

# UMA ABORDAGEM DE CIÊNCIA DE DADOS PARA IDENTIFICAR FAKE NEWS NO ÂMBITO POLÍTICO

.....

Marcelo Hideaki Iwata Kito  
Orientador: Prof. Assoc. João Pedro Albino

# INTRODUÇÃO

- Internet - tecnologias e pessoas
- Fácil acesso
- Grande fluxo de informações
  - Inclusive informações enganosas - Fake News
- Fake news afetam diversas esferas da sociedade

# INTRODUÇÃO

- Internet como ferramenta política
  - Eleição de 2008 dos EUA - Barack Obama
- Contexto brasileiro: eleições de 2018
  - Forte polarização
  - Fake news sobre os candidatos
- Ferramentas de comunicação facilitam a propagação de desinformação

# OBJETIVO

- Desenvolvimento de um aplicativo móvel que, dado uma notícia, verifica se esta é verdadeira ou não

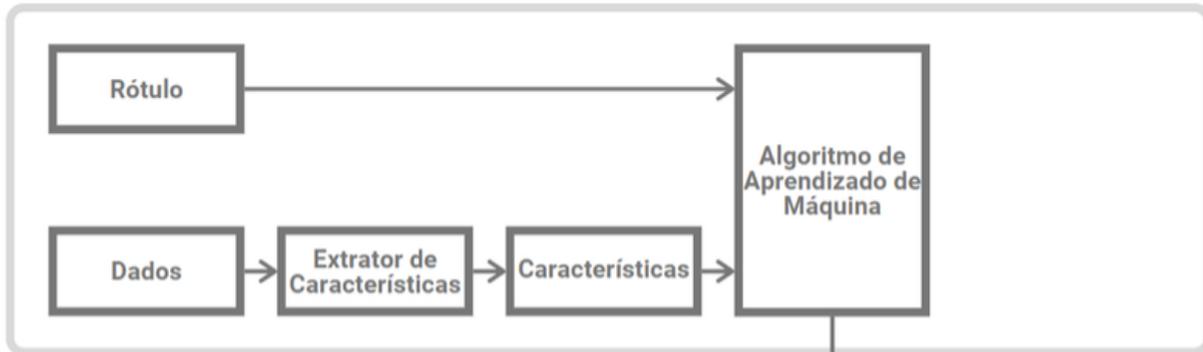
# IDENTIFICAÇÃO DE FAKE NEWS

- Abordagens para detecção automática de fake news utilizam características da linguagem
- Remete a um esforço de Processamento de Linguagem Natural (NLP)

# APRENDIZADO DE MÁQUINA

- Neste trabalho: aprendizado de máquina supervisionado - classificação [1]

## Treinamento



## Predição



# APRENDIZADO DE MÁQUINA

- Regressão Logística
- Naive Bayes
- Florestas Aleatórias
- Support Vector Machines (SVM)

- Textos em linguagem natural (português)
- Representação das palavras
  - Computadores lidam melhor com números
- Processo de vetorização
  - Transformação das palavras (tokens) em valores numéricos

- Bag-of-words
  - Frequência das palavras
- TF-IDF (Term Frequency - Inverse Document Frequency)
  - ↑ peso quanto mais uma palavra aparece em um texto
  - ↓ peso quanto mais uma palavra aparece em vários documentos

# FERRAMENTAS

PYTHON



# Flask

web development,  
one drop at a time



HTML,  
CSS,  
JAVASCRIPT



# DESENVOLVIMENTO

- Necessidade de dados das duas classes
- Conjunto primário: coletado por mim
- Conjunto Fake.Br corpus: coletado por pesquisadores da Universidade de São Paulo (USP)

