, en condición de autores del trabajo titulado ***“EasyMoji: Sugerencias de emojis al escribir un mensaje”*** declaramos que: Cedemos a título gratuito, y en forma pura y simple, ilimitada e irrevocable a la Universidad Metropolitana, los derechos de autor de contenido patrimonial que nos corresponden sobre el presente trabajo. Conforme a lo anterior, esta cesión patrimonial sólo comprenderá el derecho para la Universidad de comunicar públicamente la obra, así como, la de salvaguardar nuestros intereses y derechos que nos corresponden como autores de la obra antes señalada. La Universidad en todo momento deberá indicar que se deban hacer al tutor o a cualquier tercero que haya colaborado o fuere hecho posible la realización de la presente obra.

Autor: Richard Ascanio

C.I.:V- 25.991.891

Autor: Manuel Ojeda

C.I.: V-27.291.164

En la ciudad de Caracas, a los 21 días del mes de febrero del año 2020

APROBACIÓN

Consideramos que el Trabajo Final titulado:

***EASYMOJI: SUGERENCIAS DE EMOJIS AL ESCRIBIR UN
MENSAJE***

elaborado por los ciudadanos:

*ASCANIO CASTRO, RICHARD EDUARDO
OJEDA ALBARRACIN, MANUEL ALEJANDRO*

para optar al título de:

INGENIERO DE SISTEMAS

reúne los requisitos exigidos por la Escuela de nombre de la escuela de la Universidad Metropolitana, y tiene méritos suficientes como para ser sometido a la presentación y evaluación exhaustiva por parte del jurado examinador que se designe.

En la ciudad de Caracas, a los 21 días del mes de febrero del año 2020.

Ing. Nicolás Araque

Tutor

ACTA DE VEREDICTO

Nosotros, los abajo firmantes, constituidos como jurado examinador y reunidos en Caracas, el día ____ de _____ de 2020, con el propósito de evaluar el Trabajo Final titulado

EASYMOJI: SUGERENCIAS DE EMOJIS AL ESCRIBIR UN MENSAJE

presentado por los ciudadanos

ASCANIO, RICHARD Y OJEDA, MANUEL

para optar al título de

INGENIERO DE SISTEMAS

emitimos el siguiente veredicto:

Reprobado ____ Aprobado ____ Notable ____ Sobresaliente ____

Observaciones: _____

Jurado Presidente

Tutor Académico

Jurado

AGRADECIMIENTOS

A mis padres, Richard Ascanio y Ofelia Castro y mi hermano Alexander Ascanio por su apoyo incondicional, por siempre creer que este momento iba a llegar. A mi novia Anyela Hernandez por ser pilar importante de este logro y por convertirme a diario en mejor persona. Gracias a ustedes soy quien soy y he podido lograr todo lo que he logrado y lograré.

Richard Ascanio.

Dedicado a mis padres Sixto Ojeda y Evelyn Albarracin quienes me han formado y guiado durante toda la vida con sus valores y principios y han sido mi mas grande apoyo durante todo este proceso .A mi hermana Valentina Ojeda la cual ha sido incondicional y ha estado presente en todo momento creyendo en mí y brindándome todo su apoyo. Ustedes son mi motivación, el pilar de mi vida y gracias a ustedes he podido lograr mis metas.

Manuel Ojeda

A Venezuela, porque creemos en ella y sabemos que si trabajamos duro por ella podemos llegar a recuperarla.

A nuestra Universidad Metropolitana y a todos su profesores, quienes a través de su esfuerzo y su buena ética de trabajo nos dieron las herramientas para nuestra formación profesional.

Al Prof. Nicolás Araque de la Universidad Metropolitana, por siempre como transmitir sus conocimientos y experiencias durante todo el proceso de elaboración de este proyecto, desde el primer día hasta hoy.

Richard Ascanio

Manuel Ojeda

TABLA DE CONTENIDO

DERECHO DE AUTOR	3
APROBACIÓN	4
ACTA DE VEREDICTO	6
AGRADECIMIENTOS	8
TABLA DE CONTENIDO	10
LISTA DE TABLAS	13
LISTA DE FIGURAS	14
RESUMEN	15
INTRODUCCIÓN	17
CAPÍTULO I	19
I.1 Planteamiento del problema	19
I.2 Objetivos	22
Objetivo General	22
Objetivos Específicos	22
I.3 Limitaciones	23
CAPÍTULO II	26
II.1 Aprendizaje Automático	26
II.2 Aprendizaje Profundo	27
II.3 Redes Neuronales Artificiales	27
II.4 LSTM	28
II.5 Entrenamiento	28
II.6 Embedding	29
II.7 Dropout	29
II.8 Función Softmax	29
II.9 Análisis de Sentimiento	30
II.10 Herramientas de desarrollo	31
II.10.1 Tensorflow	31

II.10.2 Keras	31
II.10.3 SpaCy	31
II.10.4 Django	32
II.10.5 Angular	32
CAPÍTULO III	33
III.1 Preprocesamiento de los datos.	33
III.2 Fases del sistema de predicción de emojis	34
III.2.1 Desarrollo del API de Twitter	34
III.2.2 Implementación API de Twitter	34
III.2.3 Limpieza de datos almacenados	35
III.2.4 Implementación del modelo	35
III.3 Estructura del modelo.	35
III.4 Entrenamiento del modelo	36
III.5 Análisis de sentimiento	37
III.6 Arquitectura del Software	37
III.7 Desarrollo de la aplicación web	37
III.8 Pruebas	38
CAPÍTULO IV	40
IV.1 Requerimientos del sistema	40
IV.2 Diseño de la arquitectura del modelo	40
IV.3 Implementación del Modelo RRN	40
IV.3.1 Análisis de Sentimientos	45
IV.4 Desarrollo de software	47
IV.5 Pruebas	48
CAPÍTULO V	51
REFERENCIAS BIBLIOGRÁFICAS	54
APÉNDICES	62
Apéndice A: Arquitectura del modelo	62
Apéndice B: Esquema del modelo	63
Apéndice C: Esquema de procesamiento de datos	63
Apéndice D: Pruebas de software	64
Apéndice E: Pruebas de software	64
Apéndice F: Pruebas de software	65

Apéndice G: Pruebas de software	65
Apéndice H: Pruebas de software	66
Apéndice I: Pruebas de software	66
Apéndice J: Pruebas de software	67
Apéndice K: Pruebas de oraciones 1	67
Apéndice L: Pruebas de oraciones 2	67
Apéndice M: Pruebas de tweets 3	68
Apéndice N: Código Jupyter Notebook de análisis de sentimiento	68

LISTA DE TABLAS

Tablas

Tabla 1. Tabla de Hiperparametros, 43

Tabla 2. Tabla de porcentaje de sentimiento de emoji, 47

Tabla 3. Tabla de clasificación de emojis, 48

Tabla 4. Tabla de predicción errónea 1, 51

Tabla 5. Tabla de predicción errónea 2, 52

Tabla 6. Tabla de predicción errónea 3, 52

LISTA DE FIGURAS

Figuras

Figura 1. Función Softmax, 31

Figura 2. Gráfica de función de precisión, 44

Figura 3. Gráfica de función de pérdida, 44

Figura 4. Matriz de confusión, 45

Figura 5. Gráfica de Sentimientos de Emojis, 48

Figura 6. Pantalla del software elaborado, 49

Figura 7. Pantalla del software elaborado, 50

RESUMEN

Easymoji: Sugerencias de emojis al escribir un mensaje

Autores:

Richard Ascanio

Manuel Ojeda

Tutor: Ing. Nicolás Araque

Caracas, Febrero 2020

El presente trabajo de grado tuvo como objetivo desarrollar una aplicación web que evidencie la técnica de inteligencia artificial de predicción de emojis. Esta propuesta se llevó a cabo a través de un modelo de aprendizaje profundo, específicamente utilizando Redes Neuronales Recurrentes y Procesamiento del Lenguaje Natural o NLP por sus siglas en inglés. En principio se desarrolló un algoritmo donde se extraen oraciones de un API de Twitter. Posteriormente, se realizó un proceso de tokenización, donde se le asignó a cada oración un número identificador que representa un emoji. Seguidamente se desarrolló el modelo EasyMoji donde se usó la librería spaCy para implementar NLP en el modelo y preparar el texto para computarlo con deep learning. Dentro de esta etapa se utilizaron estas herramientas para crear una matriz pre-entrenada de embeddings de 50 dimensiones con todo el vocabulario cargado. Este modelo se entrenó con una arquitectura de redes neuronales recurrentes denominada LSTM y utilizando softmax como última capa de la red. Luego, se procedió a diseñar la aplicación web donde se implementa este modelo y donde es más sencillo para un usuario interactuar con el mismo. Finalmente se realizaron pruebas al sistema, donde se evaluó la precisión del modelo EasyMoji. Con esto se concluyó que las redes neuronales recurrentes son una de las mejores alternativas a la hora de estudiar un texto debido a la capacidad que mostraron para extraer el sentimiento del mismo.

Palabras clave: Inteligencia Artificial, Deep Learning, Redes Neuronales Recurrentes, Emoji, NLP, Embedding, API, EasyMoji, tokenización, spaCy, sentimiento del texto, LSTM, aplicación web.

INTRODUCCIÓN

Según Zaremba (2015) las redes neuronales recurrentes (RNN) son un modelo neuronal secuencial que brinda un rendimiento de vanguardia en tareas importantes que incluyen estudio de lenguaje, reconocimiento de voz y traducción automática. Es por eso que se consideran una de las mejores herramientas para poder predecir o estudiar textos, esto lo logran gracias a sistemas especiales en el procesamiento de lenguaje como lo es NLP. El procesamiento del lenguaje natural o (NLP) por sus siglas en inglés es un área de estudio y aplicación que explora cómo las computadoras pueden ser utilizadas para entender y manipular texto o lenguaje natural para hacer cosas útiles (Chowdhury, 2013).

El presente trabajo de investigación permitió diseñar un sistema de predicción de emojis, utilizando inteligencia artificial, específicamente redes neuronales recurrentes con la intención de ofrecer una herramienta que permita anunciar un emoji adecuado a un texto ingresado por el usuario.

La realización de este trabajo se llevó a cabo en cinco etapas. La primera consistió en la investigación de los tópicos a tratar, así como las herramientas adecuadas para la realización del proyecto. En la segunda etapa se realizó el levantamiento de los requerimientos. Tomando como base dichos requerimientos, en la tercera etapa, se diseñó su arquitectura, para posteriormente pre procesar la data que se ingresó al modelo y desarrollar el software en la cuarta etapa. La quinta etapa consistió en realizar pruebas de precisión sobre el sistema.

El contenido de la presente investigación se encuentra distribuido en diversos capítulos. El **Capítulo I** consistió en desarrollar el planteamiento del problema, los objetivos y las limitaciones en el momento de ser desarrollado el trabajo. El **Capítulo II** denominado como el Marco de Referencia de la investigación se exponen conceptos fundamentales de machine learning, deep learning y redes neuronales recurrentes. En el **Capítulo III** se halla el Marco Metodológico donde se plantean todos los pasos y procedimientos que se siguieron para la realización de este proyecto. En el **Capítulo IV** se encuentran los Resultados y Análisis, donde se presentan los resultados obtenidos y su relación con cada objetivo, junto con su análisis. Finalmente, en el **Capítulo V** se encuentran las Conclusiones y Recomendaciones donde se presentan las conclusiones obtenidas tras realizar la investigación. Además, se plantean recomendaciones para investigaciones futuras.

CAPÍTULO I

Tema de investigación

I.1 Planteamiento del problema

Según el Diccionario Cambridge (2019) un emoji es “una imagen digital que es añadida a un mensaje en la comunicación electrónica con el objetivo de expresar una idea o sentimiento en particular”.

Estos emojis están disponibles solo en teléfonos inteligentes, pero según el portal del Diario El País de España el año 2017 cerró con más de 5.000 millones de usuarios únicos de telefonía móvil a nivel mundial, lo que supone un grado de penetración del 66%. Esta información fue publicada por este diario utilizando los datos del informe anual Mobile Economy de la GSMA, la organización que organiza el Mobile World Congress (MWC) que se celebra en Barcelona.

El mismo artículo señala que “El estudio del Mobile Economy prevé que se añadirán casi mil millones de usuarios de la telefonía móvil en 2025, alcanzando los 5.900 millones de suscriptores, lo que equivale al 71% de la población mundial prevista para ese momento”. Finalmente se hace referencia a que el crecimiento de los suscriptores durante este período estará impulsado por los países en desarrollo, particularmente Bangladesh, China, India, Indonesia y Pakistán, así como por los mercados de África Subsahariana y América Latina. Además, para 2025, se espera que 5.000 millones de suscriptores utilicen sus teléfonos móviles para acceder a Internet, frente a los 3.300 millones en 2017.

