28 DE SEPTIEMBRE DE 2022

DETECCIÓN DE EMOCIONES EN TEXTOS

LUIS ANDREU, OLIMPIA FUSTER,
LAURA MACÍAS Y RAFA PÉREZ

TUTOR: RAFAEL LÓPEZ

EDEM
Escuela de Empresarios

ÍNDICE

Abstract

La cantidad de textos fácilmente accesibles en diferentes idiomas en Internet crece día a día, como también lo hace el esfuerzo y la necesidad de organizar dichos textos. No obstante, se necesitan procesos automatizados para poder extraer la información útil de las palabras, así como sentimientos, ironía u odio. En este trabajo, nos centramos en identificar las emociones alegría, tristeza, enfado, sorpresa y miedo dentro de los textos, tanto en inglés como en castellano. Para ello, se realiza un previo preprocesamiento de los datos para que puedan ser mejor interpretables para posteriormente aplicar diferentes algoritmos y encontrar el modelo de predicción óptimo.

Palabras Clave

Procesamiento del lenguaje natural, NLP, aprendizaje automático, análisis de sentimientos, detección de emociones.

Introducción

Cada vez más, las marcas quieren saber qué es lo que piensan los consumidores sobre ellas. Una de las formas para conseguirlo, es realizar un sondeo en internet: recopilar textos como tuits, comentarios en publicaciones, blogs, etc., en los que la gente exprese como se siente en relación a la marca.

Estos textos a menudo contienen un sentimiento determinado, aunque uno por sí solo no es muy expresivo. Pero si un rastreador web recopila una cantidad significativa de estos textos, un análisis de sentimientos puede proporcionar información útil.

Centrándonos en los tuits, estos son extremadamente cortos, como máximo 140 caracteres; el lenguaje utilizado es tan informal que a menudo contiene errores ortográficos; las publicaciones tratan muchos temas diferentes y el punto de vista es bastante subjetivo. Estas circunstancias hacen que sea más difícil determinar una clasificación precisa de este tipo de texto en comparación con textos bien escritos.

La mayor parte del corpus está formada por tuits escritos en inglés. Dado que Twitter también es popular en países de habla no inglesa, se puede utilizar como fuente de textos informales en español.

El uso del idioma español en el análisis de sentimientos es especialmente desafiante, ya que las reglas del idioma son más complejas en comparación con el inglés. Por ejemplo, hay más variedad en cuanto a femenino y masculino (<la>, <el>, <the>).

Este documento abordará la cuestión de crear métodos de análisis de sentimientos en diferentes textos tanto en inglés como en español (especialmente tuits) y cómo hacerlo. Se investigan diferentes pasos de preprocesamiento y extracción de características y se prueban algoritmos de clasificación en una serie compleja de experimentos.

Marco Teórico

El procesamiento del lenguaje natural (PLN) es la rama de la inteligencia artificial que se encarga de desarrollar algoritmos que permitan extraer información relevante a partir de un texto. El "PNL" permite comprender a los usuarios y tiene aplicación oral y escrita. A menudo se le relaciona con herramientas como búsquedas web, traducción automática de texto o de voz, comprobación ortográfica o el análisis de sentimiento. Esta última es en la que nos centraremos en este trabajo y mostraremos las distintas complejidades que puedan surgir.

Por otro lado, el PLN (*o NLP, por sus siglas en inglés*) depende en gran medida de la inteligencia artificial (IA), donde un análisis del sentimiento se realiza mediante un algoritmo de aprendizaje automático.

Es un hecho que el texto puede ser una fuente de datos de mucho provecho, sin embargo, debido a su naturaleza no estructurada, extraer la información de este puede llegar a ser muy complejo y llevar bastante tiempo. Para ello, se recolectan los datos textuales (textos o secuencias) como "input" o datos de entrada y, posteriormente, se etiquetan para poder hacer uso y reconocimiento.

El texto se lee y codifica de forma que la "máquina" pueda entenderlo y procesarlo. Estos nuevos datos conforman el input y el modelo de predicción devuelve el sentimiento asignado como dominante en el texto (en forma de "output"*)*.

