# ELE 606 2024.1 – Prof. José Alfredo F. Costa – Unid. 3

# ELE 606 2024 - 3ª Unidade

## Projetos de Classificação de Texto com NLP e Machine Learning

### Introdução

Este documento orienta os projetos de classificação de texto utilizando Processamento de Linguagem Natural (NLP) e Machine Learning (ML). O foco será em explorar diversas bases de dados, entender o contexto de cada uma e aplicar técnicas avançadas para resolver problemas de classificação de texto do mundo real.

### Estrutura Geral dos Projetos

Cada projeto seguirá esta estrutura geral:

1. Coleta e pré-processamento de dados
2. Exploração e análise dos dados
3. Implementação de um modelo baseline
4. Implementação de modelos avançados usando LLMs
5. Avaliação e comparação de modelos
6. Análise de resultados e conclusões

### Lista de Projetos e Bases de Dados

1. **Coronavirus Tweets NLP - Text Classification**
   - **Fonte**: Kaggle - Coronavirus Tweets NLP - Text Classification
   - **Descrição**: Contém tweets relacionados ao coronavírus para classificação de texto.
   - **Objetivo**: Classificar tweets em categorias relacionadas ao COVID-19 ou análise de sentimentos.
   - **Métricas**: Acurácia, F1-score, Matriz de Confusão
   - **Modelos**: NLTK, Langchain com GPT, Llama
2. **Ecommerce Text Classification**
   - **Fonte**: Kaggle - Ecommerce Text Classification
   - **Descrição**: Dados de texto relacionados a transações de comércio eletrônico.
   - **Objetivo**: Classificar produtos em categorias ou analisar sentimentos de avaliações.
   - **Métricas**: Precisão, Recall, AUC-ROC
   - **Modelos**: Naive Bayes, Langchain com BERT, Llama
3. **Hierarchical Text Classification**
   - **Fonte**: Kaggle - Hierarchical Text Classification
   - **Descrição**: Conjunto de dados para classificação de texto hierárquica.
   - **Objetivo**: Classificar documentos em categorias hierárquicas.

- **Métricas**: Acurácia hierárquica, F1-score ponderado
- **Modelos**: SVM, Langchain com RoBERTa, Llama

4. **Spam Text Message Classification**
   - **Fonte**: Kaggle - Spam Text Message Classification
   - **Descrição**: Mensagens de texto rotuladas como spam ou não-spam.
   - **Objetivo**: Detectar spam em mensagens de texto.
   - **Métricas**: Precisão, Recall, F1-score
   - **Modelos**: Logistic Regression, Langchain com DistilBERT, Llama
5. **Text Classification Documentation**
   - **Fonte**: Kaggle - Text Classification Documentation
   - **Descrição**: Documentos de texto variados para classificação.
   - **Objetivo**: Classificar diferentes tipos de documentos.
   - **Métricas**: Acurácia, Kappa de Cohen
   - **Modelos**: Random Forest, Langchain com ALBERT, Llama
6. **Legal Citation Text Classification**
   - **Fonte**: Kaggle - Legal Citation Text Classification
   - **Descrição**: Citações legais para classificação de texto.
   - **Objetivo**: Classificar e analisar citações em documentos jurídicos.
   - **Métricas**: Precisão, Recall, F1-score
   - **Modelos**: XGBoost, Langchain com T5, Llama
7. **BBC Full Text Document Classification**
   - **Fonte**: Kaggle - BBC Full Text Document Classification
   - **Descrição**: Documentos de texto completos da BBC.
   - **Objetivo**: Classificar notícias por categoria.
   - **Métricas**: Acurácia, F1-score macro
   - **Modelos**: CNN, Langchain com ELECTRA, Llama
8. **Text Document Classification Dataset**
   - **Fonte**: Kaggle - Text Document Classification Dataset
   - **Descrição**: Conjunto de dados de documentos de texto variados.
   - **Objetivo**: Classificar documentos em diferentes categorias.
   - **Métricas**: Acurácia balanceada, Matthews Correlation Coefficient
   - **Modelos**: LSTM, Langchain com XLNet, Llama
9. **Medical Text Dataset - Cancer Doc Classification**
   - **Fonte**: Kaggle - Medical Text Dataset - Cancer Doc Classification
   - **Descrição**: Documentos médicos relacionados ao câncer.
   - **Objetivo**: Classificar documentos médicos por tipo de câncer.
   - **Métricas**: Precisão, Recall, F1-score por classe
   - **Modelos**: Bi-LSTM, Langchain com BART, Llama
10. **Email Spam Text Classification Dataset**
    - **Fonte**: Kaggle - Email Spam Text Classification Dataset
    - **Descrição**: E-mails rotulados como spam ou não-spam.
    - **Objetivo**: Detectar spam em e-mails.
    - **Métricas**: AUC-ROC, Precisão, Recall
    - **Modelos**: FastText, Langchain com DistilGPT-2, Llama
11. **SMS Spam Collection (Text Classification)**
    - **Fonte**: Kaggle - SMS Spam Collection (Text Classification)
    - **Descrição**: Coleção de mensagens SMS rotuladas como spam ou não-spam.
    - **Objetivo**: Detectar spam em mensagens SMS.
    - **Métricas**: Acurácia, F1-score, Precisão