De esta gran cantidad de personas en el mundo que utilizan dispositivos móviles, una parte significativa utiliza en ellos mensajería instantánea para comunicarse, bien sea a través de mensajes escritos o multimedia. Las aplicaciones de mensajería instantánea más utilizadas son Whatsapp, Skype, Hangout, Telegram, Facebook Messenger, entre otros. El factor que ha permitido que se pueda ver la utilización de estas herramientas o medios de comunicación de forma muy pronunciada desde un dispositivo móvil es la transformación digital. La industria ha dado un vuelco más a la forma de comunicación entre dos o más personas, en parte debido por el exceso de información que recibe una persona diariamente. Por este motivo, la comunicación basada en imágenes, vídeos y emoticonos son los reyes actuales en el mundo digital, y explica porque los nativos digitales hacen uso de aplicaciones más visuales, como es el caso de Instagram o Snapchat. (Universia Argentina. 2017).

Ahora, dentro de estas redes sociales hay algo que año tras año va ganando importancia, los emojis. Según indica el diario El Siglo de Durango de México: “Los emojis se han convertido en formas perfectas de expresar sentimientos y palabras de una manera simplificada”. También este artículo indica que de acuerdo con una investigación publicada en la revista “Trends on Cognitive Sciences”, es necesario crear una nueva área de estudio llamada “Psicología de los emojis”, pues es importante entender a grandes rasgos la comunicación digital y la transformación que ha sufrido el lenguaje con el uso de emojis, ya que al sustituir los gestos humanos con éstos, se revelan diversos aspectos psicológicos de quienes los usan. “Las personas pueden expresar lo que están sintiendo específicamente, el uso de los emojis es tan fundamental como las expresiones que realizan las personas al hablar frente a frente, ya que ayudan a comprender el significado y a generar

empatía entre las personas que se están comunicando”, sugiere la publicación.

Por otro lado, de acuerdo con los datos de la revista *Social Neuroscience* (publicados en el portal milenio.com), “el cerebro lee a los emojis como información no verbal que traduce como emociones”. Además, Vyvyan Evans, profesor de la Universidad Bangor del Reino Unido y autor del libro “El código Emoji”, asegura que los emojis cambian la forma en que las personas interpretan los textos y hacen que la comunicación sea más efectiva.

Adicionalmente, en el trabajo de investigación *Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm* realizado por Felbo, Mislove, Søgaard, Rahwan y Lehmann en 2017 se ve como con la inteligencia artificial se pueden generar modelos en los que se relacionan los textos con los emojis y los sentimientos y emociones que estos transmiten “... millones de textos en las redes sociales con emojis se pueden usar para los modelos de pre entrenamiento, lo que les permite aprender representaciones de contenido emocional en los textos”.

Otro de los aspectos que se tomó en cuenta a lo largo de esta investigación, fue el trasfondo que conlleva utilizar emojis en conversaciones entre personas en aplicaciones de chats, “El uso de Emoji se ha convertido en una nueva forma de comunicación social, lo cual es importante porque puede ayudar a mejorar los sistemas de comunicación, como las aplicaciones de chat” (Elvis, 2018).

Tomando en cuenta todos estos aspectos dados anteriormente, se sabe la gran proporción de personas que utilizan un dispositivo móvil y quienes utilizan al menos una aplicación de mensajería instantánea como medio de comunicación y la diferencia que un emoji le puede dar a el sentido de un mensaje de texto, se planteó: ¿Por qué no basarse en la investigación de Felbo, Mislove, Søgaaard, Rahwan y Lehmann para diseñar una herramienta que le dé a los usuarios unas sugerencias de los emojis que deberían utilizar en un cierto mensaje escrito en español?

I.2 Objetivos

Objetivo General

Desarrollar una aplicación web que evidencie la técnica de inteligencia artificial de predicción de emojis.

Objetivos Específicos

- Diseñar la arquitectura de la aplicación web a desarrollar tomando en cuenta los requerimientos.
- Desarrollar un algoritmo de predicción de emojis utilizando Redes Neuronales Recurrentes.
- Estudiar el sentimiento de un texto ingresado con el uso de Words Embeddings (Palabras Embebidas), Procesamiento de Lenguaje Natural (NLP) y Machine Learning.
- Evaluar el funcionamiento del modelo a través de pruebas de performance con la data suministrada.

- Desarrollar la aplicación web que se acople al sistema de inteligencia artificial para ser usado.

I.3 Limitaciones

Este proyecto se realizó en un período de 6 meses, comprendidos entre julio del año 2019 y febrero del año 2020. El objetivo de la investigación fue elaborar a nivel práctico un sistema web donde se pueda comprobar la técnica de inteligencia artificial de estudio de un texto para la predicción de un emoji usando redes neuronales recurrentes.

A diferencia de los proyectos e investigaciones antes realizadas, este trabajo de grado propone la implementación de herramientas como Tensorflow, Keras, spaCy, entre otros. Para el estudio del texto y la relación y asociación con cada emoji.

La herramienta principal para el desarrollo de este trabajo de grado además de las señaladas anteriormente, sería la Inteligencia Artificial. Según Briega, un concepto importante para el estudio de textos utilizando la Inteligencia Artificial en Python es el de un Procesamiento del Lenguaje Natural (o NLP Natural Language Processing) lo que se define como “una disciplina que se encuentra en la intersección de varias ciencias, tales como las Ciencias de la Computación, la Inteligencia Artificial y la Psicología Cognitiva. Su idea central es darle a las máquinas la capacidad de leer y comprender los idiomas que hablamos los humanos”. Esta disciplina ayudará a descubrir el significado del texto a nivel fonético, morfológico, sintáctico, semántico, discursivo y pragmático y luego de estudiar el texto y obtener el

sentimiento del mismo ayudará a poder generar automáticamente el emoji adecuado.

Para el entrenamiento del modelo a desarrollar, se tomaron los inputs, en este caso oraciones extraídas de la red social Twitter mediante su API y con ellos se generó el output (el emoji) esperado de acuerdo al texto que el modelo estudia. Las Redes Neuronales Recurrentes requieren tiempos de procesamiento muy grande dependiendo del tipo de algoritmo de entrenamiento que se utilice. No obstante, a pesar de esto, la velocidad con la que convergen al resultado esperado, ignorando máximos y/o mínimos locales, sobrepasa con creces esta desventaja, en especial en redes con un número de neuronas reducido. En cuanto a sus porcentajes de error, las RNN son más efectivas al detectar, clasificar y aprender nuevos patrones. Para un mismo conjunto de datos de entrenamiento, entregarán mejores resultados que cualquiera de los otros tipos de redes neuronales (Oropeza C.). Es por esto que fueron las ideales para usarlas en este proyecto, para que pueda hallar con la mayor precisión posible el significado del mensaje escrito por el usuario y poder identificar cual emoji es el indicado.

En esta misma fase, para poder hallar el sentimiento u orientación del texto que escribe el usuario se tiene que utilizar una herramienta de Inteligencia Artificial llamada Análisis de texto y procesamiento de lenguaje natural o NLP, por sus siglas en inglés. El cual hace referencia a un método estadístico que usa el análisis de texto para comprender mejor la estructura, la intención y el sentimiento de las oraciones. (Estrategia y Negocios, 2018).

Para el desarrollo de esta herramienta se utilizó Python ya que gracias a su escalabilidad, robustez y fácil implementación de las bibliotecas de

Aprendizaje Profundo como Keras y Tensorflow puede hacer de este un proceso mucho más sencillo. (Soloaga, 2018).

Finalmente, se realizó una aplicación web como interfaz gráfica, con el fin de poder apreciar mejor y de forma más limpia para el usuario la funcionalidad del modelo EasyMoji, así como también poder observar los tres mejores resultados ponderados obtenidos por este. Evidenciando así la técnica de inteligencia artificial de predicción de emojis.

CAPÍTULO II

Marco Teórico

Para llevar a cabo la investigación, primero se deben conocer los temas a tratar. En este Trabajo Especial de Grado todos los planteamientos irán relacionados a la inteligencia artificial, y sus distintas herramientas para el procesamiento de texto. Según la Asociación para Maquinaria de Computación (2010) Twitter es una red social con servicio de *microblogging* que cuenta con más de 41 millones de usuarios y que sigue creciendo rápidamente. Los usuarios de Twitter tuitean sobre cualquier tema dentro de un rango de 140 caracteres y siguen a otros usuarios para recibir sus tweets. Uno de los factores que se ha vuelto más común en los tweets de los usuarios es la presencia de emojis. Según Kelly R. y Watts L. Los emojis son pictografías bidimensionales que fueron diseñados originalmente para transmitir emociones entre participantes en una conversación basada en texto.

II.1 Aprendizaje Automático

El Machine learning o aprendizaje automático en su traducción al español es definido como el estudio de métodos computacionales para mejorar el rendimiento por medio de la automatización y mecanización de la adquisición de conocimiento a partir de la experiencia. (Langley, & Simon. 1995). Según Guillón (2008) El proceso de aprendizaje de automático consiste en tener una gran base de datos que se coloca como entrada a un algoritmo que se ejecuta dentro en una máquina entrenada para obtener una salida.

El Objetivo del machine learning es proporcionar diferentes niveles de automatización aplicando procesos de ingeniería del conocimiento, reemplazando la mano de obra humana que consume mucho tiempo con técnicas automáticas que mejoran la precisión o eficiencia al trabajar con los datos de entrenamiento (Langley, & Simon, 1995).

II.2 Aprendizaje Profundo

Según Deng (2014) El Deep Learning o Aprendizaje Profundo es una clase de las diferentes técnicas de machine learning, por medio del cual muchas capas de procesamiento de información en arquitecturas jerárquicas para el aprendizaje no supervisado de características en búsqueda de clasificación y análisis de patrones, la esencia del deep learning es calcular las características jerárquicas o representaciones de los datos de observación en donde las características o factores de nivel superior se definen a partir de los de nivel inferior.

El aprendizaje profundo es una función de la inteligencia artificial que imita el funcionamiento del cerebro humano al procesar datos y crear patrones para su uso en la toma de decisiones, es considerado un subconjunto del aprendizaje automático que tiene redes neuronales capaces de aprender sin supervisión a partir de datos no estructurados o sin etiquetar. (Nielsen, 2015, pág. 35)

II.3 Redes Neuronales Artificiales

Las redes neuronales artificiales son las principales componentes de las arquitecturas de aprendizaje profundo. Según Mujtaba (2001) Las redes

neuronales artificiales están inspiradas en la arquitectura de sistemas nerviosos biológicos y consiste en un grupo de neuronas que trabajan en paralelo para facilitar y agilizar el tomar una decisión rápida, estas redes están compuestas por elementos computacionales que se caracterizan por una matriz de pesos o valores numéricos que se van ajustando mediante un proceso de aprendizaje.

II.4 LSTM

Las LSTM o Long-Short-Time-Memory son un tipo especial de redes neuronales recurrentes que posee una memoria interna y puertas multiplicativa (Hochreiter, & Schmidhuber, 1997). La característica principal de las redes recurrentes es que la información puede persistir introduciendo bucles al diagrama de la red, por lo que, básicamente, pueden “recordar” estados previos y utilizar esta información para decidir cuál será el siguiente estado que van a tomar. Esta característica las hace muy adecuadas para manejar series cronológicas.

II.5 Entrenamiento

El entrenamiento o aprendizaje es un proceso por medio del cual se ajustan los valores de los parámetros de la red, partiendo de un conjunto de pesos o valores aleatorios el objetivo del entrenamiento es buscar un conjunto de pesos que permitan a la red obtener el mejor resultado (Britos, 2005).

II.6 Embedding

El procesamiento de lenguaje natural es el que se encarga de todas las tareas al momento de trabajar con palabras, frases o documentos. El Word Embedding surge como parte del NLP. El Word Embedding se encarga de capturar características semánticas y sintácticas del cuerpo de palabras, frase o texto sin intervención humana o el procesamiento dependiente del idioma, las características del embedding son tareas independientes que las hacen ideales para el modelado de lenguaje. (Chen et al., 2013). Según Goldberg y Levy (2014) una de las técnicas que ha ganado más atracción últimamente que proporciona word embeddings es Word2Vec.

II.7 Dropout

Las redes neuronales profundas o deep neural networks se caracterizan por contener una gran cantidad de parámetros que las hacen propensas a caer en overfitting, el dropout es una solución que se consiguió para este problema. Según (Hinton et al., 2015) el dropout consiste en omitir aleatoriamente la mitad de los parámetros de funciones o detector de características en cada entrenamiento caso, esto evita adaptaciones complejas en las que un parámetro de la función o detector de características es solo útil en el contexto de varios otros detectores de características específicas.