No obstante, este procedimiento tiene un "hándicap": cómo tratar de la existencia de los fenómenos discursivos (como la ironía, el sarcasmo, etc.) que existen en el lenguaje natural. Posteriormente, explicaremos cómo se abordará el procesamiento de los textos *(o corpus)* más en detalle.

En los trabajos relacionados con la clasificación de texto para el análisis del sentimiento, comúnmente se ofrecen soluciones binarias, es decir, si el texto tiene un sentido positivo o negativo. En nuestro caso, el objetivo se centra en una clasificación de 5 categorías: alegría, enfado/ira, sorpresa, tristeza y miedo. En otras palabras, se buscan soluciones para la clasificación de textos multietiqueta o multiclase.

Cuando de datos textuales (corpus) se trata, los valores deben tratarse como una serie de palabras en lugar de procesar las palabras independientemente. Esto logrará obtener cualidades como el significado de una palabra según un contexto determinado. Para ello, el modelo "Transformer" introducido en 2004, tiene una solución que se basa en seleccionar las palabras "más determinantes" para un significado en términos generales.

Corpus

Dado que el corpus es extremadamente importante para el desempeño de la tarea de clasificación, es necesario observar más de cerca el corpus que se utiliza. Los corpus de Twitter escritos en inglés se pueden encontrar con bastante frecuencia y facilidad, pero un corpus en español completamente anotado es más difícil de encontrar.

Hay algunos corpus en castellano, pero estos sólo contienen clases: positivas o negativas y, como mucho, neutras. Además, cabe destacar la diferencia entre los titulares de noticias con los tuits, ya que un texto de noticias está escrito con más cuidado y utiliza un vocabulario completamente diferente; además, rara vez contiene emoticones, hecho en el que nos centramos en este trabajo, jergas o errores ortográficos.

Inglés

En cuanto al corpus en inglés, este está compuesto por una combinación de distintos datasets. El dataset principal es Emotion Detection from Text, disponible en Kaggle. El archivo contiene aproximadamente 40.000 registros divididos en 13 clases de emociones. De aquí se escogieron las 5 que nos interesaban y se sustrajo el dataset base.

Como era de esperar, los datos no estaban balanceados, por lo que se procedió al balanceo de las clases más minoritarias. Para ello, se buscaron datasets similares que pudiesen completar las categorías con menores registros. Para consultarlos, dirigirse a la webgrafía.

El conjunto de datos en su groso está formado por tuits, sin estar relacionados a un tema en concreto. Como veremos posteriormente, los @usuarios y diferentes elementos son eliminados en el preprocesamiento para que no influyan en el aprendizaje del modelo.

El conjunto resultante ha sido un total de 57.000 con la siguiente distribución:

EMOCIÓN	REGISTROS	% DEL TOTAL
Happy	12238	21%
Anger	11599	20%
Sadness	11430	20%
Fear	11111	19%
Surprise	10622	19%

Castellano

En cuanto al castellano, el corpus se llama TASS y fue la primera tarea compartida sobre el análisis de sentimiento en Twitter en español. Este conjunto de datos consiste en tuits extraídos de los diferentes subconjuntos de España, Perú, Costa Rica, Chile, Uruguay y México.

El conjunto de datos se basa en eventos que ocurrieron en abril de 2019, entre ellos, eventos de entretenimiento, políticos, conmemoraciones globales y huelgas globales. Los hashtags y las menciones de usuarios ya se transforman en palabras clave: “HASHTAG” y “@USER” porque estos eventos están polarizados.

Este conjunto de datos contiene un total de 8.409 tuits escritos en español, sin embargo, no todas las categorías son de nuestro interés donde el entrenamiento, la validación y la prueba ya están divididos para nosotros.

Alemán y Portugués

En este caso, este corpus no es utilizado para el sentiment analysis como los dos anteriores, sino que se usa para el Machine Translation. Ambos siguen la misma estructura: una columna con la frase en inglés, otra con la traducción en el idioma correspondiente y una última con detalles del registro.

