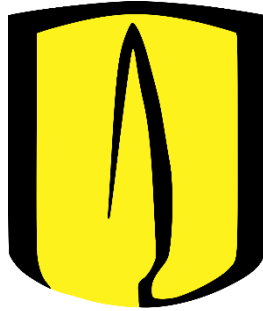


# INFORME PROYECTO 1 – PARTE 2 – NLP



Juan Esteban Rodríguez Ospino  
Wilton Esteban Martínez Hernández  
Erich Gusseppe Soto Parada

Inteligencia de Negocios  
Ingeniería de sistemas y Computación  
Universidad de Los Andes  
2023

## Contenido

Proceso de automatización del proceso de preparación de datos, construcción del modelo, persistencia del modelo y acceso por medio de API: .....	3
Correcciones de anterior etapa (sugerencias de experto en estadística) .....	3
Desarrollo de la aplicación y justificación .....	5
Resultados. ....	8
Trabajo en equipo. ....	9
Roles .....	9
Reuniones .....	10

Proceso de automatización del proceso de preparación de datos, construcción del modelo, persistencia del modelo y acceso por medio de API:

Correcciones de anterior etapa (sugerencias de experto en estadística)

En pro de escoger el mejor modelo de *machine learning*, el cual sería usado en nuestra aplicación, se determinó plantear unas pruebas por hipótesis. Lo anterior, como sugerencia de la persona de estadística que nos asesoró. A las hipótesis se les asignó un valor de significancia. Más adelante en el apartado de resultados podremos apreciar cuál de las dos se comprueba para el caso de nuestro proyecto.

### **Proponer pruebas de hipótesis:**

**Hipótesis 1:** El modelo de clasificación de textos SVM (Support Vector Machine) es la mejor opción para realizar el análisis de sentimientos de películas en español, ya que ha demostrado un alto rendimiento en problemas de clasificación de texto y es capaz de manejar grandes conjuntos de datos con alta dimensionalidad.

**Justificación:** El modelo SVM es una técnica de aprendizaje automático que ha demostrado un alto rendimiento en problemas de clasificación de texto, especialmente en tareas de análisis de sentimientos. Además, es capaz de manejar grandes conjuntos de datos con alta dimensionalidad, lo que es una ventaja importante en este proyecto, ya que se cuenta con una gran cantidad de comentarios de películas en español. Por lo tanto, se espera que el modelo SVM sea capaz de obtener una alta precisión en la clasificación de comentarios de películas en positivo o negativo.

**Hipótesis:** El modelo Random Forest implementado en el proyecto de análisis de sentimientos de películas logrará una precisión del 85% en la clasificación de comentarios de películas en español como positivos o negativos.

**Justificación:** El modelo Random Forest fue seleccionado debido a que permite la clasificación en diferentes categorías y es capaz de identificar patrones y relaciones en grandes conjuntos de datos, lo que es especialmente útil en la clasificación de comentarios de películas. Además, se utilizó la técnica de Bag of Words y la técnica TF-IDF para vectorizar las palabras de los comentarios, y se implementó la técnica de validación cruzada para evitar el overfitting y mejorar la generalización del modelo. Se espera que con estas técnicas y el uso del modelo Random Forest, se logre una precisión del 85% en la clasificación de comentarios como positivos o negativos.

