*Imagen que contiene Logotipo

Descripción generada automáticamente*Icono

Descripción generada automáticamente*Logotipo, nombre de la empresa

Descripción generada automáticamenteIcono

Descripción generada automáticamente*

**Institución:** Tecnológico de Culiacán.

**Carrera:** Ingeniería en sistemas computaciones.

**Materia:** Inteligencia Artificial 11:00 – 12:00.

**Maestro:** Zuriel Dathan Mora Felix.

**Tema:** Tarea 3

**Integrantes:**

Samano Reyes Jordan Nayar

Cervantes Araujo Carlos Iván.

índice

[**Introducción** 3](#_Toc194240852)

[**Librerias** 4](#_Toc194240853)

[**Preprocesamiento** 4](#_Toc194240854)

[**Teorema de Bayes** 5](#_Toc194240855)

[**Función “reporteF”** 5](#_Toc194240856)

[**Ejemplo de Salida:** 6](#_Toc194240857)

[**Flujo Principal** 6](#_Toc194240858)

[**Instrucciones de Uso** 7](#_Toc194240859)

# **Introducción**

En este documento se hará una documentación sobre un código el cual implementa un clasificador de spam utilizando el algoritmo Naive Bayes. Procesa textos (emails), calcula probabilidades condicionales y genera un reporte de rendimiento en español.

# **Librerias**

Lo primero que vemos en el código son las dependencias implementadas en el programa:

Texto

El contenido generado por IA puede ser incorrecto.

# **Preprocesamiento**

A continuación se crea la clase Preprocesamiento, esto con el objetivo de:

Inicializar el objeto cargando las stopwords en inglés desde NLTK y procesarlos con los siguientes pasos:

1. **Normalización**: Convierte texto a minúsculas.
2. **Limpieza**: Elimina puntuación usando regex [^\w\s].
3. **Tokenización**: Divide el texto en palabras con word\_tokenize.
4. **Filtrado**: Remueve stopwords y palabras con ≤2 caracteres.

Texto

El contenido generado por IA puede ser incorrecto.

# **Teorema de Bayes**

**Propósito:**

Implementa el clasificador de spam basado en el Teorema de Bayes.

**Atributos:**

* P\_spam: Probabilidad a priori de que un mensaje sea spam.
* P\_ham: Probabilidad a priori de no spam.
* P\_palabra\_spam: Diccionario con P (palabra | spam).
* P\_palabra\_ham: Diccionario con P (palabra | ham).

**Métodos:**

* **entrenar (correos, etiquetas)**:
  + Calcula probabilidades a priori y condicionales con suavizado de Laplace.
  + **Ecuación de Suavizado**:

P(palabra|clase) = (frec\_palabra\_en\_clase + 1) / (total\_palabras\_en\_clase + tamaño\_vocabulario)

Texto, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

* **predecir(correo)**:
  + Utiliza log-probabilidades para evitar underflow numérico.
  + Decide la clase comparando log(P\_spam) + Σ log(P(palabra|spam)) vs log(P\_ham) + Σ log(P(palabra|ham)).

# **Función “reporteF”**

**Propósito:**

Muestra un reporte de clasificación en formato tabular en español, incluyendo:

* Precisión, Sensibilidad (Recall), F1-Score y Soporte para cada clase.
* Métricas globales: Exactitud (Accuracy) y promedios.

# **Ejemplo de Salida:**

==================================================

CLASIFICACIÓN

==================================================

Precisión Sensibilidad F1-score Soporte

--------------------------------------------------

NO SPAM 0.98 0.99 0.98 946

SPAM 0.96 0.89 0.92 159

--------------------------------------------------

Promedio macro 0.97 0.94 0.95 1105

Promedio ponderado 0.97 0.97 0.97 1105

==================================================

Exactitud global del modelo: 0.9745

==================================================

# **Flujo Principal**

1. **Carga de Datos**:
   * Lee el archivo Spamrial.csv y renombra columnas a label (1=spam, 0=ham) y text.
2. **Preprocesamiento**:
   * Aplica la clase Preprocesador a todos los textos.
3. **División de Datos**:
   * Separa en 80% entrenamiento y 20% prueba usando train\_test\_split.
4. **Entrenamiento**:
   * Crea una instancia de BayesSpam y entrena con los datos de entrenamiento.
5. **Evaluación**:
   * Predice las etiquetas de prueba y genera el reporte en español.

# **Instrucciones de Uso**

1. Ejecutar el script en un entorno con las dependencias instaladas.
2. Asegurarse de que el archivo Spamrial.csv esté en el directorio correcto.
3. Los resultados se imprimirán en consola con el formato especificado.