El dataset alemán contiene un total de 255.817 frases traducidas, mientras que el portugués tiene 184.998. Gracias a estos data sets, podemos entrenar un modelo de Machine Translation que traduzca nuestros datasets ya etiquetados al alemán o portugués, para luego poder pasarlos por nuestros modelos de sentiment analysis.

Preprocesamiento

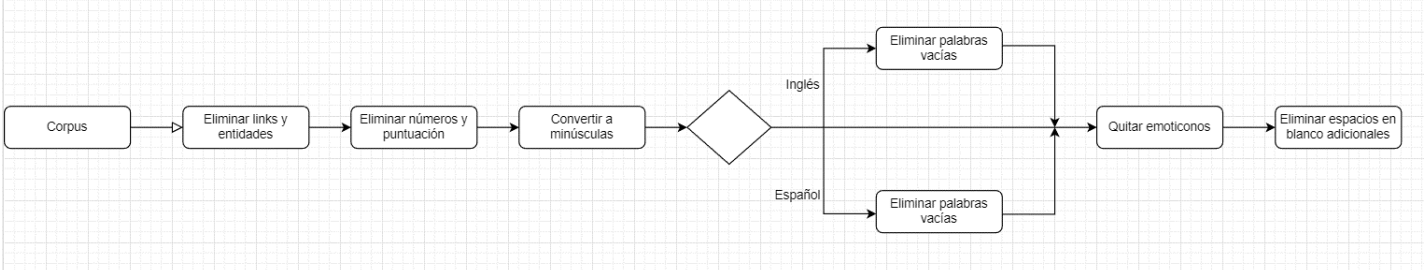
El preprocesamiento de los tuits contiene varios pasos que son comunes al procesamiento del lenguaje natural. Ahora, se describirán con más detalle.

En ambos corpus se ha realizado un preprocesamiento muy similar. En primer lugar, se han eliminado hipervínculos, hashtags, el nombre de los usuarios y los signos de puntuación, ya que son símbolos innecesarios que no aportan a la emoción del tuit.

Seguidamente, convertimos todos los tuits a minúsculas, así, garantizamos que tengamos un conjunto de datos uniforme, además de eliminar números y otros valores numéricos. De esta forma aseguramos que sólo el texto que permanece en el conjunto de datos agrega valor al modelo.

El siguiente paso en el proceso es eliminar las palabras vacías: estas no contribuyen al significado de una oración, ya que son comunes en un idioma. Usamos una lista de palabras vacías en español e inglés proporcionadas por el marco NLTK. La eliminación de palabras vacías permite que el modelo NLP se centre en palabras únicas en los textos que agregarán valor.

Finalmente, dividimos el texto en palabras o en frases más pequeñas conocidas como tokens. En la siguiente figura, podemos ver una descripción general de los pasos de preprocesamiento y sus variaciones.



Modelo Bert

Para el desarrollo de este trabajo, se ha realizado una búsqueda exhaustiva de diversos modelos para la clasificación de sentimientos que pudieran encajar con los requerimientos. Tras un análisis de los modelos encontrados, tomamos la decisión de utilizar el modelo pre-entrenado BERT para poder generar un modelo de identificación de emociones más preciso.

Lo más importante de BERT es su división en dos bloques:

1º Basado en las redes Transformer que permite el entrenamiento básico del modelo.
2º Ligado al anterior, se encarga de afinar el funcionamiento del sistema mediante deep learning.

En otras palabras, el primer bloque reconoce el lenguaje y el segundo le da la versatilidad para obtener resultados. En términos técnicos, este proceso engloba un codificador que lee la entrada del texto y un decodificador para la predicción de la tarea.

Dentro de BERT podemos encontrar dos ramas: BERT BASE que utiliza una arquitectura con 110 millones de parámetros y BERT LARGE, que utiliza 345.000 millones de parámetros. Para nuestro caso de uso, el BERT BASE era la mejor opción: cuantos menos parámetros más velocidad de procesamiento.