II.8 Función Softmax

Según Rodríguez (2018) en problemas donde el número de clases es superior a dos y las clases son mutuamente exclusivas se utiliza la función

de activación softmax en la última capa para obtener la probabilidad de cada clase dada el patrón de entrada.

En la figura 1 vemos la fórmula de la función.

$$\sigma_i(z) := \frac{\exp(\lambda z_i)}{\sum_{j=1}^n \exp(\lambda z_j)}, 1 \leq i \leq n.$$

Figura 1. Función Softmax

Fuente: Gao & Pavel (2017)

II.9 Análisis de Sentimiento

Según Liu (2015) El análisis de sentimiento es el campo de estudio que analiza las opiniones, sentimientos, evaluaciones, valoraciones y actitudes de las personas hacia entidades como productos, servicios, organizaciones, individuos, problemas, eventos, temas y sus atributos.

El Análisis de sentimiento ha sido manejado como una tarea del Procesamiento del Lenguaje Natural en muchos niveles de granularidad. Comenzando por ser una asignación de clasificación a nivel de documento, oración y hasta de frase. (Agarwal, Xie, Vovsha, Rambow y Passonneau).

El actualmente el análisis de sentimiento es el centro de las investigaciones en las redes sociales. Por lo tanto la investigación en análisis de sentimiento no solo tiene importancia e impacto en procesamiento de

lenguaje natural sino que también puede tener un profundo impacto en las ciencias políticas, económicas, y sociales. (Liu, 2015)

II.10 Herramientas de desarrollo

En esta sección se discutirá acerca de las principales herramientas que se necesitaron para la elaboración del sistema de predicción de emojis.

II.10.1 Tensorflow

Según Abadi (2016), Tensorflow es un sistema de inteligencia artificial que opera a gran escala y en ambientes heterogéneos. Tensorflow usa gráficos de flujo de datos para representar el cálculo, el estado compartido y las operaciones que mutan ese estado.

II.10.2 Keras

Según Antona (2017) Keras es una librería de redes neuronales escrita en Python y capaz de ejecutarse tanto sobre Tensorflow como sobre Theano que es otra framework de inteligencia artificial. Las principales características de esta librería es que es fácil y rápido de utilizar, soporta tanto redes neuronales convolucionales como recurrentes, soporta esquemas de conectividad arbitrarios, entre otros.

II.10.3 SpaCy

SpaCy es una biblioteca de procesamiento de lenguaje natural (o NLP por sus siglas en inglés) diseñada específicamente con el objetivo de ser una

biblioteca útil para implementar sistemas listos para producción. Es particularmente rápido e intuitivo, por lo que es un competidor superior para las tareas de NLP (SpaCy).

II.10.4 Django

Django es un framework de Python de alto nivel que fomenta el desarrollo de aplicaciones limpias, rápidas y con diseño pragmático. Este framework está diseñado por desarrolladores expertos y se ocupa de gran parte de la molestia del desarrollo web, haciéndola tan potente que es capaz de almacenar e interactuar con un modelo de red neuronal sin problema alguno. (Django)

II.10.5 Angular

Angular es un framework desarrollado en TypeScript, mantenido por Google y que cuyo objetivo es desarrollar aplicaciones web de simple pantalla. (Angular)

CAPÍTULO III

Marco Metodológico

En este capítulo explicamos el diseño de la investigación, el cual comprende distintos modelos metodológicos utilizados para el desarrollo de este proyecto. Durante este capítulo también se exponen los procedimientos que se seguirán para llevar a cabo la investigación y el desarrollo del modelo, las tecnologías y la descripción de los lenguajes que permitirán llevar a cabo este Trabajo de Grado.

III.1 Preprocesamiento de los datos.

El preprocesamiento de datos es una etapa fundamental en el proceso de extracción del conocimiento, cuyo objetivo principal es obtener un conjunto de datos final que sea de calidad y útil para la fase de extracción de conocimiento. (García et al., 2015)

Se obtuvieron los datos a través de un API de twitter, por medio de esta se realizó la petición de los tweets a través de su filtro de búsqueda en el cual se busco solo por emoji y se extrajeron los tweets que tuvieron un solo un emoji de modo que si se seleccionan tweets con múltiples emojis será más difícil saber cual es el emoji que más representa esa oración o texto, posteriormente se procedió a revisar y limpiar de forma manual estos datos obtenidos. Este proceso de limpieza consistió en eliminar data duplicada, filtrar seleccionando los datos considerados relevantes para el proceso de aprendizaje y corregir errores ortográficos para luego iniciar el procesamiento de lenguaje natural. El set de datos está formado por 301

tweets por cada uno de los emojis de manera, cantidad suficiente para optimizar los resultados durante el proceso de aprendizaje del modelo.

III.2 Fases del sistema de predicción de emojis

El proceso para la obtención y manipulación de los datos se realizó de acuerdo con el esquema representado en el apéndice C:

III.2.1 Desarrollo del API de Twitter

En primer lugar se desarrolló un *script* en Python, con la finalidad de obtener a través del API de Twitter *tweets* que contengan emojis. Esto se hizo con el objetivo de crear un dataset lo suficientemente grande para que el modelo pueda arrojar el resultado posible.

III.2.2 Implementación API de Twitter

Una vez implementado este *script* de recopilación de datos, se procedió a utilizarlo para obtener tweets de varios tópicos.

Estos datos obtenidos en esta fase son pasados a través de un apartado de tokenización, el cual separa el emoji de la frase y los inserta en el dataset de manera separada. Luego convierte el emoji en un número o etiqueta que posee cada emoji para que pueda ser ingresado en la red neuronal.

Ya tokenizados todos los datos recopilados, se procede a almacenarlos en el dataset del modelo.

III.2.3 Limpieza de datos almacenados

Todos estos datos obtenidos pasan a través de un sistema de limpieza manual. Este proceso de limpieza consistió en primero eliminar data duplicada, luego ir leyendo *tweet* por *tweet* e ir filtrando y seleccionando los datos considerados como adecuados en la relación *tweet* y *emoji* y además relevantes para el proceso de entrenamiento y aprendizaje del modelo, después se corrigieron los errores ortográficos y para luego iniciar la implementación del modelo.

III.2.4 Implementación del modelo

Finalmente, luego de la limpieza de los datos, estos están listos para ser ingresados en el dataset del modelo EasyMoji para que el mismo se pueda entrenar de la mejor manera. El modelo utilizado es el definido en el apéndice A y la estructura del modelo EasyMoji se describe a continuación.

III.3 Estructura del modelo.

A partir de varios estudios, y consultando diversas bibliografías se logró determinar la arquitectura más adecuada para las necesidades de nuestro sistema de detección de emojis. La arquitectura elaborada fué la igual a la que implementaron Felbo, Mislove, Søgaaard, Rahwan y Lehmann en su modelo de predicción pero con pequeños cambios, cuyo esquema es se muestra en el apéndice B.

Se tiene una capa de entrada donde se encuentra la frase convertida a número gracias a la herramienta de spaCy, a continuación la capa de *Embeddings* y luego de esta dos LSTM bidireccionales con dimensión de 256 separadas por un dropout con probabilidad 0.5. Gracias a los autores mencionados anteriormente se notó que algo que mejoraba este sistema era la implementación de una capa *Attention* luego de estas dos LSTM bidireccionales ya que esta ayuda a calcular la contribución de cada parte del texto o cada palabra con el sentimiento del texto global (De la Peña, 2018). Luego de la capa *Attention* se tiene otro *Dropout* con la misma probabilidad del primero y al final una capa dense como capa de salida. Para la función de activación se utilizó la función Softmax.

Se utiliza Softmax como última capa de clasificación ya que se adapta con el resultado que se quiere dar en el modelo, un vector donde cada posición se refiere a la probabilidad de que el emoji sea escogido.

III.4 Entrenamiento del modelo

El modelo se entrena con *categorical_crossentropy* como función de pérdida y adam como función optimizadora.

Se entrena con un total de 70 *epochs* los cuales son suficientes para que modelo tenga un buen rendimiento. Luego de que el modelo se entrena en su totalidad, se utiliza un herramienta llamada *pickle* para comprimir el modelo.

III.5 Análisis de sentimiento

Se estudió el sentimiento de cada uno de los tweets que obtuvimos. Para realizar esta tarea se utilizó la librería *spanish-sentiment-analysis* creada por Elliot Hofman que contiene un modelo de inteligencia artificial encargado de analizar un texto escrito y decir su porcentaje de positivismo. Con estos datos se pudo saber que emojis utiliza la gente para escribir mensajes con sentimientos positivos o negativos.

III.6 Arquitectura del Software

La arquitectura de la aplicación web desarrollada contiene un servidor web y una interfaz gráfica. La interfaz gráfica es la que interactúa con el usuario, donde el usuario escribe un texto, este es enviado al servidor web y luego muestra los emojis predecidos con su respectivo porcentaje de precisión. El servidor web recibe el texto de la interfaz, lo pasa a través del modelo EasyMoji, guarda la predicción que hace el modelo y finalmente envía los mejores tres emojis calificados con su respectiva calificación y los envía como una respuesta a la petición que la interfaz le envió a través de una conexión de red.

III.7 Desarrollo de la aplicación web

La aplicación web que contiene a este modelo para que pueda ser probado de forma más cómoda está diseñado con Django, un *framework* basado en Python para aplicaciones web de servidor que es donde se almacena la instancia comprimida del modelo EasyMoji, de tal manera que

no se entrena cada vez que se ejecute el método de la petición sino simplemente se prueba.

Para la parte de la aplicación web del lado del usuario se utiliza Angular, un framework basado en TypeScript muy robusto pero a su vez con mucha facilidad a la hora de hacer peticiones a un *API REST* o servidor.

El sistema de la aplicación web funciona de la siguiente manera:

- En la *frontend* el usuario escribe una oración que es enviada al *backend* o *API REST* por medio de una petición *HTTP* de tipo *POST* y permanece esperando una respuesta.
- El *backend* recibe el *JSON* que contiene la oración enviada por el usuario, carga la versión comprimida del modelo y ejecuta la función de predecir de este con el parámetro de la oración del usuario.
- El *backend* obtiene los resultados de la predicción del modelo e inmediatamente son enviados como respuesta al *frontend*.
- El *frontend* recibe la respuesta por parte del servidor (un vector que contiene los emojis más acertados con sus respectivos porcentajes) y la despliega en la pantalla de forma ilustrativa para que el usuario pueda observar sus resultados.

III.8 Pruebas

Para realizar el proceso de desarrollo, se llevar a cabo pruebas de desempeño con la data suministrada en el sistema de predicción de emojis. En esta se determinaron aproximadamente la cantidad de emojis que se escogieron y la cantidad de datos que se recopilaron por cada uno de los

emojis y también poder observar los casos en los que falló el modelo validando también si el fallo es muy significativo o no.

CAPÍTULO IV

Resultados

IV.1 Requerimientos del sistema

La primera fase de este trabajo de investigación consistió en levantar los requerimientos para el correcto funcionamiento del sistema de predicción de emojis, se pudo definir el diseño correcto de la arquitectura para este modelo en específico, así como también las métricas y herramientas utilizadas.

IV.2 Diseño de la arquitectura del modelo

Seguido del levantamiento de requerimientos se llevaron a cabo todas las decisiones tomadas con respecto al diseño de la arquitectura del modelo. Se tomó como base la arquitectura del modelo que realizaron Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan y Sune Lehmann en su investigación dando como resultado la arquitectura mostrada en el apéndice B.

IV.3 Implementación del Modelo RRN

Luego de definida la arquitectura y todos sus componentes se procedió a la implementación del modelo mostrado en el apéndice A.

El modelo implementó con el set datos anteriormente mencionado en el procesamiento de datos en el Capítulo III, que está formado por seis mil seiscientos veintidós tweets con su respectivo y que consta de trescientos un

tweets por cada emoji obteniendo set de datos balanceado para la obtención de mejores resultados. El set de datos fue dividido en en set de entrenamiento y set de prueba, el de entrenamiento por la herramienta *train_test_split* de *sklearn* dando como resultado un set de entrenamiento y un set de prueba con con cuatro mil novecientos sesenta y seis y mil seiscientos cincuenta y seis elementos respectivamente.

Luego de varios intentos y pruebas de ensayo y error los mejores resultados se obtuvieron con los hiperparametros mostrados en la tabla 1

LSTM	256
Dropout	0.5
Función de Activación	Softmax
Función de Pérdida	Categorical crossentropy
Función Optimizadora	Adam
Epoch	70
Batch Size	32

Tabla 1. Tabla de Hiperparametros.

Fuente: Elaboración propia.

