Proyecto #1 Analítica de Textos

Predicción de Emociones

Autores:
Miguel Ángel Acosta (201914976)
Andrés Felipe Rincón (201914118)
Ángela Liliana Jiménez (201912941)

Tabla de Contenidos

Comprensión del negocio y enfoque analítico

Oportunidad o Problema del negocio	Para cualquier negocio es importante saber cómo se sienten sus clientes con respecto a sus productos y servicios. Los post en redes sociales sobre la empresa proveen información pero falta interpretarla, en este contexto es útil el análisis de emociones, que le permitirá a una empresa conocer cómo se están sintiendo sus cliente o perfilar un grupo de clientes, para luego tomar decisiones de negocio con base a esta información
Descripción del requerimiento desde el punto de vista de aprendizaje de máquina	Como se dijo se requiere realizar un análisis de emociones. Este requerimiento implica tomar un conjunto de datos identificados cada uno con una emoción y entrenar un programa o modelo de IA que pueda ante la llegada de nuevos datos identificar adecuadamente el sentimiento al cual se asociaría la opinión de un post en redes sociales.

Detalles de la actividad de minería de datos

Tarea	Técnica	Algoritmo e hiper-parámetros utilizados
Aprendizaje Supervisado	Clasificación Multiclase	1. Naive Bayes hp: alpha 2. SVM (Support vector machine)

		hp: C, Kernel 3. OneVsRest hp: C, Kernel Los hiper parámetros se calculan por medio de grid search

Comprensión de los datos y preparación de los datos

En total tenemos 16000 datos que corresponden a post en redes sociales en inglés y que tienen asociados una etiqueta que indica la emoción que se le asoció a ese post.
En total tenemos 5 emociones: Joy, Sadness, Anger, Fear, Love y Surprise y están distribuidas de la siguiente forma:

Emoción	Conteo
sadness	4666
anger	2159
love	1304
surprise	572
fear	1937
joy	5362

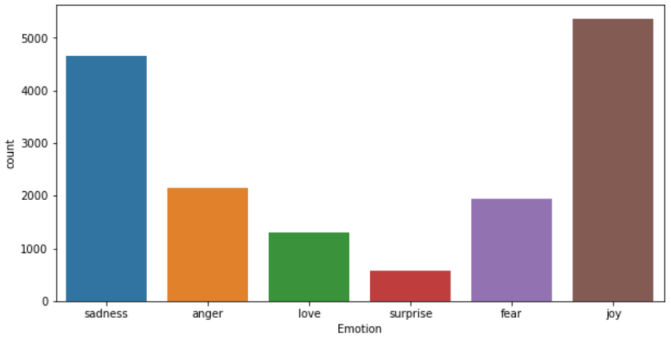

Figura 1: Conteo de post por emoción

Lo que podemos ver es una muestra de mínimo 572 y máximo 5362 datos por emoción, así como un promedio de estos de 2666 datos por emoción. Nos quedaremos con esta distribución de los datos porque para cada emoción hay una muestra lo suficientemente grande para ser representativa en nuestros algoritmos.
También es importante revisar que los datos tengan sentido, en este caso revisaremos eso con un análisis de emociones (usando la librería TextBlob). La intención de este análisis es clasificar los posts en negativo, neutral o positivo y dependiendo de la emoción veremos la clasificación dada por el dataset de entrenamiento es consistente.

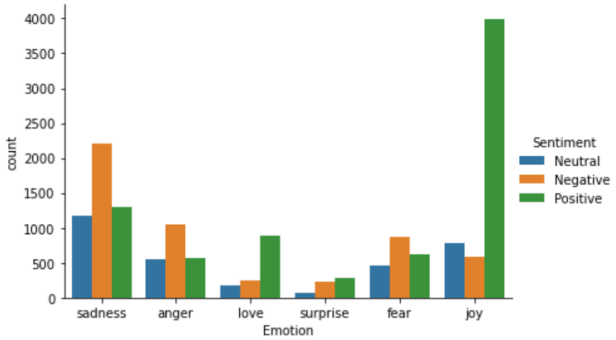

Figura 2: Análisis de sentimientos

Para explicar la consistencia hablada anteriormente se toma como ejemplo joy, como se puede evidenciar la mayoría de los posts de joy son positivos, esto tiene total sentido considerando que joy es alegría. Un análisis similar se realiza para los demás sentimientos y verificamos que son coherentes.