Otros enfoques más antiguos como word2vec o Glove fueron investigados, pero finalmente se desestimaron debido a que los resultados eran siempre peores. BERT basa su análisis en el entorno de la palabra para poder comprenderlo, y el contexto del texto que queremos analizar. Mientras tanto, otros modelos y enfoques se basan en leer la entrada del texto secuencialmente (de izquierda a derecha o viceversa), por lo que a fin de cuentas hemos considerado BERT como mejor opción, ya que se considera un modelo bidireccional. Aquí os dejamos una comparación entre diferentes Transformers:

TRANSFORMER	ACCURACY	F1_SCORE	MAX_LENGTH	EPOCHS
BERT	0,6875	0,58	8	45
RoBERTa	0,6790	0,54	8	40
DistilBERT	0,6709	0,57	8	45

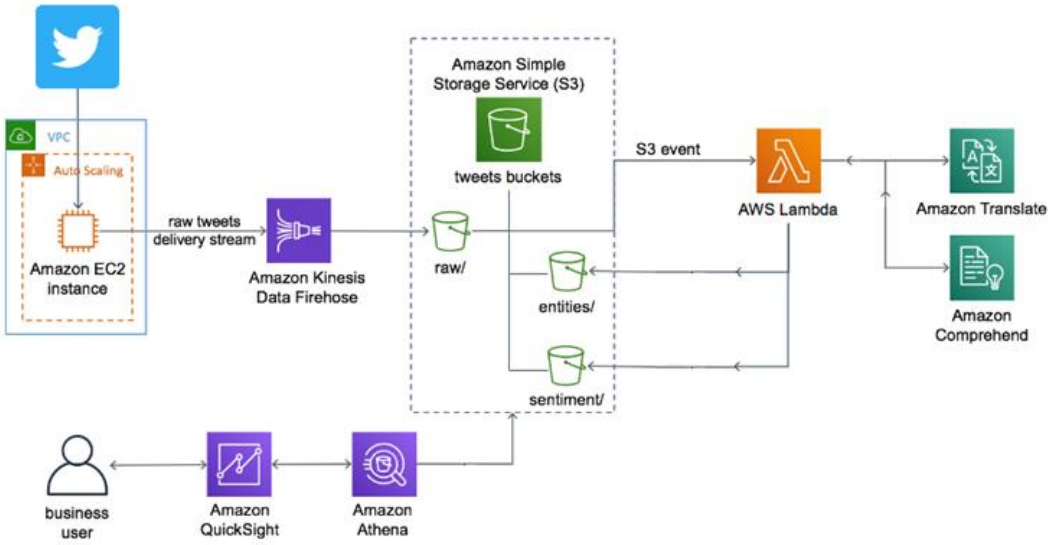
A pesar todo, este modelo también presenta desventajas: sólo tiene en cuenta la predicción de los valores enmascarados y no la de los no enmascarados (utiliza un 15% de las palabras secuenciales para la predicción). Esto hace que el modelo converja más lentamente que otros direccionales, y, en consecuencia, el tiempo de espera para obtener los resultados es más elevado. Esta espera también se verá condicionada por el tamaño del conjunto de datos y de los epochs que se valoren utilizar.

Después de haber seleccionado el mejor modelo de pre-entreno, lo siguiente más importante a la hora de predecir es la optimización de los parámetros que nos permitan

la obtención del mayor accuracy (o f1_score) del modelo. Los parámetros considerados más relevantes han sido batch size, shape y los epochs.

En cuanto al batch size, hemos alternado tamaños de 16, 32 y 56. No hemos podido jugar con mayores tamaños debido al tamaño de la RAM de la herramienta utilizada. Hablando del shape, tras varias pruebas, hemos decidido mantener una forma de 512, ya que era la más recomendada por los expertos. En cuanto a los epochs, hemos tenido más flexibilidad, ya que aumentábamos la cantidad dependiendo del accuracy obtenido.

Arquitectura de implementación



Para este proyecto, hemos pensado que AWS era la opción idónea para desplegar la arquitectura debido a las facilidades y variedad de aplicaciones ofrecidas. Brevemente, el proceso sería el siguiente:

Mediante la API de Twitter (o cualquier otra API de textos que se desee), la instancia de EC2 proporcionaría capacidad de computación escalable en la nube de AWS. Recibiríamos los tuits y, dependiendo del volumen de datos, la instancia EC2 escalaría automáticamente.