El tiempo de entrenamiento fue de alrededor de seis horas y media en un *jupyter notebook* en una computadora personal y el resultado obtenido fue en el set de entrenamiento un acierto de alrededor del 95% y un una pérdida muy cercana a 0 mientras que en el set de pruebas el acierto fue de 33.09% y la perdida muy cercana a 5. Se notó que los resultados son buenos ya que tomando como referencia el modelo elaborado por Felbo, Mislove, Søgaaard, Rahwan y Lehmann en su investigación teniendo 1.2 billones de datos

obtuvieron una precisión de 43.6%, una métrica que no está tan alejada a la del modelo EasyMoji teniendo muchos datos menos. Este es un factor significativo y que también subraya la dificultad de la tarea de la predicción de emojis.

En las figuras 2 y 3 se pueden observar las gráficas de precisión y pérdida que obtuvo el modelo.

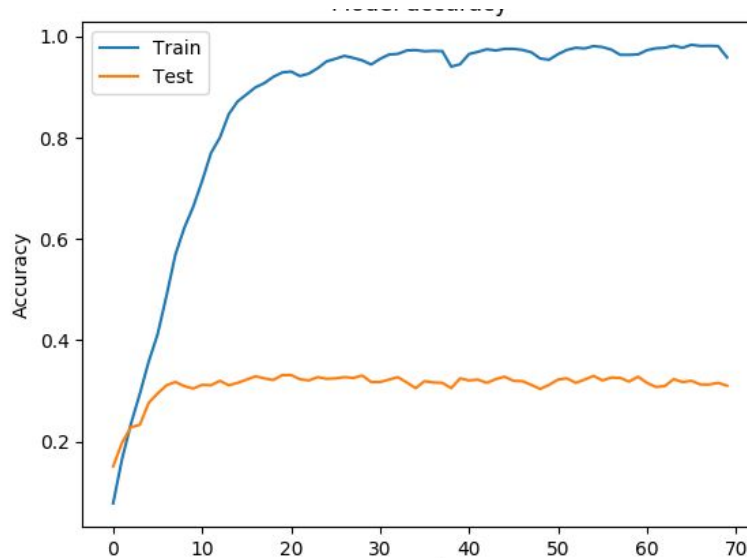


Figura 2. Grafica de funcion de precisión.

Fuente: Elaboración propia

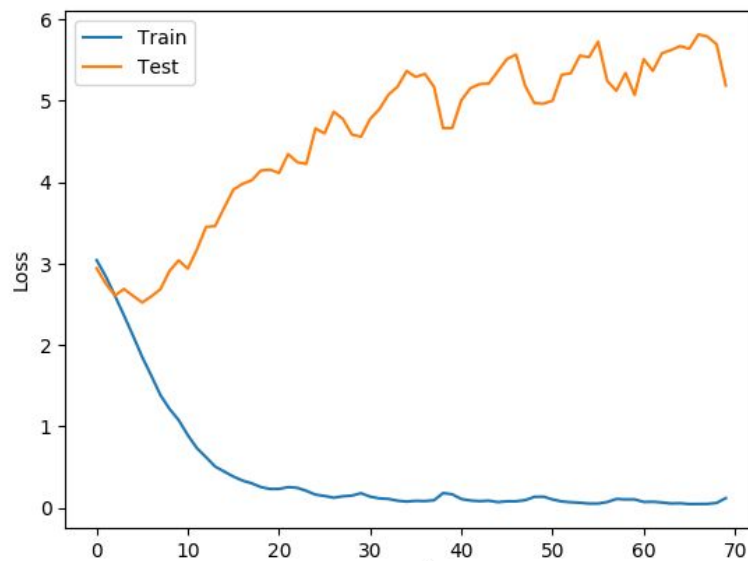


Figura 3. Gráfica de función de pérdida.

Fuente: Elaboración propia

En la figura 4 podemos observar la cantidad de veces en el que el modelo se equivoca y como se equivoca, es decir se puede ver por cada emoji cuantas veces fue acertado, cuantas veces no fue acertado y con cual emoji se equivocó.

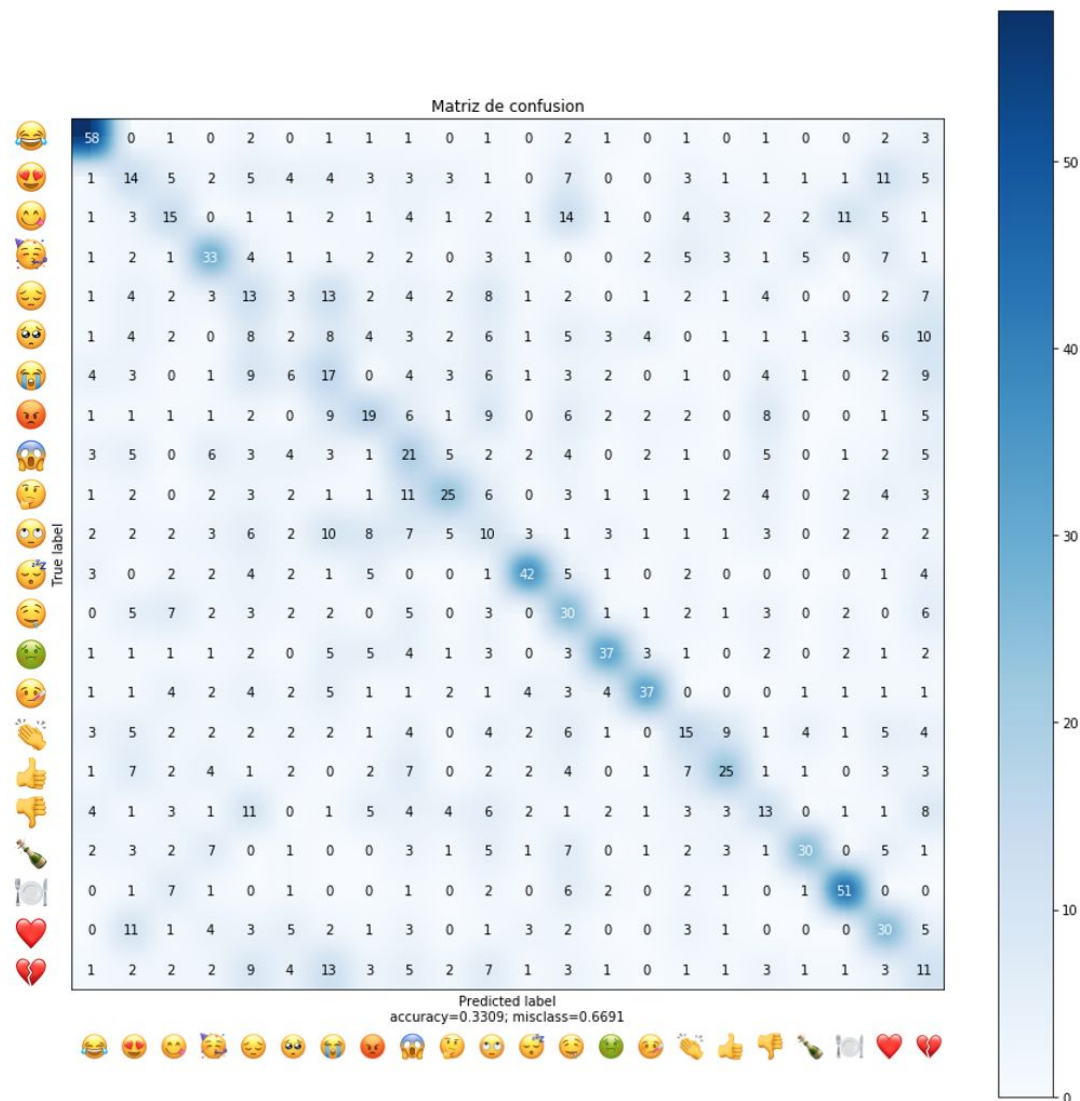


Figura 4. Matriz de confusión

Fuente: Elaboración propia

Los emojis 😂, 🍴, 😏, 🤔 y 😊 fueron notablemente los que más aciertos tuvieron, por el contrario el emoji 😐 fue el que menos aciertos tuvo.

Las equivocaciones más comunes fueron entre los pares de emojis (😊 y 😐), (😂 y 🍴), (❤️ y 😍), (👉 y 😊), (😍 y 🍴), (😏 y 🍴).

IV.3.1 Análisis de Sentimientos

Para analizar los sentimientos de cada tweet obtenido se utilizó una librería elaborada por Elliot Hofman publicada en PyPi (el manejador de paquetes de Python) llamada *spanish-sentiment-analysis 1.0.0* esta librería contiene un modelo de inteligencia artificial el cual estudia el nivel de positivismo que tiene un texto. Esta librería se implementó utilizando el código mostrado en el apéndice N.

Los resultados que arrojó el modelo fueron calculados y promediados por emoji obteniendo los valores mostrados en la tabla 2, en donde se muestra el porcentaje de sentimiento obtenido por cada emoji, siendo el valor de 0 el más negativo y 100 el más positivo.

Emoji numero:	7	😬	porcentaje	15.7
Emoji numero:	13	🤢	porcentaje	17.92
Emoji numero:	14	😏	porcentaje	21.34
Emoji numero:	17	👉	porcentaje	22.03
Emoji numero:	10	😞	porcentaje	26.23
Emoji numero:	21	💔	porcentaje	27.08
Emoji numero:	4	😓	porcentaje	27.96
Emoji numero:	6	😭	porcentaje	31.28
Emoji numero:	9	😟	porcentaje	35.48
Emoji numero:	11	😣	porcentaje	37.39
Emoji numero:	8	🤔	porcentaje	38.13
Emoji numero:	5	😓	porcentaje	40.86
Emoji numero:	0	😓	porcentaje	40.87
Emoji numero:	18	🔑	porcentaje	59.48
Emoji numero:	12	😟	porcentaje	59.6
Emoji numero:	20	❤️	porcentaje	60.34
Emoji numero:	16	👉	porcentaje	60.35
Emoji numero:	15	👉	porcentaje	60.98
Emoji numero:	3	🤔	porcentaje	61.1
Emoji numero:	2	😏	porcentaje	64.67
Emoji numero:	1	😬	porcentaje	65.16
Emoji numero:	19	👉	porcentaje	69.78

Tabla 2: Tabla de porcentaje de sentimiento de emoji

Fuente: Elaboración propia

Con los valores obtenidos mostrados en la tabla 2, se observa que el valor de sentimiento que se le calculó a cada emoji tiene sentido lo que el emoji representa para el ser humano. En la figura 5 se observan gráficamente los resultados de la tabla 2.

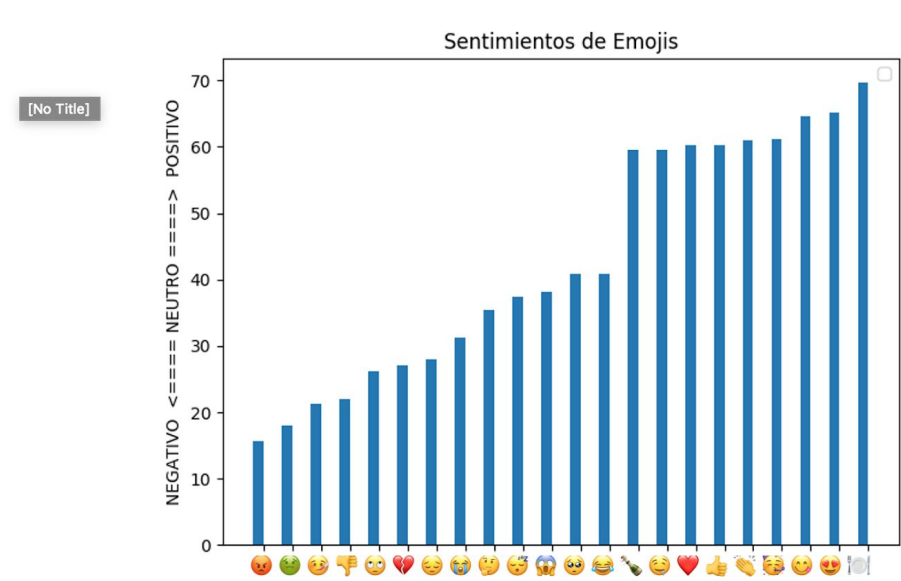


Figura 5. Gráfica de Sentimientos de Emojis.
Fuente: Elaboración Propia

Analizando la gráfica de la figura 5 podemos agrupar los emojis en 3 distintas categorías como se muestran en la tabla 3.

😡, 🤢, 😞, 🙄, 😓, ❤️, 😬, 😭	Emojis Negativos
😏, 🙌, 🤔, 😂	Emojis Neutrales
🥳, 🥰, 😄, 🙅, 🙄, 👍, 👏, 😇, 😁, 😍, 🥰	Emojis Positivos

Tabla 3: Tabla de clasificación de emojis
Fuente: Elaboración propia

IV.4 Desarrollo de software

La aplicación web del sistema de predicción de emojis es muy simple, le permite al usuario ingresar una frase que desea estudiar y esta envía una petición POST al servidor o API REST esperando una respuesta la cual son los emojis y sus porcentajes para finalmente desplegarlos como se explicó en el capítulo anterior. A continuación se puede ver la interfaz de la aplicación web desarrollada.