### **Significancia**

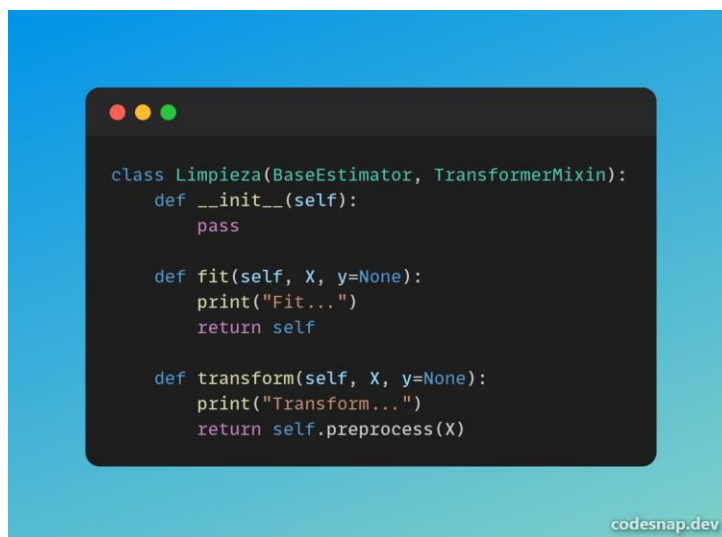
En el caso de la hipótesis sobre el modelo SVM, se podría justificar un nivel de significancia de 0.05 ya que la elección del mejor modelo es crucial para el éxito del análisis de

sentimientos de películas en español y se espera que el modelo SVM obtenga una alta precisión.

En el caso de la hipótesis sobre el modelo Random Forest, también se podría justificar un nivel de significancia de 0.05, ya que se espera que este modelo tenga un buen desempeño en la clasificación de comentarios de películas y es importante elegir el mejor modelo para obtener resultados precisos y confiables.

## Pipeline

Para automatización de los procesos hechos al dataset de datos se implementó un Pipeline. En este se realizan transformaciones sobre los datos, las cuales incluyen: la eliminación de caracteres no alfanuméricos, la eliminación de las stopwords de inglés y español, etc. Esto se realiza desde la clase limpieza la cual está implementando las interfaces BaseEstimator y TransformerMixin.



```
class Limpieza(BaseEstimator, TransformerMixin):
    def __init__(self):
        pass

    def fit(self, X, y=None):
        print("Fit...")
        return self

    def transform(self, X, y=None):
        print("Transform...")
        return self.preprocess(X)
```

codesnap.dev

Aquí se invoca la clase Limpieza para automatizar el proceso de limpieza de datos, luego aplicamos el método CountVectorizer a las palabras para posteriormente aplicar el modelo.



```
1 # Crear el pipeline
2 text_pipeline = Pipeline([
3     ('preprocessing', Limpieza()),
4     ('vectorizer', CountVectorizer(tokenizer=nlTK.TweetTokenizer().tokenize, stop_words=stop_words_complete, lowercase=True)),
5     ('classifier', RandomForestClassifier(random_state=2, n_estimators=200, min_samples_split=4, max_depth=3, criterion='entropy'))
6 ])
7
```

Para hacer posible la conexión entre el modelo de aprendizaje automático y la aplicación web, se creó una clase llamada “**Model\_prediction**”, que es la encargada de cargar el modelo de aprendizaje automático y hacer las predicciones. Además, se creó otra clase llamada “**Comentario**”, que se utiliza para convertir el comentario ingresado en el formulario en un DataFrame de Pandas que puede ser procesado por el modelo de aprendizaje automático.

Cabe aclarar que nuestro modelo de análisis de sentimientos se encuentra integrado en un Pipeline que nos permite encadenar varias etapas de procesamiento de datos. El modelo seleccionado para llevar a cabo la tarea de clasificación de textos es el SVM (Support Vector Machine), el cual ha sido entrenado con un conjunto de datos etiquetados previamente.

**Más adelante justificaremos el porqué de la elección de este modelo, frente a los otros dos presentados en la anterior etapa de este proyecto. Sin embargo, de antemano podemos decir que unas de las ventajas de este modelo** que ha demostrado un alto rendimiento en problemas de clasificación de texto, especialmente en tareas de análisis de sentimientos.

