Informe de Resultados

Proyecto Final PLN

Maestría en Ciencia de Datos

Equipo:

Mario Estrada Ferreira

María Janneth Rivera Reyna

Contexto del Proyecto

Estás trabajando en una empresa dedicada a la venta de alimentos por internet. Uno de los principales retos del negocio es garantizar la satisfacción de los clientes, atendiendo con prioridad a aquellos que han tenido experiencias negativas. Para ello, cuentas con un conjunto de reviews de usuarios, donde cada review incluye un texto y una calificación de 1 a 5 (1 es la calificación más baja, 5 la más alta).

Tu objetivo es desarrollar un sistema basado en técnicas de Procesamiento de Lenguaje Natural (NLP) y Aprendizaje Automático (ML) que permita:

Identificar automáticamente los reviews más negativos (calificaciones 1 y 2) para priorizar su atención.

Realizar un análisis de los temas predominantes en las opiniones negativas y responder: ¿Cuáles son las quejas más comunes?

Al final, se evaluará el desempeño de tu modelo con un conjunto de prueba, el cual les compartiré el domingo. Se medirá el porcentaje de mensajes negativos (calificaciones 1 y 2) que logró identificar correctamente tu modelo.

Análisis de sentimientos

Objetivo

El objetivo principal del proyecto es identificar automáticamente los reviews más negativos (calificaciones de 1 y 2) para priorizar su atención. Esto es clave para mejorar la satisfacción del cliente en una empresa de alimentos por internet. Adicionalmente, se busca analizar los temas predominantes en las opiniones negativas para responder a la pregunta: **¿Cuáles son las quejas más comunes?** .

El sistema desarrollado emplea técnicas de Procesamiento de Lenguaje Natural (NLP) y Aprendizaje Automático (ML) para clasificar automáticamente las reviews. Los resultados del modelo serán evaluados con un conjunto de prueba para medir qué tan bien identifica los mensajes negativos y si es capaz de generalizar en escenarios reales.

Contenido

El dataset contiene un total de 426,340 reviews, los cuales están agrupados bajo los siguientes scores:

Score	Count
5	272086
4	60490
1	39385
3	31972
2	22407

Tomando en cuenta la observación de este ejercicio, donde la clase negativa corresponde a los scores 1 y 2, podemos observar que el dataset se encuentra desbalanceado, ya que sólo el 14% de los datos correspondería a la clase negativa y el 86% restante a la clase no negativa (scores 3, 4, 5).

Este sería el principal desafío, ya que la clase de interés, en este caso la clase negativa, está subrepresentada. Aún si se tomara como clase no negativa sólo a los reviews con score 5, el problema del balance de clases persiste.

Es por esto que se toma como métrica principal para comparar entre modelos el F1 Score.

Modelo tipo BERT

Metodología

1.1 Preparación del Dataset

1. **Origen de los datos**:
    a. Reviews de clientes con calificaciones de 1 a 5.
    b. Labels binarios:
        i. "Negativo" (calificaciones 1 y 2).
        ii. "No Negativo" (calificaciones 3, 4 y 5).
2. **Creación de la muestra de datos:**
    a. Las clases se agruparon de la siguiente manera:
        i. Clase 1 (Negativos) = Score 1 y 2
        ii. Clase 0 (Positivos) = Scores 5
    b. Se agrego una nueva columna al dataset llamada Negative para agregar el valor 0 y 1 dependiendo de la agrupación.
    c. Se genero una muestra del 20% del dataset original, esto para ayudar al performance del entramiento y observando que una muestra de dicho tamaño sería suficiente.
    d. Se realizó un balance de clases, cargado levemente a los textos con etiqueta identificados negativo, esto para favorecer el objetivo del proyecto.
        i. Negativos: 51% de la muestra del 20% tomada del dataset orignal.
        ii. No negativos: 49% de la muestra del 20% tomada del dataset original,
3. **Preprocesamiento**:
    a. Limpieza de texto:
        i. Eliminación de caracteres HTML, puntuación y símbolos innecesarios utilizando expresiones regulares y `BeautifulSoup`.
    b. Conversión a minúsculas.
    c. Tokenización y eliminación de stopwords usando `nltk`.
        i. Se dejaron algunos stopwords basado en un análisis y pruebas realizadas, con el objetivo de favorecer al modelo en su interpretación, pues sin estas stopword muchos textos carecían de sentido gramatical y afectaban en el desempeño del modelo.

    d. La correcta aplicación de limpieza sobre los textos, ayudo al score del modelo, pasando de un 89% a llegar a un 96% ya en la evaluación final en las metricas globales.
4. **División del dataset de la muestra que se tomó (20% del original)**:
    a. Conjunto de entrenamiento: 80%.
    b. Validación: 10%.
    c. Prueba: 10%.
    d. Esta división asegura que el modelo aprenda de un conjunto variado y sea evaluado de forma justa.