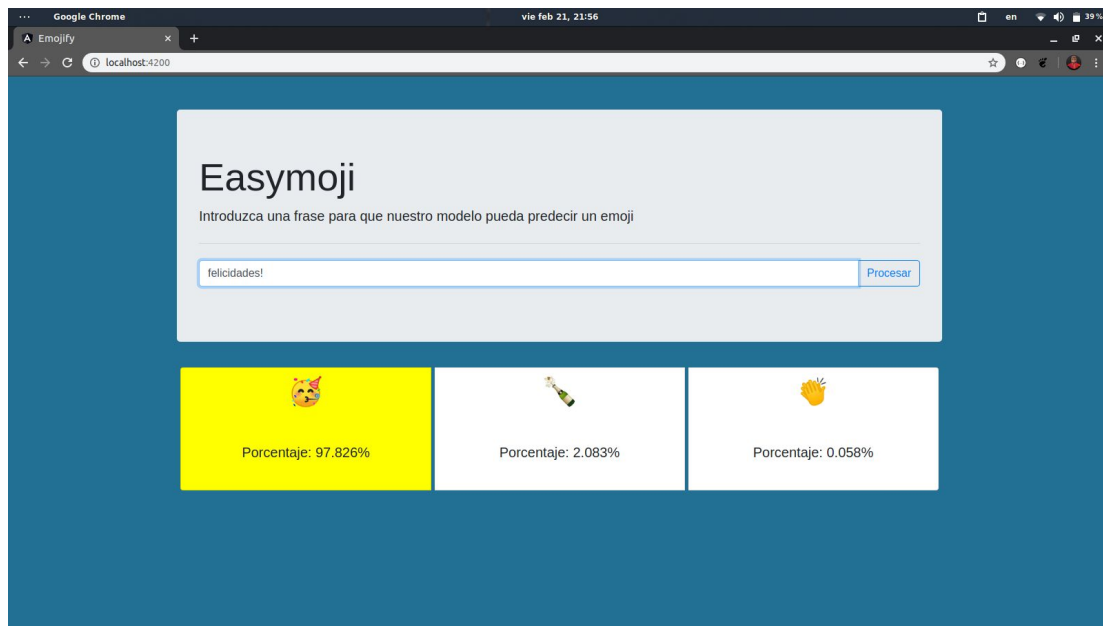


Figura 6. Pantalla del software desarrollado.

Fuente: Elaboración Propia

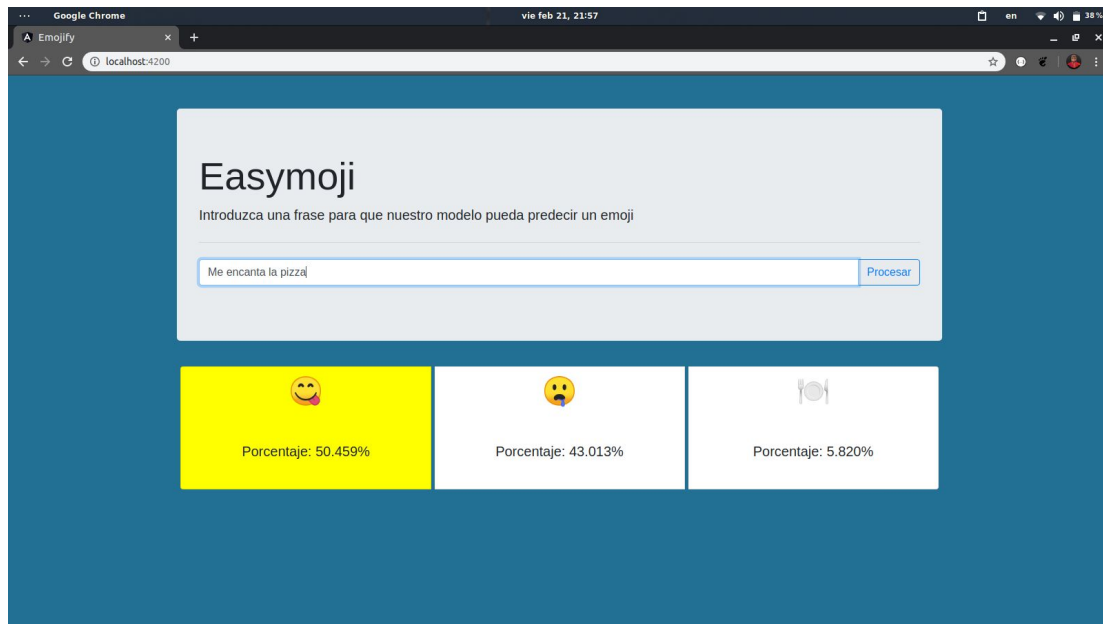


Figura 7. Pantalla del software elaborado.

Fuente: Elaboración Propia

En los apéndices D, E, F, G, H, I, J y K se pueden observar la interfaz gráfica de la aplicación web, donde se puede evidenciar el uso del modelo desde la misma y su interacción con el modelo de una forma más amigable a nivel de usuario.

IV.5 Pruebas

Se realizaron una cantidad de pruebas en el modelo, se ingresaron muchísimas oraciones con el fin de poder ver donde estaba fallando y poder solucionar el caso, algunas de las pruebas se pueden observar en los anexos K, L, M y en las tablas 4, 5 y 6.

Luego de obtener todos los datos se pudo observar que no se estaban indexando los signos de exclamación e interrogación, lo que hacía que el modelo no lo diera importancia a los mismos cuando no es así. Se pudo hallar una forma de que el sistema pudiera indexar estos signos ya que estos aportan emoción y significado a una frase.

También se realizaron diversas pruebas cambiando la arquitectura del modelo para saber cual de todas es la mejor opción. Al inicio se probó con dos capas LSTM de 64 dimensiones con el dropout conectado directamente a la capa dense. Seguidamente se notó que añadiendo una capa Flatten luego de la última capa LSTM mejoraba los resultados significativamente, pero por último se intentó eliminar nuevamente la capa Flatten y colocar la Attention lo que generó definitivamente un aumento en la precisión del modelo y en su desempeño.

Expected emoji: 🤔	prediction: Que sorpresa! 🤔
Expected emoji: 😊	prediction: No saben ni la mitad y hablan el doble 🤔
Expected emoji: 😊	prediction: que rápido creció 😊
Expected emoji: 😊	prediction: suena bien 🤔
Expected emoji: 😊	prediction: Cada vez me cansa mas esto 😊
Expected emoji: 😊	prediction: que fastidio 😊
Expected emoji: 😊	prediction: Que rica la tetona Longoria 😊
Expected emoji: 😊	prediction: ¿Quién a dicho que me quiero morir ? 🤔
Expected emoji: 😊	prediction: Dónde mero queda 🤔
Expected emoji: 🤔	prediction: Unas ganas de cortar semana mañana con unas birras 😊
Expected emoji: 😊	prediction: Todavía sin Luz 🤔
Expected emoji: 😊	prediction: Con lo que me encanta cocinar me doy los gustitos sola 😊
Expected emoji: 🤔	prediction: Por el fin de gira Salud 🤔
Expected emoji: 😊	prediction: Cuenta regresiva para comenzar en mi nuevo puesto 😊
Expected emoji: 😊	prediction: Que miedo un violador en las calles 🤔
Expected emoji: 😊	prediction: La resaca que tengo y el calor que no ayuda mucho que digamos 🤔
Expected emoji: 😊	prediction: Adivinad quien se va a apuntar a boxeo 🤔
Expected emoji: 😊	prediction: No quiero ir a la escuela 😊
Expected emoji: 🤔	prediction: Necesito una de esas agüitas relajantes cervezas creo que le llaman 😊
Expected emoji: 😊	prediction: Que disfrutes mucho ❤️
Expected emoji: 😊	prediction: Me duele todo me agarré una gripe seca del terror ❤️
Expected emoji: 😊	prediction: No les pasa que ven una foto y quieren regresar a ese momento de su vida 😊
Expected emoji: 😊	prediction: Oraré para que cambie tu perspectiva de triunfo en la vida 🤔
Expected emoji: 😊	prediction: Igual nos enteramos de las babosadas que dice 😊
Expected emoji: 🤔	prediction: En lo que piensa es en unas ricas y deliciosas Big Mac 🤔
Expected emoji: 😊	prediction: estoy muy gorda 😊
Expected emoji: ❤️	prediction: Siento que este capítulo es más terrible que la descripción de la guerra de los anteriores 😊
Expected emoji: 🤔	prediction: Buen partido de Cornejo 🤔

Tabla 4. Tabla de predicción errónea 1

Fuente: Elaboración propia

Expected emoji: 🤢	prediction: Estoy enfermo y voy a vomitar 🤢
Expected emoji: 😍	prediction: Maravillosa foto 😍
Expected emoji: ❤️	prediction: el lugar de mis sueños tiene a la gente más asquerosa y descerebrada del mundo 🤢
Expected emoji: 🤔	prediction: Tiene potencial 🤔
Expected emoji: 🙌	prediction: Ya estoy hasta la madre de fingir que todos cantan bien Salud 🙌
Expected emoji: 🤔	prediction: Te quedo muy bien 🤔
Expected emoji: 🤔	prediction: Se ha pasado del precio 🤔
Expected emoji: 🤔	prediction: Feo feísimo Le lanzan un mechero durante la celebración del gol y cantan desde la grada cosas sancionables 🤔
Expected emoji: ❤️	prediction: No quiero que se termine enero 🤔
Expected emoji: 🤔	prediction: Yo la como con tenedor y cuchillo 🤔
Expected emoji: 🤔	prediction: No hice la sesión por baja de presión 🤔
Expected emoji: 🤔	prediction: De tu tierra amiga maggieligg delicioso 🤔
Expected emoji: 🤔	prediction: Todo mi odio para la gente que abandona y no cuida los animales 🤔
Expected emoji: 🤔	prediction: me caso 🤔
Expected emoji: 🤔	prediction: Ojalá salgan campeones 🤔
Expected emoji: ❤️	prediction: No quiero ver mas imágenes de Australia 🤔
Expected emoji: 🤔	prediction: Me encanta el arroz con queso y atún 🤔
Expected emoji: 🤔	prediction: Menos mal los lunes no tengo clase porque que pereza tan asquerosa 🤔
Expected emoji: 🤔	prediction: Si me encantaría ir 🤔
Expected emoji: ❤️	prediction: YA ME RENDÍ 🤔
Expected emoji: 🤔	prediction: Los quiero todos 🤔
Expected emoji: 🤔	prediction: Un camino hermoso 🤔
Expected emoji: 🤔	prediction: Típico de imbécil Juzgar el físico de los demás sin foto de perfil propia 🤔
Expected emoji: 🤔	prediction: Que pinche suerte de este vato 🤔
Expected emoji: 🤔	prediction: Felicitaciones por tu presente 🤔
Expected emoji: 🤔	prediction: Yo siempre abrazo cuando duermo si no les gusta suban a otro colectivo 🤔
Expected emoji: ❤️	prediction: Os deseo cosas muy bonitas a los dos siendo uno 🤔
Expected emoji: 🤔	prediction: ME VA A DAR UN INFARTO 🤔
Expected emoji: 🤔	prediction: Odio a la gente que le hablas y la cuelgan 🤔
Expected emoji: ❤️	prediction: literal es lo más lindo que existe 🤔
Expected emoji: ❤️	prediction: si te dije de vernos y me dijiste que no 🤔

