



TACTICAL

STRATEGIC

TRANSFORMATIONAL

PROJECT NAME: Lyrics Classifier
DATE: 08/01/2025

TEAM MEMBERS: Gabriel Rocha

DATA ANALYSIS & EXPERIMENT MANAGEMENT

3

A análise dos dados de entrada é feita com Python em um notebook Jupyter e pode ser reproduzida tanto localmente quanto em nuvem.

Bibliotecas como Pandas, Numpy, Scikit-Learn e Matplotlib foram utilizadas para desenvolver as análises.

DATA SOURCES & DATA VERSIONING

2

O versionamento dos dados é indicado por se tratar de uma aplicação com foco em tendências atuais, e há valor em obter um histórico dessas tendências.

As letras de músicas usadas para treinamento podem ser extraídas de sites que armazenam letras originais e traduções, e serão armazenadas em uma Feature Store no HopsWorks.

A rotulagem dos dados é de responsabilidade dos sites de origem.

VALUE PROPOSITION

1

[Keep solving the real problem]

O modelo é destinado a escritores, compositores e autores de músicas em geral. Eles serão capazes de classificar as suas composições em um gênero específico, baseado nas tendências atuais.

Com isso, é possível direcionar seu produto a artistas específicos que se encaixam nos temas abordados em suas letras.

CI/CT/CD: ML PIPELINE ORCHESTRATION

6

[Building, testing, packaging and deploying of ML pipeline]

[EM PROGRESSO]
Os modelos serão retreinados mensalmente com novos dados de entrada extraídos das fontes escolhidas. O pipeline projeto terá os seguintes passos:

- Pipeline de ETL (Extract, Transform, Load) em Python executada via Github Actions, com extração de dados via Web Scraping e carregamento no HopsWorks.
- Treinamento e Busca por Hiperparâmetros com Weights & Biases.
- Deploy do modelo via [A DEFINIR].
- Monitoramento via [A DEFINIR].

MODEL REGISTRY & MODEL VERSIONING

7

[EM PROGRESSO]

MODEL DEPLOYMENT

8

[EM PROGRESSO]

PREDICTION SERVING

9

O modelo fará previsões em tempo real (online) para possibilitar a edição dinâmica dos dados de entrada com um feedback rápido. Pelo baixo custo computacional projetado, o modelo pode ser hospedado de forma central.

FEATURE STORE & WORKFLOWS

4

[Keep feature computation consistent along the ML lifecycle]

O uso de Feature Stores é altamente indicado devido a filosofia de automação do projeto. Os dados coletados mensalmente devem ser processados, armazenados e utilizados para treinamento para garantir a atualidade do modelo.

O Workflow dos dados será automatizado via Github Actions.

FOUNDATIONS

5

[Reflecting DevOps]

[EM PROGRESSO]

DADOS: HOPSWORKS
ORQUESTRA: GITHUB ACTIONS

DEFINIR FERRAMENTAS DE MONITORAMENTO E DEPLOY

MODEL & DATA & APPLICATION MONITORING

10

- Is this optional or mandatory?
- What ML metrics are collected?
- What domain-specific metrics are collected?
- How is the model staleness detected? (Data Monitoring)
- How is the data skew detected? (Data Monitoring)
- What operational aspects need to be monitored? (SRE)
- What is the alerting strategy? (thresholds)
- What triggers the model re-training?

MLOPS DILEMMAS

10

[EM PROGRESSO]

METADATA STORE

11

Dados referentes aos processos de coleta, limpeza e treinamento são armazenados para detectar possíveis pontos de melhorias no processo. Tempo de extração dos dados, tempo de treinamento e inferência do modelo, duração de execução da pipeline completa e consumo de recursos computacionais são meta métricas que serão coletadas durante o tempo de vida do projeto.

METADATA MANAGEMENT

ML DESIGNKIT:

DATALANDSCAPE CANVAS: <https://www.canvasgeneration.com/canvas/data-landscape/>

ML DESIGN CANVAS: <https://www.louisdorard.com/machine-learning-canvas>

Designed by Larysa Visengeriyeva @visenger

Licensed under a Creative Commons

Attribution-ShareAlike 4.0 International License.

DATA & CODE MANAGEMENT

MODEL MANAGEMENT