Con la ayuda de Amazon Kinesis Data Firehouse, ingerimos, almacenamos y procesamos los datos en tiempo real, los cuales pasarían a un bucket de S3, aplicación para el almacenamiento. Una vez almacenados en el bucket, dentro de la "carpeta" raw, se generaría un evento que desencadenaría una función en AWS Lambda.

Esta función movería los datos de S3 a Amazon Translate para traducir los textos (y poder generar así nuevos datasets) y a Amazon Comprehend, dónde se desplegarían todas las técnicas de NLP para poder predecir la clase (sentimiento). Cuando se tuviera generado el mejor modelo y los datos procesados, recorrerían el camino inverso para almacenarse de vuelta en el bucket de S3.

Para acceder a los nuevos datos, se haría por medio de lenguaje SQL a través de la herramienta Amazon Athena. Finalmente, visualizaríamos estas queries mediante Quicksight, de forma que el cliente pueda tomar decisiones informadas.

Líneas futuras y recomendaciones

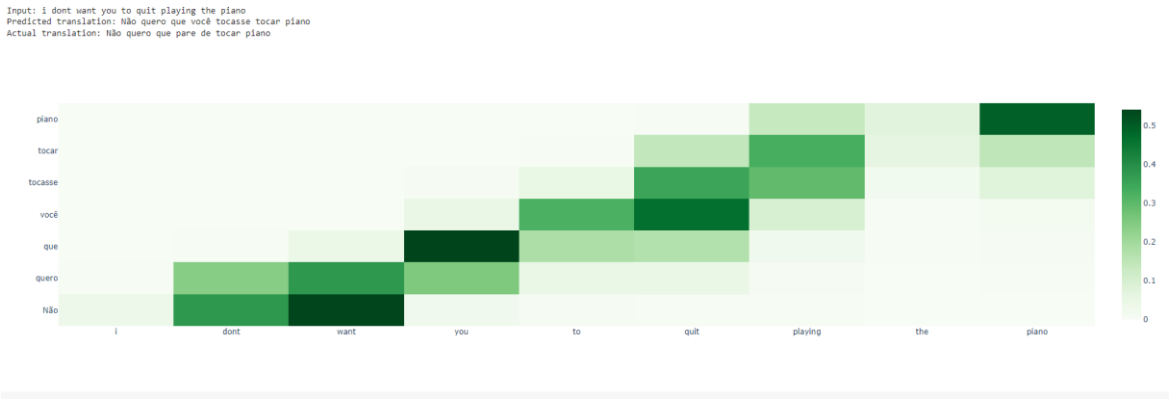
Somos conocedores de que las arquitecturas en Cloud cada vez están más demandadas, por lo que recomendamos que, en un futuro, se implementara una solución similar a la previamente expuesta para automatizar las predicciones.

A pesar de la grandísima utilidad de los modelos de predicción y el NLP, hemos considerado de utilidad el tener alternativas a ellos. Para ello, hemos desarrollado una serie redes neuronales capaces de traducir textos a Portugués y Alemán, siendo objetivo futuro de implementación añadir más idiomas.

Estos modelos de traducción no solo traducen, si no que nos permiten generar nuevos datasets, los cuales podemos almacenar y utilizarlos para los algoritmos del sentiment analysis. Todo esto es un valor añadido para la empresa, por lo que recomendamos el desarrollo de este tipo de tecnología, ya que en ocasiones encontrar corpus de ciertos idiomas es más complicado.

El *Machine Translator* o traducción automática, consiste en convertir automáticamente un idioma natural a otro, conservando el significado del texto de entrada y produciendo un texto fluido en el idioma de salida.