Tabla 5. Tabla de predicción errónea 2

Fuente: Elaboración propia

Expected emoji: 🤔	prediction: Últimamente en todos mis planes algo sale mal 🤔
Expected emoji: ❤️	prediction: Amandote tanto 🤔
Expected emoji: 🤔	prediction: Filete y Serrano BAR Restaurant BUEN GUSTO 🤔
Expected emoji: 🤔	prediction: Vas sintiendo que los años pasan cuando te escogen para ser madrina de bautizo 🤔
Expected emoji: ❤️	prediction: No hay mejor placer que comer chocokrispis 🤔
Expected emoji: 🤔	prediction: O se le fue por accidente 🤔
Expected emoji: 🤔	prediction: Un Buen Vino Dura Un Instante y Te Deja En La Boca Un Sabor Que Deleita Tus Sentidos 🤔
Expected emoji: 🤔	prediction: Que día tan malo en serio 🤔
Expected emoji: 🤔	prediction: casi me pongo a llorar en el trabajo Puta vida 🤔
Expected emoji: ❤️	prediction: OS JURO QUE SE ME HA ENCOGIDO EL CORAZÓN 🤔
Expected emoji: 🤔	prediction: Ese nivel no debería existir 🤔
Expected emoji: 🤔	prediction: Mujeres que hacen esto me encantan 🤔
Expected emoji: 🤔	prediction: eres lo mejor te quiero ❤️
Expected emoji: 🤔	prediction: Que encanto 🤔
Expected emoji: 🤔	prediction: Ay se me antojaron 🤔
Expected emoji: 🤔	prediction: Cocidito rico para este miércoles 🤔
Expected emoji: 🤔	prediction: Todo salió mal 🤔
Expected emoji: 🤔	prediction: Alexandra no así no 🤔
Expected emoji: ❤️	prediction: La peor noticia en años 🤔
Expected emoji: 🤔	prediction: Burritos de pierna bañado de frijol 🤔
Expected emoji: 🤔	prediction: esto es un infierno solo necesito ver a kagura con más de como mínimo es lo único que suplico 🤔
Expected emoji: 🤔	prediction: No podés estar más buena 🤔
Expected emoji: ❤️	prediction: Belleza e inteligencia 🤔
Expected emoji: 🤔	prediction: Si si lo entendimos solo el resaltar y dar este énfasis y protagonismo es muy resalta ble 🤔
Expected emoji: 🤔	prediction: colddbgb La estúpida que se pone a llorar porque le pusieron tetas 🤔
Expected emoji: 🤔	prediction: Mi confianza absoluta en YolandaDiaz Le sobra capacidad fuerza e inteligencia Sí se p uede 🤔
Expected emoji: 🤔	prediction: Muy buena definición 🤔
Expected emoji: 🤔	prediction: qué ganas de no existir esta semana 🤔
Expected emoji: 🤔	prediction: Estoy enfermo y voy a vomitar 🤔

Tabla 6. Tabla de predicción errónea 3

Fuente: Elaboración propia

CAPÍTULO V

Conclusiones y Recomendaciones

Este trabajo se realizó con el objetivo de evidenciar la técnica de inteligencia artificial de predicción de emojis. Este proyecto ha podido cumplir sus objetivos, destacando sobre todo un sistema capaz de predecir un emoji al instante de que un texto sea introducido en el sistema con muy buenos resultados, se puede concluir que:

Se lograron plasmar los requerimientos de la problemática en una aplicación web, esta permitió mostrar las funcionalidades necesarias del sistema de aprendizaje profundo utilizado. Además sirvió para demostrar el uso que se le puede dar este sistema en el día a día, y lo que puede llegar a significar para el usuario común.

Se realizó con éxito un algoritmo compuesto por redes neuronales recurrentes capaz de predecir emojis, con los resultados obtenidos en porcentajes de acierto y perdida en los sets de entrenamiento y de prueba observamos que a pesar de que durante el entrenamiento el porcentaje de acierto fue alto, en la práctica el modelo no es tan acertado y tiende a confundirse entre emojis que representan o tienen significados similares. Por lo tanto, el éxito de este modelo no debe medirse solo por el porcentaje de acierto sino también se debe tomar en cuenta la cantidad de emojis en donde se emplea y qué cantidad de estos representan o tienen significados similares.

La ventaja de utilizar las redes neuronales recurrentes es que la arquitectura se adapta mejor al conjunto de datos de las oraciones debido a

la reutilización de los pesos, capacidad de encontrar patrones en textos y gran comprensión. La capa *Attention* y su capacidad de capturar el contexto de cada palabra ayuda en gran cantidad al sistema para que aprenda de mejor manera y pueda obtener una mejor precisión.

Se realizó con éxito el estudio de análisis de sentimiento en el cual los resultados obtenidos hacen sentido con el sentimiento que pueda representar la figura de cada emoji, tomando en cuenta que solo se clasificaron de cero a uno siendo cero lo más negativo y uno lo más positivo.

El modelo arroja resultados coherentes en el cual la mayor cantidad de equivocaciones ocurrieron entre emojis que representan o tienen un sentimiento o significado muy similar, por otro lado tuvo pocas equivocaciones entre emojis que no son similares. Esto nos permite concluir que los resultados y precisión en este modelo de predicción de emojis va a variar dependiendo del grupo de emojis con el cual se trabaje, mientras el significado y su uso sea más parecido el modelo tendrá a equivocarse más y bajar la precisión, mientras más disperso y diferente en cuanto a significado de emojis sea el grupo con el que se trabaje se podrá obtener una mejor precisión.

Dado por finalizado el proyecto, se presentan ahora las siguientes recomendaciones para trabajos futuros:

Utilizar otro modelo pre entrenado y llevar a cabo el *transfer learning* para obtener mejores resultados.

Realizar un *hosting* a la aplicación en un servidor web, para poder ingresar desde cualquier dispositivo con conexión a internet

Utilizar un dataset más amplio, ya sea buscar uno que ya esté hecho, o realizar uno no solo con la API de twitter sino también con la de otras redes sociales, reseñas de películas, compras, etc.

Implementar otro modelo más preciso a la hora de estudiar el sentimiento del texto, y que tenga más etiquetas, que pueda diferenciar sobre más estados de ánimo además de positivo y negativo.

Incorporar este sistema a redes sociales y sistemas de comunicación instantánea como chats pero con una velocidad de respuesta muchísimo mayor.

REFERENCIAS BIBLIOGRÁFICAS

Abadi M., Barham, P. y Chen J. Tensorflow: A system for Large-Scale Machine Learning. Recuperado de: <https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf>

Antona C. Herramientas Modernas en Redes Neuronales: La Librería Keras. Recuperado de: https://repositorio.uam.es/bitstream/handle/10486/677854/antona_cortes_carlos_tfg.pdf?sequence=1

Angular. Angular IO. Recuperado de: <https://angular.io/>

Association for Computing Machinery (2010). What is Twitter, a social network or a news social?. Recuperado de: <https://dl.acm.org/doi/abs/10.1145/1772690.1772751>

Ayvar Carmen (2019), *7 beneficios del uso de emoticones para expresar tus emociones en Internet. Webespacio*. Recuperado de: <https://www.webespacio.com/7-beneficios-del-uso-de-emoticones-para-expresar-tus-emociones-en-internet/>

Bengio, Y. (2000). Gradient-based optimization of hyperparameters. *Neural computation*, 12(8), 1889-1900. Recuperado de: http://www.iro.umontreal.ca/~lisa/bib/pub_subject/language/pointeurs/n.c.pdf

- Bjarke Felbo, Alan Mislove, Anders Søgaard, Iyad Rahwan y Sune Lehmann (2017), *Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm*. Recuperado de: <https://arxiv.org/pdf/1708.00524.pdf>
- Britos, M. I. P. (2005). Entrenamiento de redes neuronales basado en algoritmos evolutivos. Recuperado de: <http://laboratorios.fi.uba.ar/lsi/bertona-tesisingenieriainformatica.pdf>
- Calderón Lucia (2015). *Los pros y contra de usar 'emoticones' en nuestros mensajes*. Clases de Periodismo. Recuperado de: <http://www.clasesdeperiodismo.com/2015/08/06/los-pros-y-contra-de-usar-emoticones-en-nuestros-mensajes/>
- Cambridge Dictionary (2019). Recuperado de: <https://dictionary.cambridge.org/es/diccionario/ingles-espanol/emoji>
- Chen, Y., Perozzi, B., Al-Rfou, R., & Skiena, S. (2013). The expressive power of word embeddings. *arXiv preprint arXiv:1301.3226*. Recuperado de: <https://arxiv.org/pdf/1301.3226.pdf>
- Chowdhury, G. (2003). Natural Language Processing. Annual Review of Information Science and Technology. Recuperado de: <https://strathprints.strath.ac.uk/2611/1/strathprints002611.pdf>
- De la Peña G., Pons R., Muñoz C., Rosso P. Hate Speech Detection using Attention-based LSTM. Recuperado de: <http://ceur-ws.org/Vol-2263/paper040.pdf>

Deng, L. (2014). A tutorial survey of architectures, algorithms, and applications for deep learning. *APSIPA Transactions on Signal and Information Processing*, 3. Recuperado de: <https://www.cambridge.org/core/services/aop-cambridge-core/content/view/023B6ADF962FA37F8EC684B209E3DFAE/S2048770313000097a.pdf/a-tutorial-survey-of-architectures-algorithms-and-applications-for-deep-learning.pdf>

Django. The web framework for perfectionists with deadlines. Recuperado de: <https://www.djangoproject.com/>

Dunne, R. A., & Campbell, N. A. (1997, June). On the pairing of the softmax activation and cross-entropy penalty functions and the derivation of the softmax activation function. In *Proc. 8th Aust. Conf. on the Neural Networks, Melbourne* (Vol. 181, p. 185). Recuperado de: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.49.6403&rep=rep1&type=pdf>

El País (2018) (España). *El número de líneas móviles supera por primera vez a la población mundial*. Recuperado de: https://elpais.com/tecnologia/2018/02/27/actualidad/1519725291_071783.html

Elizondo López (2002). *Metodología de la investigación contable*. Recuperado de: https://books.google.es/books?id=BLO9spGHxrwC&lpg=PA3&ots=0r_A63x7hf&lr&hl=es&pg=PP1#v=onepage&q&f=false

- Elvis (2018). *Deep Learning and Time Embeddings To Predict Emojis*. Medium. Recuperado de: <https://medium.com/dair-ai/deep-learning-and-time-to-predict-emojis-4a6256c16475>
- Estrategia y Negocios (2018). *5 herramientas de inteligencia artificial claves para el lenguaje y marketing digital*. Recuperado de: <https://www.estrategiaynegocios.net/empresasmanagement/1163807-330/5-herramientas-de-inteligencia-artificial-claves-para-lenguaje-y-marketing-digital>
- Fragua Cristina (2016). *¿Qué ventajas tiene el uso de emojis en la comunicación de tu negocio?*, Observatorio de RRHH. Recuperado de: <https://www.observatoriorh.com/comunicacion-y-branding/que-ventajas-tiene-el-uso-de-emojis-en-la-comunicacion-de-tu-negocio.html>
- Gao, B., & Pavel, L. (2017). On the properties of the softmax function with application in game theory and reinforcement learning. *arXiv preprint arXiv:1704.00805*. Recuperado de: <https://arxiv.org/pdf/1704.00805.pdf>
- García, S. Luengo, J. y Herrera, F. (2015). *Data Preprocessing in Data Mining*. Berlin: Springer
- Goldberg, Y. y Levy, O. (2014). Word2Vec Explained: Deriving Mikolov et al.'s Negative-Sampling Word-Embedding Method. Recuperado de: <https://arxiv.org/pdf/1402.3722.pdf>
- Guyon, I. (2008). *Introduction to machine learning*. Slides and Video Lectures

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. arXiv preprint arXiv:1207.0580. Recuperado de: <https://arxiv.org/pdf/1207.0580.pdf>

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735-1780. Recuperado de: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.676.4320&rep=rep1&type=pdf>

Kelly R., Watts L. (2015). Characterising the Inventive Appropriation of Emoji as Relationally Meaningful in Mediated Close Personal Relationships. Recuperado de: https://purehost.bath.ac.uk/ws/files/130966701/emoji_relational_value.pdf

Langley, P., & Simon, H. A. (1995). Applications of machine learning and rule induction. *Communications of the ACM*, 38(11). Recuperado de: <https://apps.dtic.mil/dtic/tr/fulltext/u2/a292607.pdf>

Liu, B. (2012). Sentiment analysis and opinion mining. *Synthesis lectures on human language technologies*, 5(1), 1-167. Recuperado de: <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.244.9480&rep=rep1&type=pdf>

Lopez Briega Raul E (2017). *Procesamiento del Lenguaje Natural con Python*. Recuperado de:

<https://relopezbriega.github.io/blog/2017/09/23/procesamiento-del-lenguaje-natural-con-python/>

Matich, D. J. (2001). *Redes Neuronales: Conceptos básicos y aplicaciones*. Universidad Tecnológica Nacional, México. Recuperado de: <ftp://decsai.ugr.es/pub/usuarios/castro/Material-Redes-Neuronales/Libros/matich-redesneuronales.pdf>

Mujtaba, I. M. (2001). *Application of neural networks and other learning technologies in process engineering*. World Scientific. Recuperado de : <https://gpreview.kingborn.net/273000/e0524fd1270847a2842f1a77ba8f00de.pdf>

Nielsen, M. A. (2015). *Neural networks and deep learning* (Vol. 2018) San Francisco, CA, USA: Determination press

Oropeza C. *Redes Neuronales Recurrentes*. Recuperado de: http://catarina.udlap.mx/u_dl_a/tales/documentos/lem/oropeza_c_ca/capitulo3.pdf