1.2 Modelo Utilizado

El modelo seleccionado es **BERT (Bidirectional Encoder Representations from Transformers)**, específicamente una variante preentrenada de HuggingFace (albert-base-v2), debido a su capacidad para comprender el contexto y manejar datos textuales complejos, además que funcionó correctamente para tareas de clasificación.

Hiperparámetros Justificados

1. **Batch Size**:
    a. **Valor**: 16.
    b. **Justificación**: Este tamaño permite un uso eficiente de memoria en la GPU sin comprometer el rendimiento del entrenamiento. Además, valores más grandes podrían causar errores de memoria.
2. **Número de épocas**:
    a. **Valor**: 3.
    b. **Justificación**: Basado en experimentos previos, este número fue suficiente para alcanzar convergencia sin riesgo de sobreajuste. Más épocas podrían causar sobreentrenamiento en datasets medianos.
3. **Learning Rate**:
    a. **Optimizador**: AdamW (Weight Decay).
    b. **Tasa inicial**: 2e-5.
    c. **Justificación**: Este valor, comúnmente recomendado para modelos preentrenados, asegura un ajuste gradual y controlado. Se evita un aprendizaje abrupto que pueda perturbar los pesos iniciales de BERT.
4. **Scheduler**:
    a. **Warmup Steps**: 10% del total de pasos.

b. **Justificación**: Esto mejora la estabilidad al comienzo del entrenamiento, ayudando al modelo a ajustarse a los datos antes de aumentar la tasa de aprendizaje.

5. **Función de pérdida**:
    a. **CrossEntropy Loss**:
        i. Ideal para problemas de clasificación multiclase/binaria. Calcula la diferencia entre las predicciones del modelo y los valores verdaderos.

Configuración Técnica

- **Frameworks**:
    - HuggingFace Transformers.
    - PyTorch.
- **Tokenizador**:
    - `AutoTokenizer` de HuggingFace con subpalabras (WordPiece).
- **Arquitectura**:
    - BERT con una capa de clasificación adicional (capa densa de salida de tamaño 2).

1.3 Evaluación

1. **Métricas utilizadas**:
    a. **Precisión (Precision)**: Proporción de predicciones correctas sobre todas las predicciones positivas.
    b. **Recall (Sensibilidad)**: Capacidad del modelo para identificar todas las instancias verdaderamente negativas.
    c. **F1-Score**: Media armónica de precisión y recall, para balancear ambos aspectos.
    d. **Exactitud (Accuracy)**: Proporción de predicciones correctas totales.
    e. **Matriz de Confusión**: Visualización del rendimiento por clases.
    f. **Curva ROC-AUC**: Métrica que evalúa el rendimiento del modelo considerando todas las configuraciones posibles del umbral de decisión.
        i. El modelo alcanzó un **AUC de 0.98**, lo que indica una excelente capacidad para discriminar entre clases "Negativo" y "No Negativo".
        ii. Esto es crucial para el contexto, ya que asegura que el modelo prioriza adecuadamente las experiencias negativas que requieren atención inmediata.

Resultados del modelo:

- **Reporte en conjunto de prueba**:
  Precisión:
  - No Negativo: 97%
  - Negativo: 96%

  Recall:
  - No Negativo: 96%
  - Negativo: 97%

  F1-Score:
  - No Negativo: 96%
  - Negativo: 97%

  Accuracy global: 96%.

- **Matriz de Confusión**:
  - Falsos Positivos (FP): 679 casos (opiniones "No Negativas" clasificadas como negativas).
  - Falsos Negativos (FN): 521 casos (opiniones negativas clasificadas como "No Negativas").

Interpretación de Resultados

- El modelo logra un excelente desempeño en la tarea de clasificación binaria.
- La **alta precisión y recall en ambas clases** demuestran su capacidad para identificar correctamente los reviews negativos, mientras minimiza los errores de clasificación.
- Un F1-Score de 96-97% evidencia un balance sólido entre precisión y recall, fundamental para el contexto empresarial.