En el **pre-procesamiento** se va a realizar 3 técnicas de minería de textos sobre los posts de entrenamiento: tokenización, limpieza y normalización. En esta sección tomaremos como ejemplo el pre-procesamiento que se realizó con un elemento del dataset de entrenamiento “i didn't feel humiliated”. Este proceso se realizará para cada post y finalmente juntamos los resultados hallados.

1. Tokenización: Dividimos la frase en tokens o palabras

   “i didn't feel humiliated” → [i, did, not, feel, humiliated]

2. Limpieza de texto: En esta sección se quitará o modificará todo lo que se considere como ruido: Corregimos las contracciones (you're -> you are), quitamos caracteres no ascii, se pasará de mayusculas a minusculas, se eliminará la puntuación, se reemplazarán los números, se quitarán las palabras vacías (artículos, pronombres, preposiciones)

   [i, didn't, feel, humiliated] → [feel, humiliated]

3. Normalización: Se utiliza Lematize, que haya un representante de un grupo de palabras flexionadas, ej: (correrá, correr, corrió) → corre. Y se usa Stem que consiste en conseguir una palabra de la que parten otras al extenderse (correrá, correr, corrió) → corr.

   [feel, humiliated] → [feel, humy, feel, humiliate]

Lo que tenemos en este punto es el data frame limpio con las técnicas de minería básicas, pero vale la pena revisar el resultado de lo obtenido para ver si es posible tomar alguna decisión de negocio.

Primero podemos analizar las palabras más significativas para una emoción, por lo que creamos un super texto con todos los post relacionados a una determinada emoción y luego realizaremos un conteo de sus palabras más usadas. En esta sección tomaremos como ejemplo la emoción love.

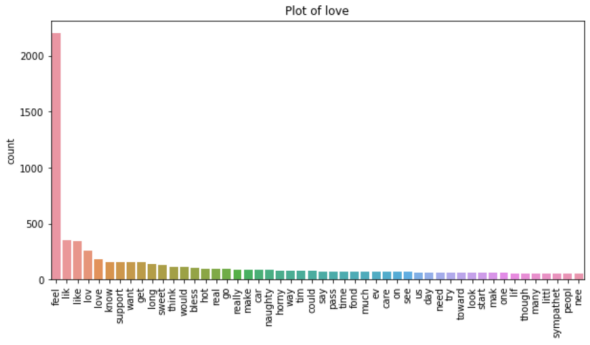

Figura 3: Palabras más significativas para love (con feel y like)

Acá podemos evidenciar como hay unas palabras que parecen repetirse en casi todos los posts (63/1304), se realiza el mismo experimento con las otras emociones y encontramos que son palabras que aparecen al parecer en la mayoría de posts, por lo que se toma la decisión de retirarlas para que no hagan ruido porque al estar

en todos los posts, no están brindando información que permita clasificar si el post corresponde a una u otra emoción.

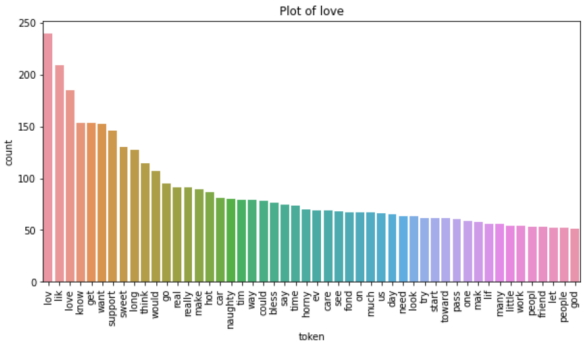

Figura 4: Palabras más significativas para love (sin feel y like)

Una representación alternativa de los datos se puede ver en la siguiente nube de palabras (las palabras con mayor tamaño de letra son las más significativas):

Figura 5: nube de palabras de love (sin feel y like)

Más adelante se realizaron pruebas que evidenciaron que no considerar estas palabras para la creación de modelos nos entregaba modelos de predicción más fiables.

