



**UNIVERSIDAD DE
CÓRDOBA**



LICENCIATURA EN 
INFORMÁTICA

INTRODUCCIÓN A LA CIENCIA DE DATOS

Integrantes:

Kevin Andrés Nisperuza Sierra

Daniel Andrés Sánchez Mendez

Docente:

Alexander Toscano Ricardo

Universidad de Córdoba
Facultad de Educación y Ciencias Humanas
Licenciatura en Informática
2024

Introducción

El análisis de sentimientos es una técnica de procesamiento del lenguaje natural (NLP) que permite identificar si un texto expresa una opinión positiva, negativa o neutral. En el contexto educativo, esta técnica puede aplicarse para evaluar encuestas de satisfacción, identificar áreas de mejora y comprender mejor la experiencia de los estudiantes.

1. Objetivo del Proyecto

El objetivo de este proyecto es analizar los sentimientos de los comentarios de los estudiantes obtenidos a través de encuestas de satisfacción, con el fin de identificar tendencias positivas o negativas en la retroalimentación.

2. Descripción del Proyecto

Bibliotecas y Herramientas

- **Hugging Face Transformers:** Utilizamos el modelo pre entrenado DistilBERT ajustado para análisis de sentimientos.
- **Pandas:** Para la manipulación y visualización de datos.
- **Matplotlib:** Para la creación de gráficos y análisis visual de resultados.

Modelo Seleccionado

- **DistilBERT fine-tuned para análisis de sentimientos:**
 - **Modelo:** [distilbert-base-uncased-finetuned-sst-2-english](#)
 - **Propósito:** Análisis de sentimientos, específicamente entrenado para detectar sentimientos positivos y negativos.

3. Datos Utilizados

Se utilizaron comentarios de estudiantes simulados para este proyecto. Los comentarios expresan la opinión de los estudiantes sobre diferentes aspectos de un curso, tales como la metodología de enseñanza, materiales y actividades prácticas.

Ejemplos de comentarios:

1. "El curso fue increíble, aprendí mucho."
2. "No me gustó la metodología de enseñanza, fue muy confusa."

3. "Los materiales estaban bien, pero el profesor no fue muy claro."
4. "Me encantaron las actividades prácticas."
5. "Hubo algunos temas que no se explicaron bien."

4. Arquitectura del Sistema

1. **Entrada de Datos:**
 - Comentarios de los estudiantes obtenidos a través de encuestas de satisfacción.
 - Los datos se almacenan en un archivo CSV y se cargan en un DataFrame de Pandas.
2. **Análisis de Sentimientos:**
 - Aplicación del modelo DistilBERT para clasificar cada comentario como positivo o negativo.
 - Los resultados se agregan al conjunto de datos original como una columna adicional.
3. **Visualización de Resultados:**
 - Generación de gráficos con Matplotlib para observar tendencias en los datos analizados.
4. **Resultados y Almacenamiento:**
 - Los datos procesados se guardan en un archivo CSV para futuras consultas y análisis.

Instalación y Configuración

1. **Clonar el repositorio:**

```
git clone <URL_DEL_REPOSITORIO>
cd <NOMBRE_DEL_PROYECTO>
```

2. **Crear un entorno virtual:**

```
python -m venv env
source env/bin/activate # En Windows: env\Scripts\activate
```

3. **Instalar las dependencias:**

```
pip install -r requirements.txt
```

4. Configurar el proyecto:

- Verificar que los datos de entrada estén disponibles en la carpeta designada.
- Configurar los parámetros necesarios en el archivo de configuración (si aplica).

Explicación Técnica del Código

Carga de Datos

El proyecto utiliza **Pandas** para cargar los datos de los comentarios desde un archivo CSV:

```
import pandas as pd

# Cargar datos desde un archivo CSV
comments_df = pd.read_csv('student_comments.csv')
```

Aplicación del Modelo

El modelo **DistilBERT** ajustado para análisis de sentimientos se utiliza para procesar los comentarios:

```
from transformers import pipeline

# Cargar el modelo de análisis de sentimientos
sentiment_analer = pipeline('sentiment-analysis', model='distill

# Aplicar el modelo a cada comentario
comments_df['sentiment'] = comments_df['comment'].apply(lambda x: 
```

```
ntos
lysis', model='distilbert-base-uncased-finetuned-sst-2-english')

mment'].apply(lambda x: sentiment_analyzer(x)[0]['label']))yz
```

Visualización de Resultados

Se emplea **Matplotlib** para crear gráficos que visualicen las tendencias:

```
import matplotlib.pyplot as plt

# Contar sentimientos positivos y negativos
sentiment_counts = comments_df['sentiment'].value_counts()

# Crear gráfico de barras
sentiment_counts.plot(kind='bar', title='Distribución de Sentim:
plt.xlabel('Sentimiento')
plt.ylabel('Número de Comentarios')
plt.show()
```

```
matplotlib.pyplot as plt

r sentimientos positivos y negativos
nt_counts = comments_df['sentiment'].value_counts()

gráfico de barras
nt_counts.plot(kind='bar', title='Distribución de Sentimientos')
bel('Sentimiento')
bel('Número de Comentarios')
w()
```

Resultados y Almacenamiento

Finalmente, los datos procesados se guardan en un archivo CSV:

```
# Guardar el DataFrame actualizado
comments_df.to_csv('processed_comments.csv', index=False)
```

Con esta estructura, el proyecto permite analizar de manera efectiva los comentarios de los estudiantes, identificando tendencias clave en su retroalimentación y proporcionando una base para mejoras en los cursos evaluados.

https://github.com/Grupo-Investigacion-Bimadino/paul_poll.git