### Desarrollo de la aplicación y justificación

Este proyecto surge con la intención de crear una herramienta que permita analizar de manera rápida y precisa los comentarios de los usuarios sobre películas en español. De esta manera, con la intención de dar solución a los objetivos del proyecto hemos desarrollado una aplicación web que el proceso de negocio de análisis de sentimientos de comentarios de películas en español por medio un modelo de aprendizaje automático. Lo anterior, permite una mayor eficiencia en la toma de decisiones por parte de las empresas que desean conocer la opinión general de su público objetivo. De igual manera, es importante destacar que la existencia de esta aplicación es crucial para el rol del analista de datos, en el entorno de una organización que se dedican a la reproducción de películas, pues brinda la posibilidad a este de centrarse en otros aspectos importantes del análisis de datos. En esta el analista de datos es responsable de realizar el análisis de sentimientos de comentarios de películas en español y a partir de lo anterior, apoyar a las diferentes áreas de la empresa marketing y ventas, producción y servicio al cliente. Finalmente, es importante mencionar que para el desarrollo de la aplicación se llevo a cabo el supuesto de que los **comentarios tienen asociada una película**. Esto se hizo porque permite que las predicciones del modelo tengan unos Insight más significativos. Asimismo, es importante mencionar que agregar este atributo no afecta el modelo, pues este solo toma el comentario de la película.

Por otra parte, para el desarrollo de la aplicación web se utilizó el framework Flask del lenguaje de programación Python. En este se utilizaron diferentes librerías de Python para las diferentes implementaciones que permiten el funcionamiento de la aplicación. Por otra parte, para la sección visual de la aplicación se usó **HTML** para estructurar el contenido visual, **Bootstrap 5** para dar estilo de una manera más fácil al contenido y **Chartjs** para la creación de gráficas. Por último, se hace uso de librerías como **Pandas** para el manejo de datos y, **Joblib** y **Pkl** para cargar el modelo de aprendizaje automático.

En cuanto al funcionamiento de la aplicación esta funciona de la siguiente manera:

En primer lugar, el usuario debe ser registrado y posteriormente se loguearse, así garantizamos la privacidad y seguridad de los datos.

## Bienvenido

Regístrame

Iniciar sesión

Seguido a esto el usuario tiene dos opciones:

1. Ingresar un comentario en un formulario para predecir manualmente cada uno de los comentarios.
2. Cargar un dataset con muchos comentarios y generar predicciones a cada uno de los comentarios del conjunto de datos.

Snoopers movies Inicio Ingresar comentario Cargar documento Dashboard

Salir

Ingresa un comentario para  
que los analice el software

Carga archivos con  
comentarios

En este la aplicación muestra el resultado de inmediato.

Snoopers movies Inicio Ingresar comentario Cargar documento Dashboard

Salir

Aquí puedes escribir un comentario para que el sistema lo analice

Nombre película

El indio

Escribe tu opinión...

Esta es una de las mejores películas para mascotas para niños.Lloro cada vez que veo la sombra gritando "¡Espera, espere a Peter!"A medida que el coche familiar se está alejando,¡Esto es una visita obligada si amas a los animales!La mejor película de la historia!Las líneas en la película son a veces estúpidas.Como cuando Sassy dice al azar;"¡Regla de gato y perros babeal!"Líneas como esta puede hacer sin, pero cuando tenía seis años, me encantó esa línea.La historia puede parecer gancho para algunos, pero me gusta.Shadow como el perro mayor que está preparando la oportunidad de hacerse cargo de él cuando se ha ido realmente se está moviendo cuando lo piensas.Me recordó a mi perro infantil.Creo que todos pueden

Enviar

*El análisis dice que el  
comentario es:  
positivo*

Para esta opción la aplicación crear un dashboard de las predicciones.

Snoopers movies

Inicio

Ingresar comentario

Cargar documento

Dashboard

Salir

Aquí puedes cargar un archivo con comentarios para que el sistema los analice

Escribe tu opinión...