Modelado y Evaluación

Se trabajarán 3 modelos de clasificación multiclase:

Naive Bayes:

Es un método de clasificación que supone fuerte independencia entre sus características, es decir que este método asume que x característica particular de una clase, no está relacionada con cualquier otra característica. Este método es adecuado cuando se quiere realizar una clasificación con características discretas y/o un conjunto de set grande. Particularmente usamos MultinomialNB que es usado para modelos multi clase

HiperParametros:

- alpha: Es la variable que usa el suavizado de Laplace para abordar el problema de la probabilidad cero, sus valores varían entre 0 y 1

Resultado (**score = 0.518**):

	precision	recall	f1-score	support
anger	0.60	0.31	0.40	275
fear	0.66	0.23	0.34	212
joy	0.56	0.71	0.62	704
love	0.27	0.41	0.33	178
sadness	0.53	0.60	0.56	550
surprise	0.33	0.01	0.02	81
accuracy			0.52	2000
macro avg	0.49	0.38	0.38	2000
weighted avg	0.53	0.52	0.50	2000

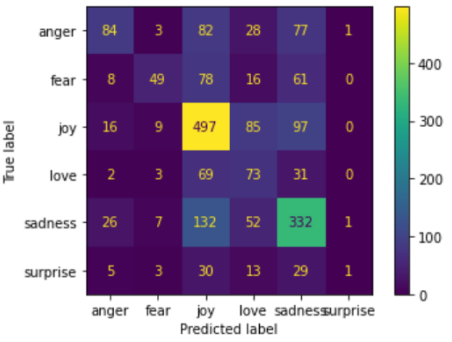

Figura 6: Resultados de Naive Bayes

SVC (Support vector machine):

Es un método de clasificación que busca devolver el hiperplano de "mejor ajuste" que divide o categoriza sus datos. A partir de ahí, después de obtener el hiperplano, puede enviar algunas características a su clasificador para ver cuál es la clase "predicha".

- C: El parámetro c le dice a la optimización SVM que tanto se quiere evitar la clasificación incorrecta de los ejemplos de entrenamiento.
- Kernel: Especifica el kernel que se usará, puede ser ‘linear’, ‘poly’, ‘rbf’, ‘sigmoid’ o ‘precomputed’.

Resultado (**score = 0.6825**):

	precision	recall	f1-score	support
anger	0.85	0.47	0.60	275
fear	0.81	0.50	0.62	212
joy	0.60	0.93	0.73	704
love	0.83	0.43	0.57	178
sadness	0.74	0.70	0.72	550
surprise	0.86	0.15	0.25	81
accuracy			0.68	2000
macro avg	0.78	0.53	0.58	2000
weighted avg	0.73	0.68	0.66	2000

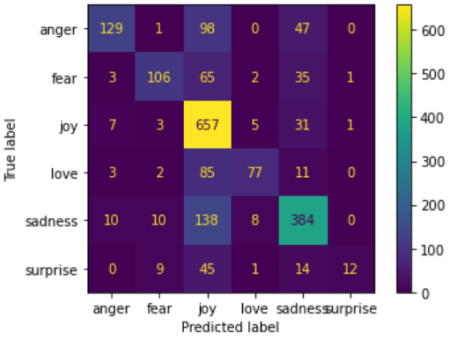

Figura 7: Resultados de SVC

OneVsRest:

La estrategia de este algoritmo es realizar un clasificador por clase, es decir realiza un clasificador por ‘love’, ‘joy’, etc, cada clasificador es comparado con los otros clasificadores y es un buen algoritmo para darle una interpretación propia a cada clasificador.

En este caso los hiperparametros son los mismos de SVC, porque se usó este algoritmo para realizar los clasificadores por clase.