Peralta Daniel (2016). *Un Estudio sobre el Preprocesamiento para Redes Neuronales Profundas y Aplicación sobre Reconocimiento de Dígitos Manuscritos*. Recuperado de: https://www.researchgate.net/publication/308901913_Un_Estudio_sobre_el_Preprocesamiento_para_Red_Neuronales_Profundas_y_Aplicacion_sobre_Reconocimiento_de_Digitos_Manuscritos

Salas, R. (2004). *Redes neuronales artificiales*. Universidad de Valparaíso. Departamento de Computación, 1. Recuperado de:

https://s3.amazonaws.com/academia.edu.documents/50358783/Redes_Neuronales_Artificiales.pdf?response-content-disposition=inline%3B%20filename%3DRedes_Neuronales_Artificiales.pdf&X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=AKIAIWOWYYGZ2Y53UL3A%2F20200220%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20200220T161213Z&X-Amz-Expires=3600&X-Amz-SignedHeaders=host&X-Amz-Signature=9730ce125ca19da17e4da22a77cfcab73fdcbc71614b8f669e409ec1a3939acf

Rodríguez, R. (2018) Valoración de películas basada en Procesamiento del Lenguaje Natural y Deep Learning. Recuperada de: http://oa.upm.es/53401/1/TFM_RUBEN_RODRIGUEZ_FERNANDEZ.pdf

Soloaga Ana (2018). *Principales usos de Python*. Akademus. Recuperado de: <https://www.akademus.es/blog/programacion/principales-usos-python/>

SpaCy. Industrial-Strength Natural Language Processing in Python. Recuperado de: <https://spacy.io/>

Telesur (2018). *¿Cuántos venezolanos viven en el exterior?*. Recuperado de: <https://www.telesurtv.net/bloggers/Cuantos-venezolanos-viven-en-el-exterior-20180227-0001.html>

Universia (2017) (Argentina). *Mensajería instantánea, el medio de comunicación más utilizado*. Recuperado de: <http://noticias.universia.com.ar/ciencia-tecnologia/noticia/2017/12/12/1156840/mensajeria-instantanea-medio-comunicacion-utilizado.html>

Zaremba, W., Sutskever, I. y Vinyals, O. (2015). Recurrent Neural Network Regularization. Recuperado de: <https://arxiv.org/pdf/1409.2329.pdf>

Zhang, G. P. (2000). Neural networks for classification: a survey. IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), 30(4), 451-462. Recuperado de: <https://pdfs.semanticscholar.org/67ad/9b3f3d91b101909297d79b912532446485c0.pdf>

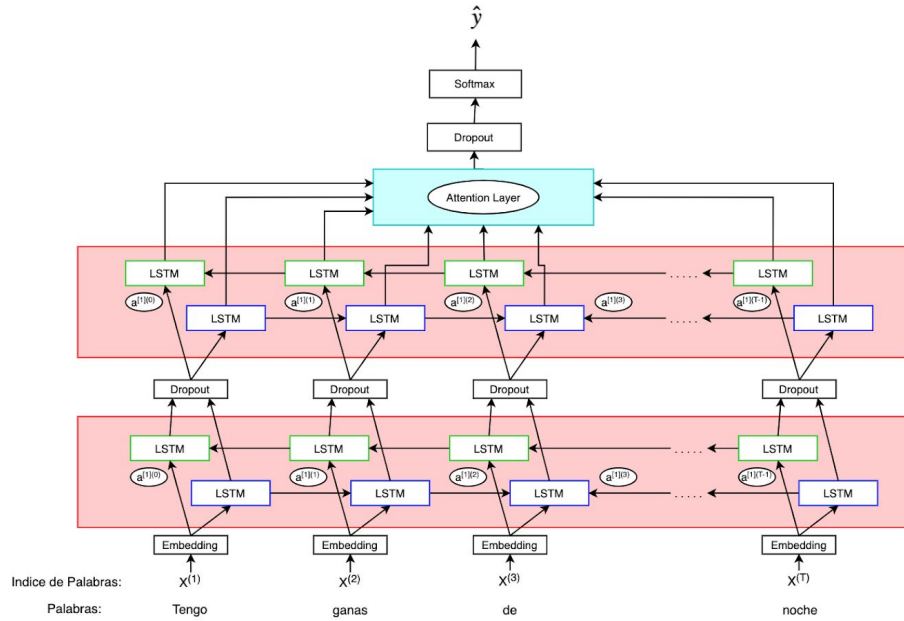
APÉNDICES

Apéndice A: Arquitectura del modelo

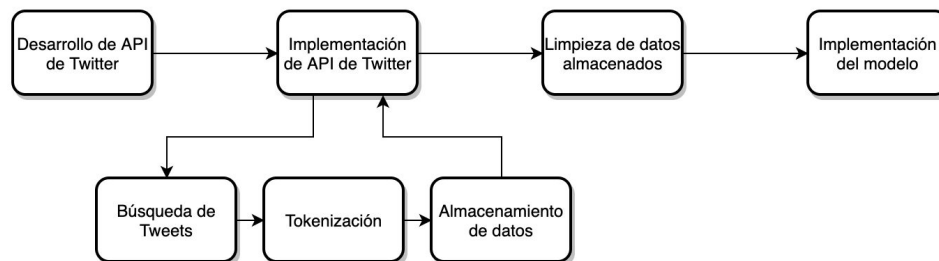
Model: "model_2"

Layer (type)	Output Shape	Param #
=====		
input_2 (InputLayer)	(None, 47)	0
embedding_1 (Embedding)	(None, 47, 50)	81604200
bidirectional_3 (Bidirection	(None, 47, 512)	628736
dropout_3 (Dropout)	(None, 47, 512)	0
bidirectional_4 (Bidirection	(None, 47, 512)	1574912
attention_2 (Attention)	(None, 512)	559
dropout_4 (Dropout)	(None, 512)	0
dense_2 (Dense)	(None, 22)	11286
activation_2 (Activation)	(None, 22)	0
=====		
Total params: 83,819,693		
Trainable params: 83,819,693		
Non-trainable params: 0		

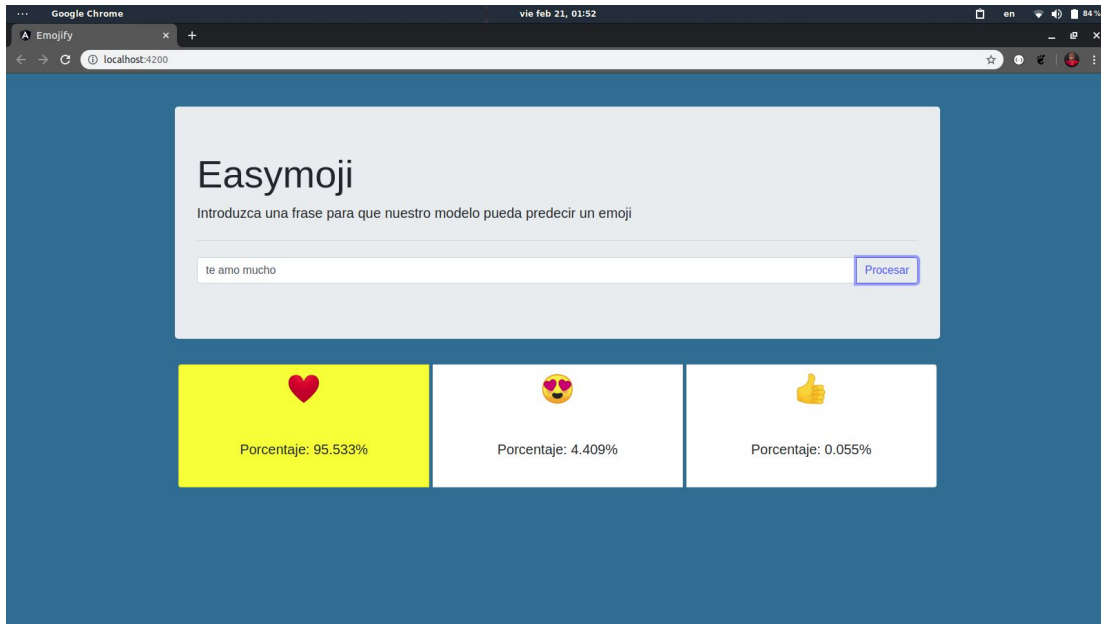
Apéndice B: Esquema del modelo



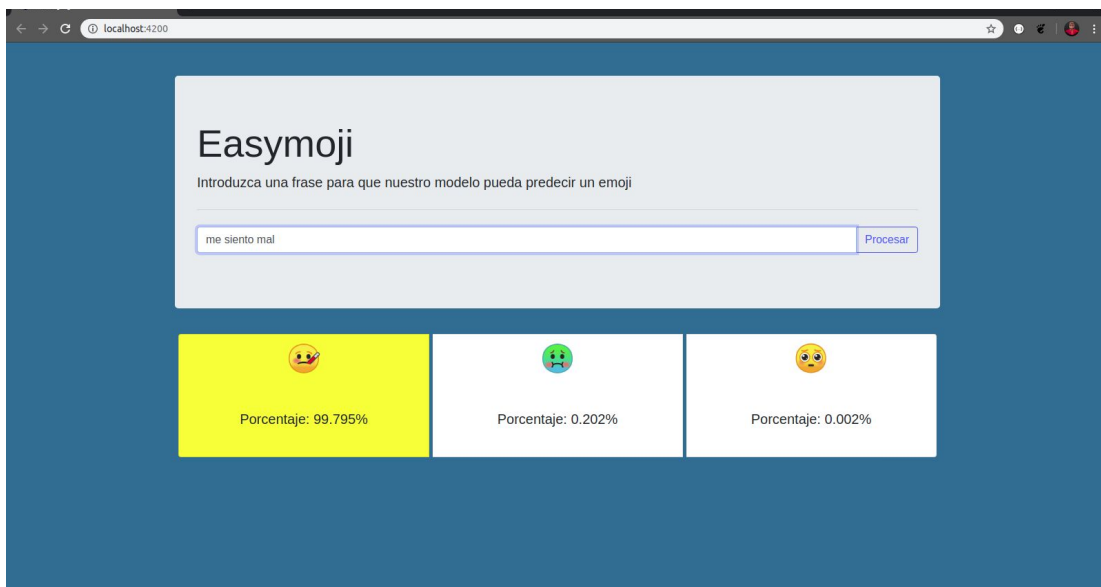
Apéndice C: Esquema de procesamiento de datos