Choose File

No file chosen

Enviar

También es posible que el usuario pueda visualizar las predicciones de los comentarios:

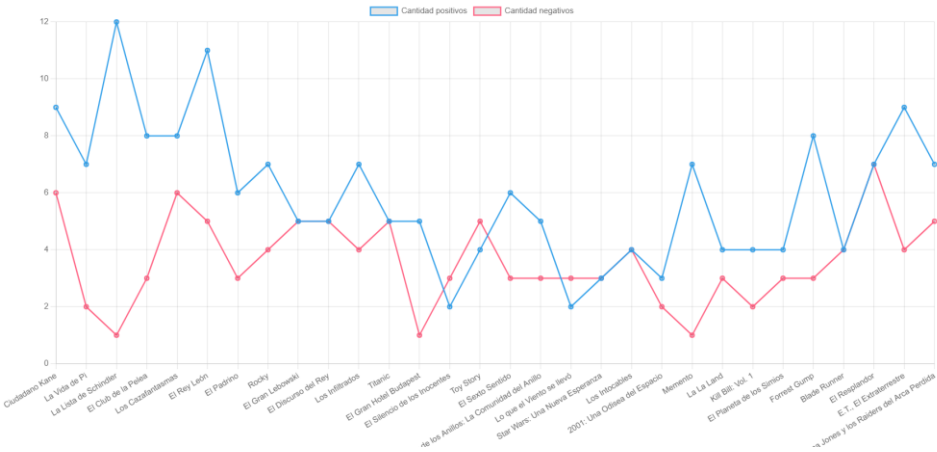
### Estadísticas de los comentarios

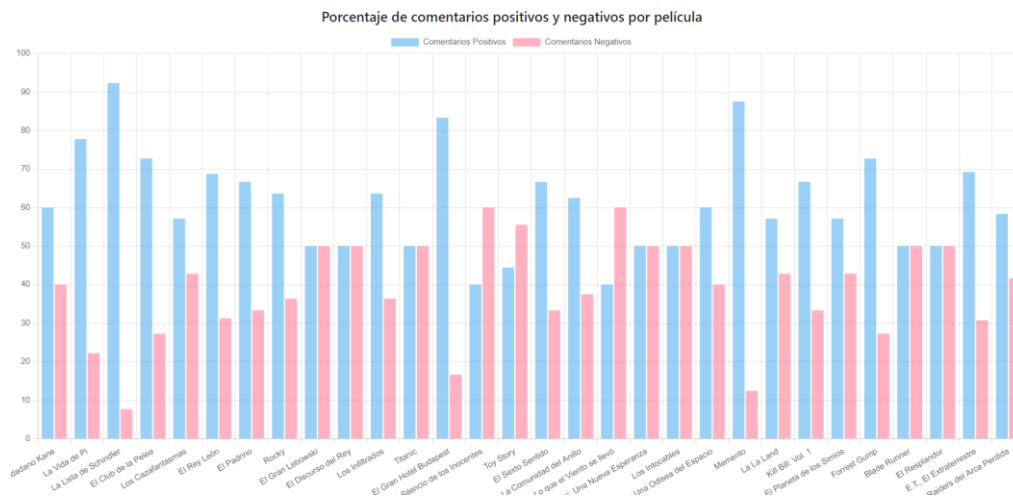
Cantidad comentarios por película

Nombre	Cantidad
Ciudadano Kane	15
La Vida de Pi	9
La Lista de Schindler	13
El Club de la Pelea	11
Los Cazafantasmas	14
El Rey León	16
El Padrino	9
Rocky	11
El Gran Lebowski	10
El Discurso del Rey	10
Los Infiltrados	11
Titanic	10
El Gran Hotel Budapest	6

Cantidad de comentarios negativos y positivos por película

Nombre	Positivos	Negativos
Ciudadano Kane	9	6
La Vida de Pi	7	2
La Lista de Schindler	12	1
El Club de la Pelea	8	3
Los Cazafantasmas	8	6
El Rey León	11	5
El Padrino	6	3
Rocky	7	4
El Gran Lebowski	5	5
El Discurso del Rey	5	5
Los Infiltrados	7	4
Titanic	5	5
El Gran Hotel Budapest	5	1