Validación con Datos Externos

- El modelo fue probado con un conjunto adicional de reviews y mantuvo una **exactitud del 96%**, validando su capacidad de generalización.

2. Relación con el Contexto del Proyecto

El modelo desarrollado responde perfectamente a las necesidades del proyecto:

1. **Identificación Automática de Reviews Negativos**:

    a. Gracias al desempeño del modelo, la empresa puede priorizar automáticamente aquellos clientes con experiencias negativas (calificaciones 1 y 2).
    b. Esto permitirá tomar acciones inmediatas para mejorar la satisfacción del cliente.
2. **Escalabilidad**:
    a. El enfoque basado en BERT es altamente adaptable y puede ajustarse a otros lenguajes o dominios si la empresa decide expandir sus operaciones.
3. **Análisis de Quejas Comunes**:
    a. Aunque no se implementó en esta fase, las representaciones generadas por BERT pueden utilizarse para realizar análisis de temas predominantes en reviews negativos. Esto ayudaría a identificar áreas críticas de mejora (e.g., calidad de alimentos, tiempos de entrega, servicio al cliente).

3. Conclusión

El modelo BERT entrenado ha demostrado ser una solución eficaz para la identificación de reviews negativos, con resultados robustos tanto en métricas de evaluación como en pruebas externas. Su implementación en la empresa podría mejorar significativamente la experiencia del cliente, proporcionando información valiosa para priorizar acciones correctivas.

Regresión Logística

Primeramente, las clases se agruparon de la siguiente forma:

- Clase 1 (Negativos) = Score 1 y 2
- Clase 0 (Positivos) = Scores 5

Se crearon los conjuntos de entrenamiento con el 80% de los datos y validación con el 20%, para después aplicarles los procesos de limpieza y preprocesamiento anteriormente descritos.

También se utilizó la técnica de balanceo de clases implementada en el modelo BERT, para tratar de mitigar este problema.

Además, se utilizó un modelo TF-IDF para la vectorización de los documentos, tomando el parámetro max_features = 2000.

Posteriormente, utilizando *RandomizedSearchCV* de *sklearn* se realizó una búsqueda de los mejores hiperparámetros del modelo *LogisticRegression* de *sklearn*. Se tomaron en cuenta los siguientes:

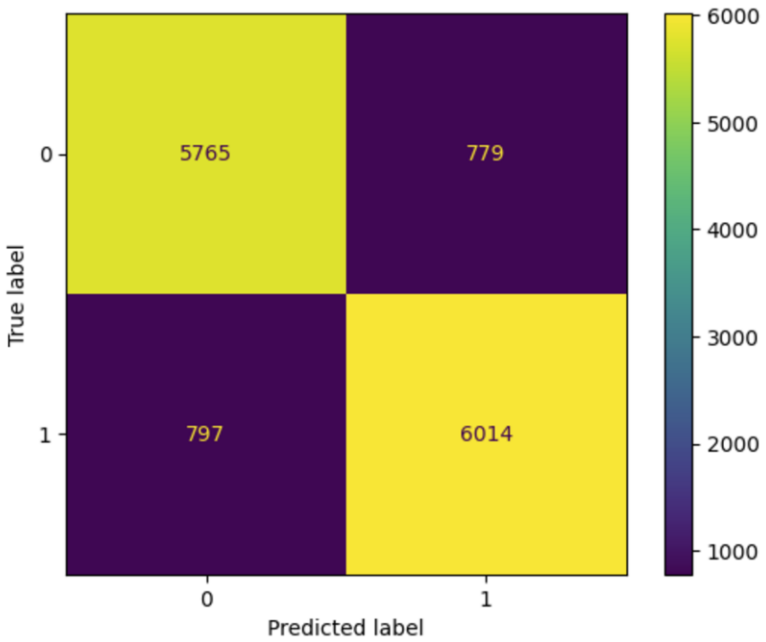
- C: parámetro de regularización
- Penalty: tipo de regularización
- Solver: algoritmo de optimización
- Max_iter: número de iteraciones para la convergencia

De esta manera se obtuvieron los siguientes resultados para el *ClassificationReport* y la matriz de confusión:

```
Accuracy: 0.8819917633845001
F1-score: 0.8841517200823287
              precision    recall  f1-score   support

           0       0.88      0.88      0.88      6544
           1       0.89      0.88      0.88      6811

    accuracy                           0.88     13355
   macro avg       0.88      0.88      0.88     13355
weighted avg       0.88      0.88      0.88     13355
```



Con este modelo obtenemos un **F1 score del 88%**, el cual difiere sólo de un 8% del resultado obtenido con el modelo tipo BERT (96%). La gran diferencia es que este modelo

no requiere de tanto poder de cómputo, por lo tanto, consideramos que es una buena opción cuando existen limitantes en este aspecto.