- **Modelos**: Transformer, Langchain com ALBERT, Llama

12. **Dataset Text Document Classification**
    - **Fonte**: Kaggle - Dataset Text Document Classification
    - **Descrição**: Conjunto de dados de documentos de texto para classificação.
    - **Objetivo**: Classificar documentos por categoria.
    - **Métricas**: Acurácia, F1-score ponderado, Log Loss
    - **Modelos**: BERT, Langchain com T5, Llama
13. **The Social Dilemma Tweets - Text Classification**
    - **Fonte**: Kaggle - The Social Dilemma Tweets - Text Classification
    - **Descrição**: Tweets relacionados ao documentário "The Social Dilemma".
    - **Objetivo**: Analisar sentimentos ou classificar tópicos relacionados ao documentário.
    - **Métricas**: Acurácia, F1-score, AUC-ROC
    - **Modelos**: RoBERTa, Langchain com GPT, Llama
14. **BBC Full Text Document Classification (Repetido)**
    - **Fonte**: Kaggle - BBC Full Text Document Classification
    - **Descrição**: Documentos de texto completos da BBC para classificação.
    - **Objetivo**: Classificar notícias por categoria.
    - **Métricas**: Acurácia, F1-score macro, Kappa de Cohen
    - **Modelos**: XLNet, Langchain com ELECTRA, Llama
15. **Text Classification on Emails**
    - **Fonte**: Kaggle - Text Classification on Emails
    - **Descrição**: E-mails variados para classificação de texto.
    - **Objetivo**: Classificar e-mails por categorias específicas.
    - **Métricas**: Precisão, Recall, F1-score por classe
    - **Modelos**: ALBERT, Langchain com T5, Llama

## Passos Detalhados para os Projetos

1. **Coleta e pré-processamento de dados**
    - Baixar o conjunto de dados especificado
    - Realizar limpeza de texto (remover pontuação, converter para minúsculas, etc.)
    - Tokenização e remoção de stopwords
    - Dividir os dados em conjuntos de treinamento, validação e teste
2. **Exploração e análise dos dados**
    - Análise estatística básica (contagem de classes, comprimento dos textos, etc.)
    - Visualização da distribuição de classes
    - Análise de frequência de palavras
    - Identificação de padrões ou características importantes nos dados
3. **Implementação de um modelo baseline**
    - Usar técnicas clássicas de NLP (bag-of-words, TF-IDF)
    - Implementar um modelo simples (por exemplo, Naive Bayes ou Logistic Regression)
    - Avaliar o desempenho do modelo baseline usando as métricas especificadas
4. **Implementação de modelos avançados usando LLMs**

    - Usar Langchain para integrar modelos pré-treinados (como GPT, BERT, RoBERTa)
    - Implementar fine-tuning ou transfer learning conforme necessário
    - Experimentar com diferentes arquiteturas e hiperparâmetros
    - Implementar um modelo usando Llama e comparar com os outros
5. **Avaliação e comparação de modelos**
    - Avaliar todos os modelos usando as métricas especificadas
    - Realizar validação cruzada para obter resultados mais robustos
    - Comparar o desempenho dos diferentes modelos
    - Analisar casos de erro e identificar pontos fortes e fracos de cada abordagem
6. **Análise de resultados e conclusões**
    - Interpretar os resultados obtidos
    - Discutir as vantagens e desvantagens de cada abordagem
    - Propor melhorias ou direções futuras para o projeto
    - Preparar uma apresentação ou relatório final

## Pontos Adicionais para Discussão

1. **Ética e Viés**: Discuta a importância de considerar questões éticas ao trabalhar com esses conjuntos de dados, especialmente aqueles que lidam com conteúdo sensível.
2. **Pré-processamento**: Enfatize a importância do pré-processamento de dados e como diferentes técnicas podem afetar o desempenho do modelo.
3. **Escolha do Modelo**: Discuta as vantagens e desvantagens de usar modelos pré-treinados versus treinar modelos do zero para tarefas específicas.
4. **Avaliação de Modelos**: Explique a importância de usar múltiplas métricas de avaliação e como interpretar os resultados.
5. **Otimização de Hiperparâmetros**: Introduza o conceito de otimização de hiperparâmetros e como isso pode melhorar o desempenho do modelo.
6. **Interpretabilidade**: Discuta a importância da interpretabilidade do modelo, especialmente em aplicações do mundo real.
7. **Desafios Específicos**: Para cada conjunto de dados, discuta os desafios específicos que os alunos podem encontrar (por exemplo, desequilíbrio de classes, dados ruidosos, etc.).
8. **Extensões do Projeto**: Sugira maneiras de expandir cada projeto, como incorporar análise de sentimentos em várias línguas ou combinar múltiplos conjuntos de dados.

## Conclusão

Estes projetos proporcionarão uma experiência prática abrangente com técnicas de NLP e machine learning, permitindo aos alunos enfrentar desafios reais de classificação de texto e explorar diversas aplicações práticas. Ao trabalhar com diferentes conjuntos de dados e implementar vários modelos, os alunos ganharão uma compreensão profunda dos desafios e oportunidades no campo de NLP e aprendizado de máquina.