

ANO

2024



UNINTER

**CADERNO DE RESPOSTAS DA ATIVIDADE
PRÁTICA DE:**

NLP

ALUNO: GUILHERME LOURENÇO - RU: 3799879

**Caderno de Resposta Elaborado por:
Prof. MSc. Guilherme Ditzel Patriota**

Prática 01 – Criação de modelo de classificação supervisionado para análise de fake News.

Questão 01 – Apresente aqui o código referente ao modelo gerado e a nuvem de palavras que foram usadas para identificar textos VERDADEIROS

ENUNCIADO: Veja o Roteiro da Atividade Prática para mais detalhes.

I. Apresentação do Código (não esquecer do identificador pessoal):

```
[32] # Adicione as previsões ao DataFrame de teste
df_teste['previsoes'] = previsoes

# Separe os textos por classe
text_real = ' '.join(df_teste[df_teste['previsoes'] == 'true']['coluna_noticias'])

wordcloud = WordCloud(background_color="white",
                        max_words=500,
                        mask=imagem_mascara,
                        contour_color="grey",
                        contour_width=0.3,
                        min_font_size=5,
                        random_state=42).generate(text_real)

# Mostrando a imagem
def gerar_wordcloud(texto, titulo):
    plt.figure(figsize=(8, 8), facecolor=None)
    plt.imshow(wordcloud)

    #plt.title(titulo)

# Remove os eixos do gráfico
plt.axis('off')
plt.tight_layout(pad=0)
plt.text(0.07, 0.5, titulo, transform=plt.gca().transAxes, rotation=90, va='center', ha='left', fontsize=14)
plt.show()

# Gerar nuvem de palavras para TRUE
gerar_wordcloud(text_real, 'NUVEM DE PALAVRAS TRUE - RU_3799879')
```

Figura 1 **CÓDIGO TRUE**

II. Apresentação das Imagens/Print do resultado (não esquecer do identificador):



Figura 2: WORDCLOUD TRUE

III. Responda à pergunta: Quantas palavras, bigramas e trigramas foram usados dos textos rotulados como REAL para a criação de seu modelo e qual a acurácia?

Resposta: Quantidade de palavras usadas: 65816

Quantidade de bigramas usados: 2496262

Quantidade de trigramas usados: 1465374

Acurácia do modelo para textos rotulados como TRUE: 1.0

Prática 01 – Criação de modelo de classificação supervisionado para análise de fake News.

Questão 02 – Apresente aqui o código referente ao modelo gerado e a nuvem de palavras que foram usadas para identificar textos FALSOS.

ENUNCIADO: Veja o Roteiro da Atividade Prática para mais detalhes.

I. Apresentação do Código (não esquecer do identificador pessoal):

```
[37] # Adicione as previsões ao DataFrame de teste
df_teste['previsoes'] = previsoes

# Separe os textos por classe
text_fake = ' '.join(df_teste[df_teste['previsoes'] == 'fake']['coluna_noticias'])

wordcloud = WordCloud(background_color='white',
                        max_words=500,
                        mask=imagem_mascara,
                        contour_color='grey',
                        contour_width=3,
                        min_font_size=3,
                        random_state=42).generate(text_fake)

[38] # Mostrando a imagem
def gerar_wordcloud(texto, titulo):
    plt.figure(figsize=(8, 8), facecolor=None)
    plt.imshow(wordcloud)

    #plt.title(titulo)

    #remove os eixos do gráfico
    plt.axis("off")
    plt.tight_layout(pad=0)
    plt.text(0.07, 0.5, titulo, transform=plt.gca().transAxes, rotation=90, va='center', ha='left', fontsize=14)
    plt.show()

# Gerar nuvem de palavras para FAKE
gerar_wordcloud(text_fake, 'NUVEM DE PALAVRAS FAKE - RU_3799879')
```

Figura 3: CÓDIGO FAKE

II. Apresentação das Imagens/Print do resultado (não esquecer do identificador):

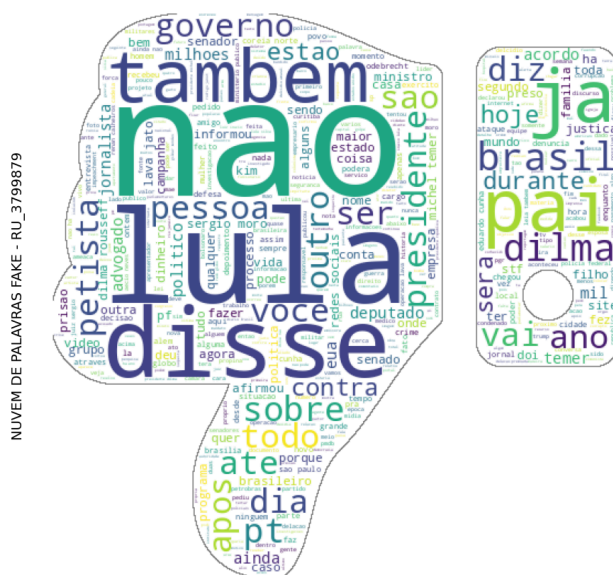


Figura 4: WORDCLOUD FAKE

III. Responda à pergunta: Quantas palavras, bigramas e trigramas foram usados dos textos rotulados como FAKE para a criação de seu modelo, quais foram as técnicas usadas em seu pré-processamento e qual tipo de modelo foi escolhido para este classificador?

Resposta: Quantidade de palavras usadas: 27967

Quantidade de bigramas usados: 483629

Quantidade de trigramas usados: 268475

Acurácia do modelo para textos rotulados como FAKE: 1.0

Técnica de usar o SelectKBest para selecionar as TOP 50 palavras com TDF e bigramas

Modelo foi criado usando Multinomial Naive Bayes

LINK PARA O REPOSITÓRIO: https://github.com/guilourenco17/NLP_UNINTER