Topic Modelling

Análisis

¿Cuáles son las quejas más comunes de los consumidores?

Para responder esta pregunta, se utilizó el modelo BERTopic, con los siguientes parámetros:

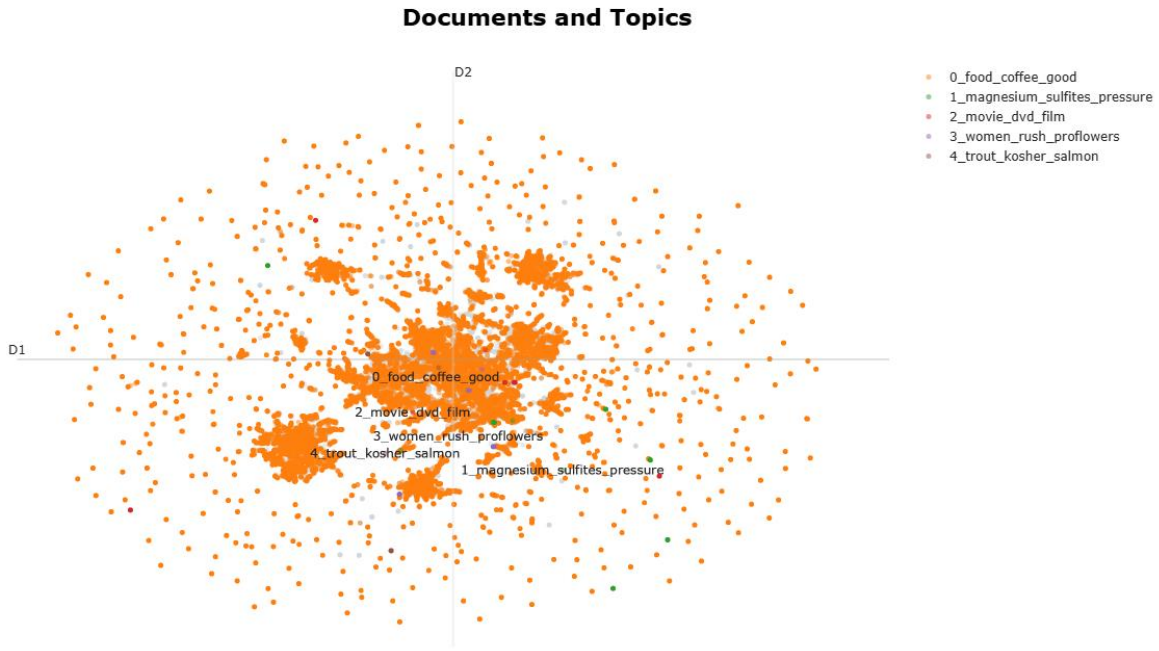
```
top_n_words=10, # Número de palabras más representativas a incluir en cada
            tópico
nr_topics=num_topics,# Número de tópicos a crear
min_topic_size=5, # Número mínimo de documentos necesarios para formar un
            tópico
```

Primeramente, se realizó el ejercicio de quitar solo las stopwords default, y se observó que tanto el tópico -1 (outliers) como el tópico 1 concentraba palabras como **"n't" "product", "taste", "like", "would", "flavor", "one"**, y estas aparecían repetidamente en todos los tópicos, lo cual nos habla de cómo la mayoría de las quejas van en torno a si los productos son o no del agrado de los usuarios.

Posteriormente, se agregaron estas palabras a la lista de stopwords y se removieron. Esto nos permitió ver qué tipos de productos se agrupaban por tópicos.

Otra palabra que aparece frecuentemente en los tópicos es **"amazon"** por lo cual podríamos deducir de donde provienen estos reviews.