Empleamos la traducción automática mediante unidades recurrentes cerradas (GRU), un modelo popular que se usa en traductores web de nivel industrial debido a la eficiencia con la que maneja datos secuenciales. En comparación a esto encontramos la memoria a corto plazo (LSTM), útil en el modelado de lenguaje con conjuntos de datos más pequeños. Una salida de la traducción automática sería la siguiente:



Conclusiones

En primer lugar, la búsqueda de corpus en otros idiomas diferentes al inglés es un reto complicado de superar, más aun exigiendo datasets etiquetados con las emociones requeridas (alegría, tristeza, miedo, sorpresa e ira). Las búsquedas exhaustivas no siempre han dado resultados y se ha invertido mucho tiempo en la búsqueda de textos asociados a su sentimiento específico.

En segundo lugar, en el mundo del NLP existe una variedad de modelos pre-entrenados que podrían ser considerados a la hora de realizar el sentiment analysis. De todos los analizados, concluimos con que BERT es el mejor de todos, ya que no solo tiene en cuenta el significado del texto per se, si no que el entorno (o contexto) también influye.

Una vez elegido el modelo pre-entrenado, el desarrollo del modelo propio ha sido relativamente sencillo. Tras varias combinaciones de parámetros en los diferentes datasets (inglés y castellano), los mejores modelos han sido los siguientes:

DATASET	F1_SCORE	BATCH	SIZE	EPOCHS
Inglés	0,875	16	512	2
Castellano	0,72	6	512	5

En cuanto a los datasets Alemán y Portugués, no hemos podido obtener resultados, ya que la capacidad de RAM no era suficiente para soportar el peso de los datos y no permitía procesar la información.

Finalmente, se trata la recomendación del uso del Machine Translation como línea de actuación futura para la empresa Atribus. Se ha llegado a la conclusión de que esta línea de aprendizaje automático podría ser muy útil en el ámbito empresarial, ya que se pueden obtener muchos más datos para poder utilizarlos en el sentiment analysis y predecir emociones de manera más certera.

Links Al Trabajo

GitHub: https://github.com/olimpiaf99/TFM

Video: https://www.youtube.com/watch?v=6Aem6bysFS8

Presentación Canva: https://www.canva.com/design/DAFMlh70RyY/73vqkJ0p42HJPGdqgJrXbg/view?utm_content=DAFMlh70RyY&utm_campaign=designshare&utm_medium=link&utm_source=publishsharelink

Webgrafía

WordPress. (2020). *TASS*. Obtenido de tass.sepln.org: http://tass.sepln.org/2020/?page_id=74

Hugging Face. (s.f.). Obtenido de https://huggingface.co/models

Alammar, J. (2018). The illustrated BERT, ELMo, and co. (how NLP cracked transfer learning). Obtenido de http://jalammar.github.io/illustrated-bert

Transformer neural networks - (attention is all you need) (2020). CodeEmporium (Director). Obtenido de https://www.youtube.com/watch?v=TQQlZhbC5ps

Devlin, J., & Chang, M. (2018).Open sourcing BERT: State-of-the-art pre-training for natural language processing. Recuperado de http://ai.googleblog.com/2018/11/open-sourcing-bert-state-of-art-pre.html

Devlin, J., Chang, M., Lee, K., & Toutanova, K. (2018). BERT: Pre-training of deep bidirectional transformers for language understanding. Recuperado de https://arxiv.org/abs/1810.04805v2

Rizvi, M. S. Z. (2019). Demystifying BERT: The groundbreaking NLP framework. Recuperado de https://medium.com/analytics-vidhya/demystifying-bertthe-groundbreaking-nlp-framework-8e3142b3d366

6Xiao, H. (2019). Serving google BERT in production using tensorflow and ZeroMQ . Recuperado de https://hanxiao.io/2019/01/02/Serving-Google-BERT-inProduction-using-Tensorflow-and-ZeroMQ/

Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. Recuperado de https://doi.org/10.1016/j.asej.2014.04.011

Dataset links:

http://www.manythings.org/anki/

https://www.site.uottawa.ca/~diana/resources/emotion_stimulus_data/

https://github.com/declare-lab/MELD/tree/master/data

https://www.kaggle.com/datasets/pashupatigupta/emotion-detection-from-text

https://aclanthology.org/I17-1099/

https://github.com/google-research/google-research/tree/master/goemotions/data