## Justificación de las funcionalidades

Se implementó el apartado de autenticación para el proyecto con el objetivo de asegurar que solamente los usuarios autorizados tengan acceso a la funcionalidad de la aplicación. Esto es especialmente importante si el modelo de Machine Learning contiene información sensible o privada que no debe ser vista por usuarios no autorizados. Además, al requerir que los usuarios se autenticquen antes de acceder a la aplicación, podemos rastrear mejor quién está utilizando la aplicación y cómo se está utilizando, lo que puede ser valioso para fines de análisis y mejora de la aplicación en el futuro.

De otro lado, se decidió el usuario tenga dos maneras de utilizar el modelo. La primera ingresando un comentario, pues dadas ciertas circunstancias el analista de datos solo deberá revisar comentarios específicos. La carga de conjuntos de comentarios se hizo con la intención de optimizar el trabajo, pues esto ayuda a agilizar el apoyo a otras áreas y así obtener unos resultados más concisos. Por último, la inclusión del dashboard es de su importancia, ya que le permite al analista de datos sacar conclusiones más significativas como, por ejemplo: ver que top de películas tiene más comentarios negativos y positivos o ver el porcentaje de comentarios positivos y negativos de cada película, etc.

## Resultados.

En conclusión, la hipótesis planteada afirma que el modelo Random Forest Classifier es la mejor opción para realizar el análisis de sentimientos de películas en español. La justificación se basa en que el modelo Random Forest Classifier ha demostrado un alto rendimiento en problemas de clasificación de texto y es capaz de manejar grandes conjuntos de datos con alta dimensionalidad. Además, se espera que este modelo sea capaz de obtener una alta precisión en la clasificación de comentarios de películas en positivo o negativo.



El modelo de clasificación de sentimientos de películas en español desarrollado en este proyecto utilizando el algoritmo de **Random Forest Classifier** fue capaz de alcanzar una precisión promedio de 0.785 en el conjunto de entrenamiento y 0.731 en el conjunto de prueba. Además, se obtuvo un **recall** promedio de 0.921 en el conjunto de entrenamiento y 0.877 en el conjunto de prueba, y un **F1** promedio de 0.847 en el conjunto de entrenamiento y 0.797 en el conjunto de prueba.

	Train	Test
Precision	0.801038	0.738056
Recall	0.926000	0.896000
F1	0.858998	0.809395

Es importante destacar que se observó un cierto grado de sobreajuste (overfitting) del modelo, ya que los valores de **precisión**, **recall** y **F1** en el conjunto de entrenamiento son significativamente mejores que los obtenidos en el conjunto de prueba. Esto indica que el modelo se ajustó demasiado a los datos de entrenamiento y no generalizó de manera óptima para nuevos datos. A pesar de este sobreajuste, los resultados del modelo de Random Forest Classifier son muy prometedores y sugieren que este algoritmo es una opción viable para el análisis de sentimientos de películas en español.

Trabajo en equipo.

Roles

### Roles del equipo

**Líder de proyecto y Ingeniero de software responsable de desarrollar la aplicación**

**final:** Wilton Esteban Martinez Hernandez

**Tiempo:** 12 horas

**Tarea:** Implementación de la aplicación web

**Retos del proyecto y como resolverlos:** un gran reto fue conectar el modelo persistido en el archivo **joblib** con la aplicación web. En este se tuvieron varios problemas a la hora utilizar el modelo con los datos de entrada, pues no se encontraba un módulo que se utilizo para la construcción del Pipeline. El problema se resolvió creando un nuevo proyecto desde cero y ejecutando de nuevos los archivos para la construcción del Pipeline.