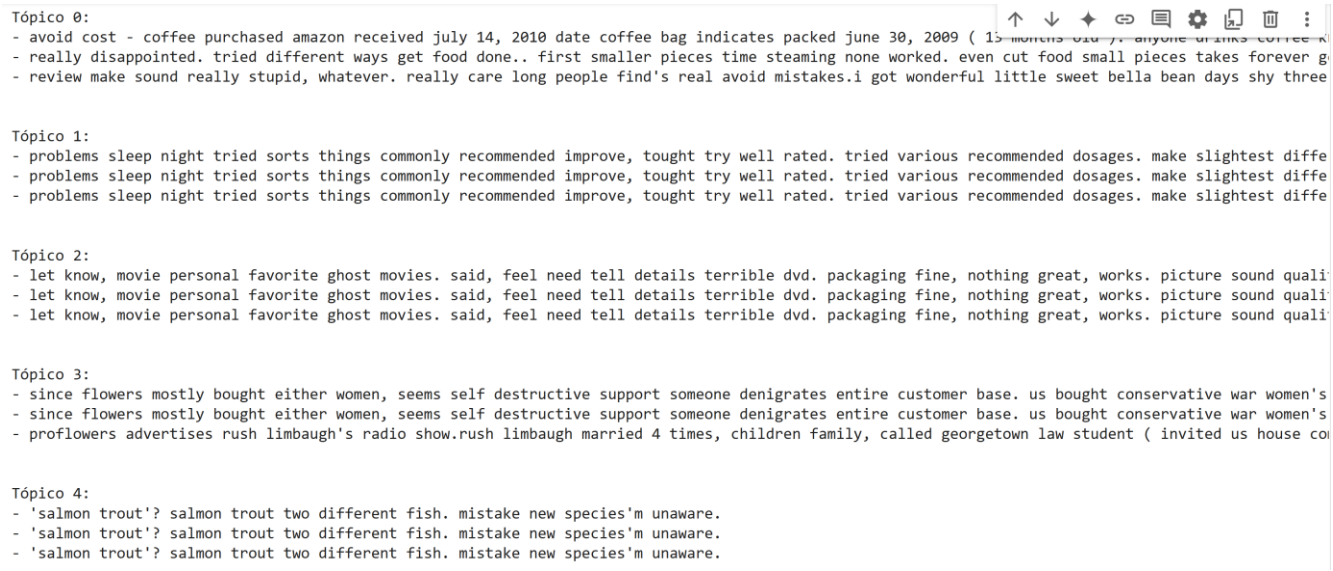
Los resultados visuales del agrupamiento son los siguientes:



Analizando cada uno de los tópicos, podemos ver que los documentos están relacionados con temas como:

```
Topic 0: ['food', 'coffee', 'good', 'even', 'get', 'buy', 'really', 'much', 'tea', 'amazon']
Topic 1: ['magnesium', 'sulfites', 'pressure', 'diet', 'cysts', 'calcium', 'cocaine', 'blood', 'stream', 'absorption']
Topic 2: ['movie', 'dvd', 'film', 'latelyd', 'badso', 'extras', 'release', 'tim', 'beetlejuice', 'owners']
Topic 3: ['women', 'rush', 'proflowers', 'limbaugh', 'lace', 'support', 'loserthis', 'houtts', 'gloves', 'size']
Topic 4: ['trout', 'kosher', 'salmon', 'speciesm', 'unaware', 'certified', 'fish', 'two', 'different', 'mistake']
```

En los documentos asociados a cada tópico, podemos ver algunos ejemplos:



Podemos ver que el **Topic 0** es grande y trata temas relacionados a café, té, y otros alimentos incluyendo de mascotas, con quejas sobre si el producto era de calidad, si estaba viejo, o incluso si es saludable.

En el **Topic 1**, trata temas relacionados a problemas del sueño y el uso de magnesio.

En el **Topic 2**, habla sobre películas y DVDs (lo cual nos hace pensar que los reviews no son muy recientes)

En el **Topic 3**, menciona una empresa dedicada a la venta de flores (ProFlowers) y su relación con Rush Limbaugh, alguien que al parecer ha hecho comentarios ofensivos hacia las mujeres.

Por último, el **Topic 4** parece tratar sobre diferentes peces como el salmón y la trucha.

Conclusiones

A pesar de que algunos temas parecen discriminarse bastante bien del resto, y hay otros donde no es tan clara la separación. El ejercicio de dejar que BERTopic decidiera el número de tópicos causaba que se creara un tópico para cada producto, obteniendo más de 1000 tópicos.

Es posible que la manera en que se reduce la dimensionalidad también esté afectando. También valdría la pena analizar con otros modelos de clustering.