Resultado (**score = 0.7025**):

	precision	recall	f1-score	support
anger	0.81	0.51	0.62	275
fear	0.84	0.54	0.66	212
joy	0.63	0.92	0.75	704
love	0.81	0.55	0.66	178
sadness	0.75	0.70	0.73	550
surprise	0.84	0.20	0.32	81
accuracy			0.70	2000
macro avg	0.78	0.57	0.62	2000
weighted avg	0.73	0.70	0.69	2000

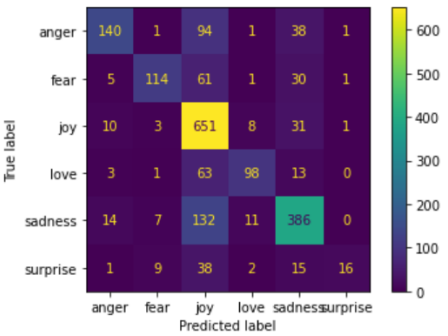

Figura 8: Resultados de OneVsRest

Resultados

Encontramos que la mayoría de emociones positivas giran en torno a las emociones Love y Joy, si la empresa se quiere centrar en generar estas emociones debe apuntarle a entender los posts que están relacionados con las palabras clave de cada nube de palabras. Luego podemos decir que nuestro modelo óptimo es capaz de predecir estas frases en un 81% en el caso de love y en un 63% en el caso de joy.

