نموذج توقع انسحاب المستخدمين (Customer Churn)

يوضح المستند خطوات تطوير نموذج لتوقع انسحاب المستخدمين (Customer Churn) من منصة إلكترونية تعتمد على نموذج الاشتراكات. يعتمد المشروع على سجلات نشاط المستخدمين في خدمة بث موسيقا. وبناءً على سلوك المستخدمين في استخدام المنصة يتنبأ نموذج التعلم الآلي على المستخدمين الذين من المرجّح أن يلغوا اشتراكهم.

# أولًا: تحليل البيانات وفهم تحدياتها

## وصف البيانات:

وصف

## تحليل البيانات:

أهم

## هندسة الخصائص:

ما هي أهم المزايا

## تحدي عدم توازن الفئات (Class Imbalance):

كيف يمكن أن يتم حله

## معالجة تسرب بيانات (Data Leakage).

ما هو وماذا يعني وكيف يحدث

# ثانيًا: تدريب النموذج وتقييمه

## اختيار النموذج:

اختيار نموذج ملائم للمشكلة وشرحه بشكل واضح

## تدريب النموذج:

اختيار نموذج ملائم للمشكلة وشرحه بشكل واضح

## تقييم النموذج:

تقييم الأداء باستخدام المقاييس المناسبة.

## تحليل أخطاء النموذج

تحليل الأخطاء

# ثالثًا: بناء نظام متكامل.

## بناء نظام لإعادة التدريب الدوري للنموذج لمعالجة التغير في البيانات.

عند إطلاق نموذج للتعامل مع بيانات في (production) قد تختلف البيانات في بيئة (production) مع مرور الوقت عن البيانات التي دُرب عليها النموذج هذا الاختلاف قد يكون في توزيع البيانات (Data Drift) وقد يكون في طبيعية العلاقات بين المدخلات والمخرجات (Concept Drift) بمعنى أن يقل تأثير بعض المزايا على المخرجات وقد يزداد تأثير مزايا أخرى.

يضمن وجود هذا النظام وجود نسخة مختلفة من النموذج تدرب بشكل تلقائي على البيانات الجديدة بحيث يظل النموذج متوافقًا مع طبيعة البيانات الجديدة، يتم هذا عن طريق:

* تحديد آلية Triggering قد تكون دورية مثلًا كل شهر أو عند عتبه محددة
* تجهيز Pipeline لتنظيف وتحليل البيانات ومن ثمَّ تدريب النموذج عليها
* آلية لحفظ نسخة مختلفة من النموذج باستخدام MLFlow
* اختبار النموذج الجديد واستبداله مع النموذج الحالي

## تهيئة المشروع (Packaging) ومشاركته عبر «قيت هب»

## تقديم النموذج عبر (API) FastAPI.

**Tasks**

* Get to know the data on ipynb
* Creating the enviormement using uv
* Using docker DockerFile, and command
* Do Pre-commit with black and ruff for code quality
* Makefile for shortscuts
* README.md file
* Using Mlflow
* Monitoring System for data & concept drift and Model Performance

1. توثيق المشروع من خلال مستند تقني واضح يحتوي على تفاصيل الخطوات، والميزات المستخدمة، واختيارات النماذج، وإستراتيجية إعادة التدريب. ذكر أي تحديات تقنية أو اقتراحات لتحسين المشروع.
2. كتابة README يحتوي على توثيق تشرح فيه طريقة حلك للمشكلة، وأهم الصعوبات التي واجهتها، وإن كانت لديك إقتراحات أو أفكار لحل المشكلة بطريقة مختلفة او بشكل أفضل.

## ملاحظات:

* تم تعريب جميع المصطلحات التقنية في المستند باستخدام [معجم البيانات والذكاء الاصطناعي](https://siwar.ksaa.gov.sa/public-dict-information/4cd164a7-7160-4de8-af5c-34382f5da657)

وهو معجم تم بناءه من قبل مَجمع الملك سلمان العالمي للغة العربية بالتعاون مع الهيئة السعودية للبيانات والذكاء الاصطناعي (سدايا)

Data Leakage

This function didn't pass the test

def prob\_word\_given\_class(word, cls, word\_frequency, class\_frequency):

"""

Bernoulli NB: P(word | class) using document frequency with Laplace smoothing.

Parameters

----------

word : str

cls : 'spam' or 'ham' (or 1/0, True/False)

word\_frequency : dict -> {word: {"spam": df\_spam, "ham": df\_ham}}

class\_frequency : dict -> either {'spam': n\_spam\_emails, 'ham': n\_ham\_emails}

or {1: n\_spam\_emails, 0: n\_ham\_emails}

Returns

-------

float

(df(word,class) + 1) / (N\_class + 2)

"""

# Normalize class key to 'spam' or 'ham'

if cls in (0, 1, True, False):

cls\_str = 'spam' if int(cls) == 1 else 'ham'

else:

cls\_str = str(cls) # expect 'spam' or 'ham'

# Document frequency: how many emails in this class contain the word (0 if unseen)

df = word\_frequency.get(word, {"spam": 0, "ham": 0}).get(cls\_str, 0)

# Number of emails in this class (support both string- and int-keyed dicts)

if cls\_str in class\_frequency:

N\_cls = class\_frequency[cls\_str]

else:

N\_cls = class\_frequency[1 if cls\_str == 'spam' else 0]

# Bernoulli Laplace smoothing (add-1 / add-2)

return (df + 1) / (N\_cls + 2)