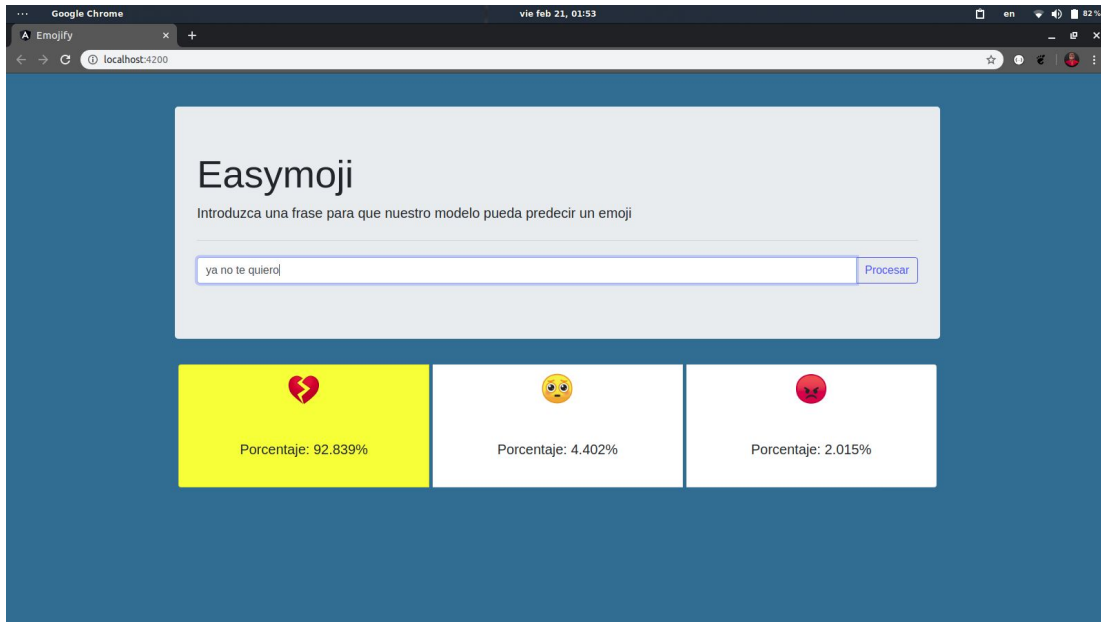
Apéndice D: Pruebas de software



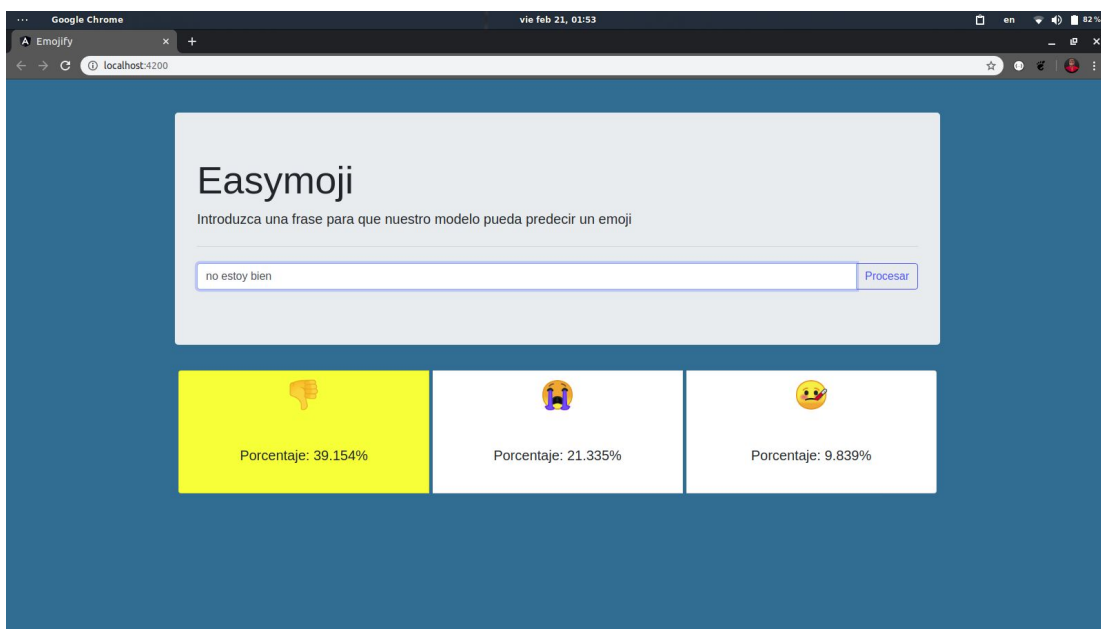
Apéndice E: Pruebas de software



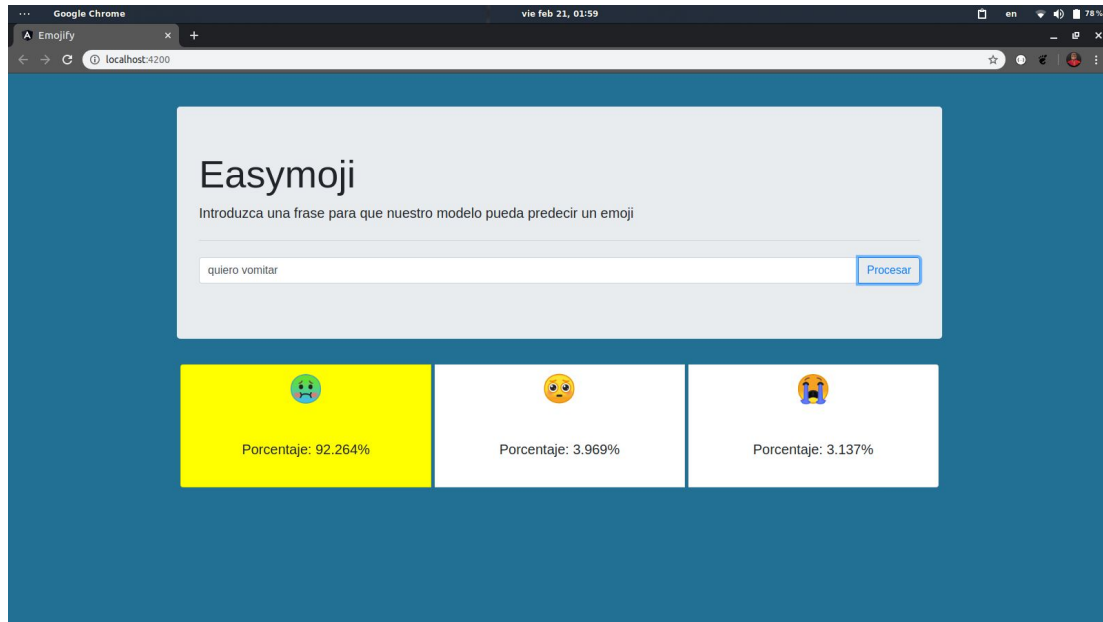
Apéndice F: Pruebas de software



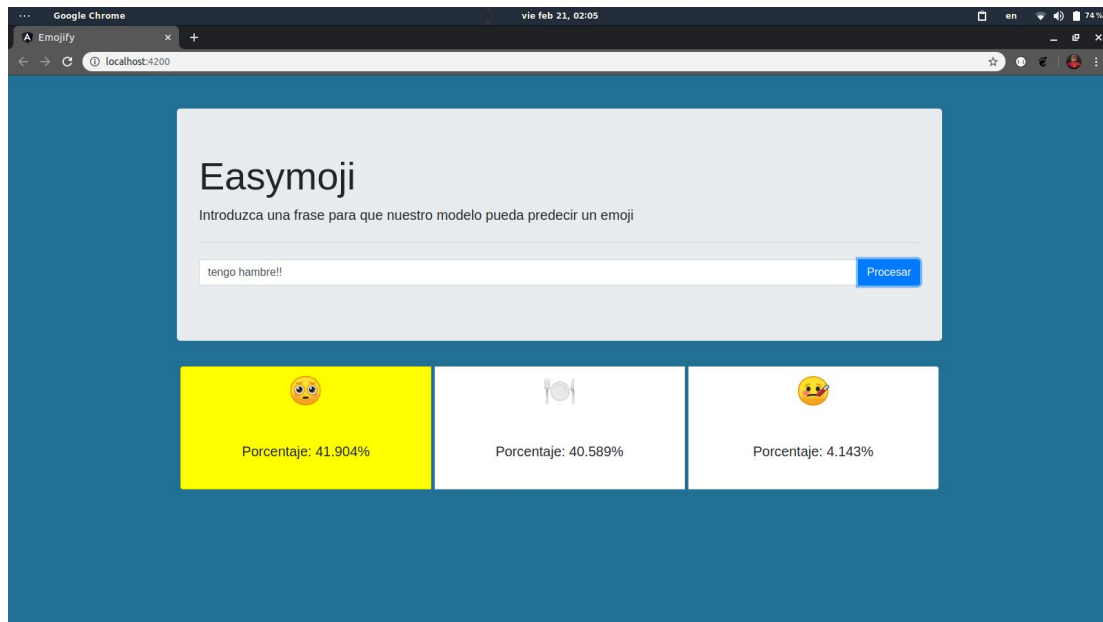
Apéndice G: Pruebas de software



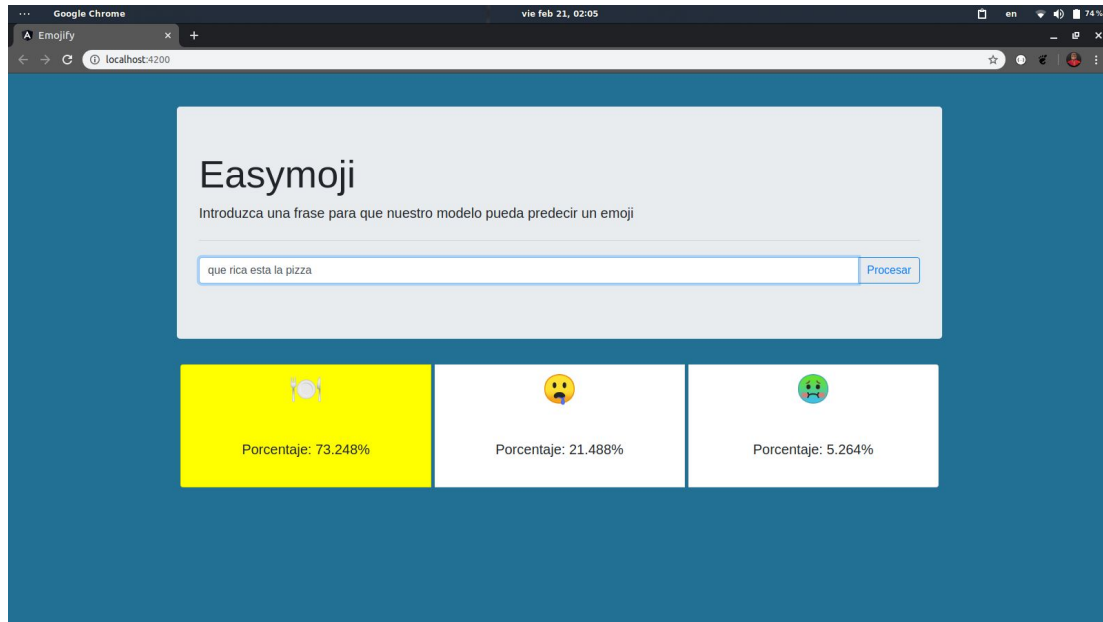
Apéndice H: Pruebas de software



Apéndice I: Pruebas de software



Apéndice J: Pruebas de software



Apéndice K: Pruebas de oraciones 1

```
1 # Probar oraciones.
2 x_test = np.array(['estoy muy contento'])
3 X_test_indices = sentences_to_indices2(x_test, maxLen)
4 print(x_test[0] + ' ' + label_to_emoji(np.argmax(model.predict(X_test_indices))))
5 print(label_to_emoji(np.argsort(model.predict(X_test_indices)).item(21)) + ' con un porcentaje de:', model.predict(X_test_indices).item(21))
6 print(label_to_emoji(np.argsort(model.predict(X_test_indices)).item(20)) + ' con un porcentaje de:', model.predict(X_test_indices).item(20))
7 print(label_to_emoji(np.argsort(model.predict(X_test_indices)).item(19)) + ' con un porcentaje de:', model.predict(X_test_indices).item(19))
```

1
estoy muy contento 🍴
🍴 con un porcentaje de: 0.9899066090583801
😞 con un porcentaje de: 0.004942398518323898
😬 con un porcentaje de: 0.00447567086668224

Apéndice L: Pruebas de oraciones 2

```
1 # Probar oraciones.
2 x_test = np.array(['esto no esta mal'])
3 X_test_indices = sentences_to_indices2(x_test, maxLen)
4 print(x_test[0] + ' ' + label_to_emoji(np.argmax(model.predict(X_test_indices))))
5 print(label_to_emoji(np.argsort(model.predict(X_test_indices)).item(21)) + ' con un porcentaje de:', model.predict(X_test_indices).item(21))
6 print(label_to_emoji(np.argsort(model.predict(X_test_indices)).item(20)) + ' con un porcentaje de:', model.predict(X_test_indices).item(20))
7 print(label_to_emoji(np.argsort(model.predict(X_test_indices)).item(19)) + ' con un porcentaje de:', model.predict(X_test_indices).item(19))
```

1
esto no esta mal 🍴
🍴 con un porcentaje de: 0.9993751645088196
😞 con un porcentaje de: 0.0005898101953789592
😬 con un porcentaje de: 1.3291722098074388e-05

Apéndice M: Pruebas de tweets 3

```
1 # Probar oraciones.
2 x_test = np.array(['que felicidad tengo'])
3 X_test_indices = sentences_to_indices2(x_test, maxLen)
4 print(x_test[0] + ' ' + label_to_emoji(np.argmax(model.predict(X_test_indices))))
5 print(label_to_emoji(np.argsort(model.predict(X_test_indices)).item(21)) + ' con un porcentaje de:', model.predict(X_test_indices).item(21))
6 print(label_to_emoji(np.argsort(model.predict(X_test_indices)).item(20)) + ' con un porcentaje de:', model.predict(X_test_indices).item(20))
7 print(label_to_emoji(np.argsort(model.predict(X_test_indices)).item(19)) + ' con un porcentaje de:', model.predict(X_test_indices).item(19))
```

1
que felicidad tengo 😊
😊 con un porcentaje de: 0.9844958782196045
👉 con un porcentaje de: 0.0034743351861834526
❤️ con un porcentaje de: 0.0034029430244117975

Apéndice N: Código Jupyter Notebook de análisis de sentimiento

```
In [ ]: 1 from classifier import *
        2 import pandas as pd

In [ ]: 1 data = pd.read_csv('data/data.csv')

In [ ]: 1 clf = SentimentClassifier()

In [ ]: 1 X = data['texto']
        2 Y = data['indice']

In [ ]: 1 for frase in X:
        2     print(frase, '==>', clf.predict(frase))

In [ ]: 1 sent = []
        2 for frase in X:
        3     sent.append(clf.predict(frase))

In [ ]: 1 df = pd.DataFrame(data)

In [ ]: 1 df['sent'] = sent

In [ ]: 1 df

In [ ]: 1 df.to_csv('sentimiento.csv', index = False)
```