**Puntos para mejorar:** Creo que para este proyecto fallo un poco la comunicación, en algunas asignaciones de tareas la comunicación no fue clara, sin embargo, lo anterior no causo grandes problemas.

**Ingeniero de software responsable del diseño de la aplicación y resultados:** Juan Esteban Rodríguez Ospino

**Tiempo en horas:** 6h

**Tareas:** Realización documento, login de aplicación, hipótesis y conclusiones

**Retos del proyecto y como resolverlos:** Crear una propuesta que sea concluyente y beneficiosa para una empresa o usuario en cuestión. Es importante eliminar la generalidad presentada la anterior etapa del proyecto. Analizar y poner en práctica las recomendaciones hechas por la especialista en estadística.

**Otro reto es mantener el modelo actualizado,**

**Puntos para mejorar:** Plantear la hipótesis y hipótesis alterna desde la primera etapa del proyecto, de este modo las conclusiones y resultados tendrán una mayor relevancia a la hora de ser presentadas al usuario final.

**Ingeniero de datos:** Erich Giuseppe Soto Parada

**Tiempo en horas:** 7h

**Tareas:** construcción de Pipeline

**Retos del proyecto y como resolverlos:** de alguna manera la limpieza que aplicamos a los datos en la parte 1 a la hora de integrarlos a pipeline no permitieron que las métricas de error fueran iguales. Para dar solución se probaron de diferentes formas para dar unas métricas más aproximadas a la entrega 1.

**Puntos para mejorar:** Se podría tal vez corrido el modelo con más combinaciones (para lo que se requeriría unas 6 horas más de entrenamientos) y tener los datos de mejor calidad y con una limpieza de datos mejor, pero de resto está bien.

## Reuniones

**Primera reunión:** En esta reunión se plantearon las diferentes responsabilidades que conllevaría cada integrante del grupo, además se planteó el modo en el que nos comunicaríamos con la persona de especialista en estadística que nos ayudaría a hacer un mejor análisis de nuestro proyecto.

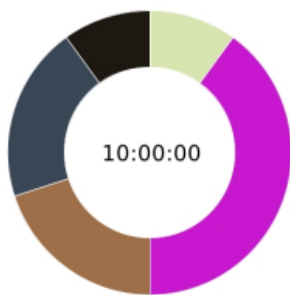
**Reunión con especialista estadística:** En esta nos reunimos con Valery Fonseca podemos destacar 2 diferentes apartados de esta reunión.

- 1- Se nos sugirió que el análisis de nuestro proyecto debía ser mucho más concluyente. Para esto se nos explico e indicó que la mejor manera es haciendo una prueba por hipótesis desde la cual pudiéramos ver cual de los modelos se adapta y responde mejor a las circunstancias específicas de nuestro proyecto.
- 2- En términos de roles para los usuarios finales pudimos concluir que, dado al carácter y finalidad del proyecto, el verdadero usuario final y a quien en mayor parte beneficiaría este aplicativo sería al analista de datos. Lo anterior debido a que, desde su posición, gracias a la aplicación que hemos desarrollado, podrá automatizar, analizar y tomar decisiones sobre los datos y comentarios de las películas. De este modo, poder crear acciones de mejoras de procesos y trabajo en la empresa, aumentando las rentabilidades.

**Tercera reunión:** Esta reunión fue clave en pro de analizar cómo iba el desarrollo de las diferentes tareas y que hacía falta para terminar el desarrollo de la aplicación.

## Distribución de tiempo y tareas

### Description



● Reunion Estadística	01:00:00	10,00%
● Documento	02:00:00	20,00%
● Pruebas en modelo	02:00:00	20,00%
● Desarrollo de Api-Rest	04:00:00	40,00%
● Login y Autenticación	01:00:00	10,00%

## Summary report

01/04/2023 - 30/04/2023



Total: 10:00:00 Billable: 10:00:00 Amount: 0,00 USD