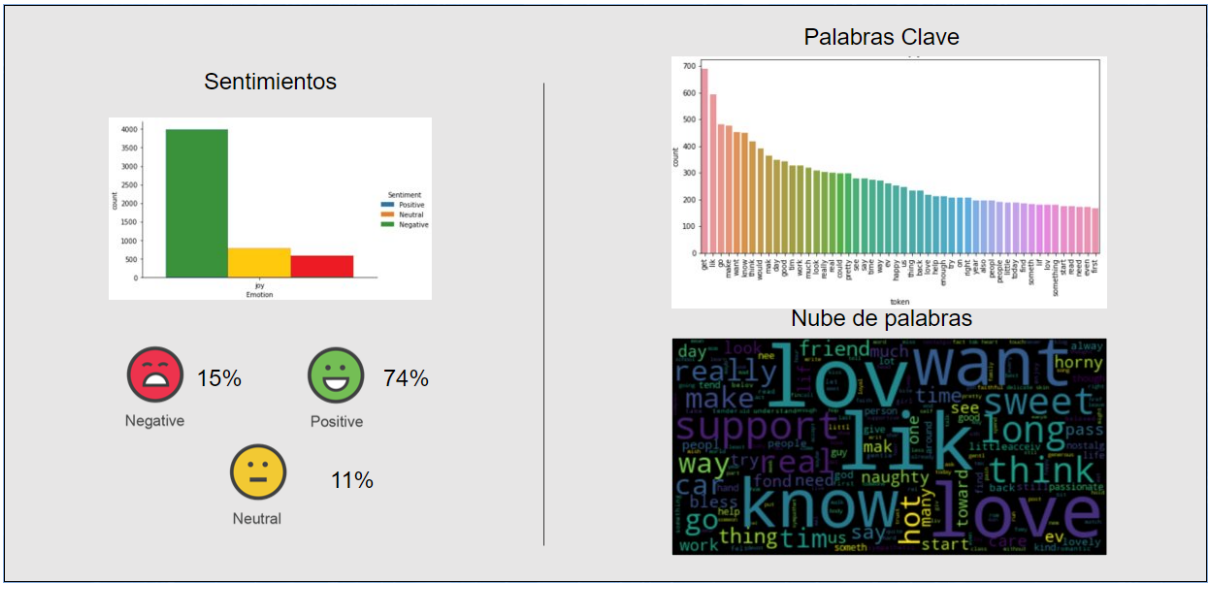

Figura 9: Tablero de control Joy

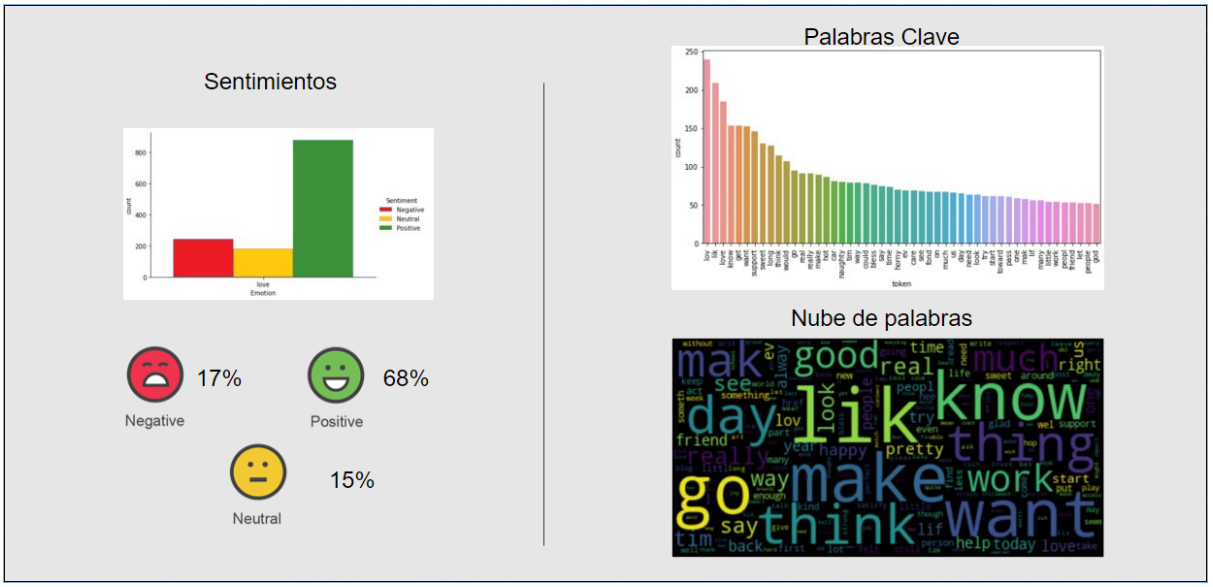

Figura 10: Tablero de control Love

Finalmente como recomendaciones para la empresa:

1) Esta puede usar nuestro modelo para percatarse de que una publicación es negativa aproximadamente el 80% del tiempo, por lo que puede tratar de ver que causa esas reacciones y actuar para solucionar este problema.
2) Un incremento en el número de palabras clave de love sería un buen indicador de que alguna publicación está teniendo éxito entre el público por lo que también es conveniente revisar estas publicaciones

Trabajo en Equipo

Ángela Jimenez:

Rol: Líder de proyecto
Algoritmo: Naive Bayes
Retos: En el preprocesamiento hubo que ajustar los datos con base a una decisión de negocio (quitar like y feel) esto conllevo un seguimiento constante para revisar que esta fuera una decisión acertada.
Tareas:

- Coordinar las reuniones grupales y planear el trabajo individual del grupo.
- Realizar el pre-procesamiento de los datos.
- Realizar su correspondiente algoritmo.
- Trabajo en el documento y presentación.

Andrés Rincón:

Rol: Líder de datos
Algoritmo: SVC
Retos: En un principio no estaba clara la diferencia entre análisis de sentimientos y análisis de emociones, por lo que tras un profundo análisis se consideró que era necesario realizar ambos análisis.

Tareas:

- Realizar el análisis de sentimientos en la etapa inicial.
- Realizar la representación de conteo de los datos en la etapa inicial
- Realizar su correspondiente algoritmo.
- Trabajo en el documento y presentación.

Miguel Acosta:

Rol: Líder de negocio y análitica
Algoritmo: OneVsRest
Retos: El entrenamiento de modelos en un inicio llevaba muchas pruebas y no teníamos un claro entendimiento de los pipelines por lo que no pudimos guardar nuestros modelos, posteriormente con la lectura de documentación logramos exportar los modelos.
Tareas:

- Representación en nube de palabras de los datos
- Realizar la evaluación de métricas de todos los algoritmos de forma consolidada para preparar la entrega de conclusiones al negocio.
- Realizar su correspondiente algoritmo.
- Trabajo en el documento y presentación.