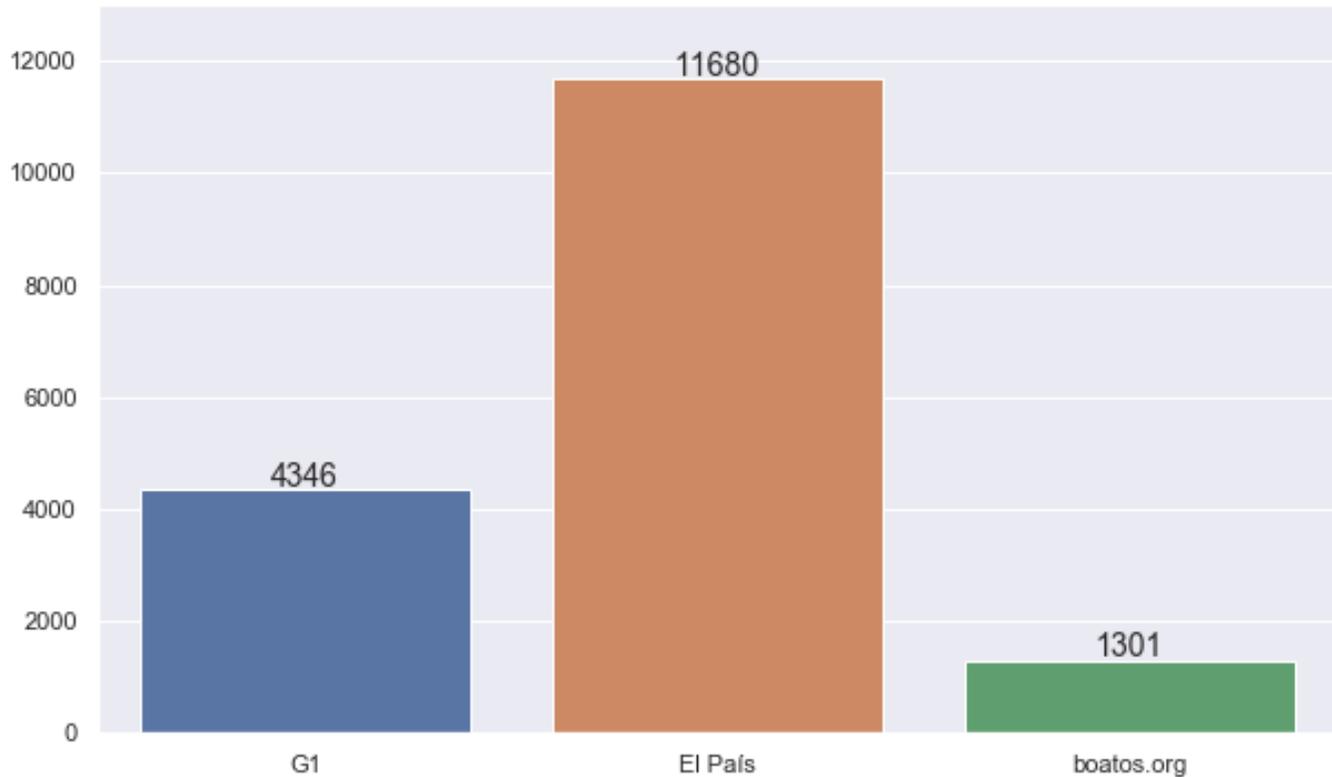
# AQUISIÇÃO DOS DADOS

## CONJUNTO PRIMÁRIO

- Desenvolvimento de web-crawlers:
  - Notícias falsas: boatos.org
  - Notícias verdadeiras: G1 e El País
- Conjunto desbalanceado
  - Muito mais amostras verdadeiras do que falsas
  - Viés ao treinamento do modelo

# AQUISIÇÃO DOS DADOS

CONJUNTO PRIMÁRIO



# AQUISIÇÃO DOS DADOS

CONJUNTO FAKE.BR CORPUS

- Notícias coletadas por pesquisadores da USP [2]
- Diversos assuntos (dia-a-dia, política, etc.)
- Uma notícia verdadeira para cada notícia falsa tratando de um assunto
- Tratam do mesmo intervalo temporal

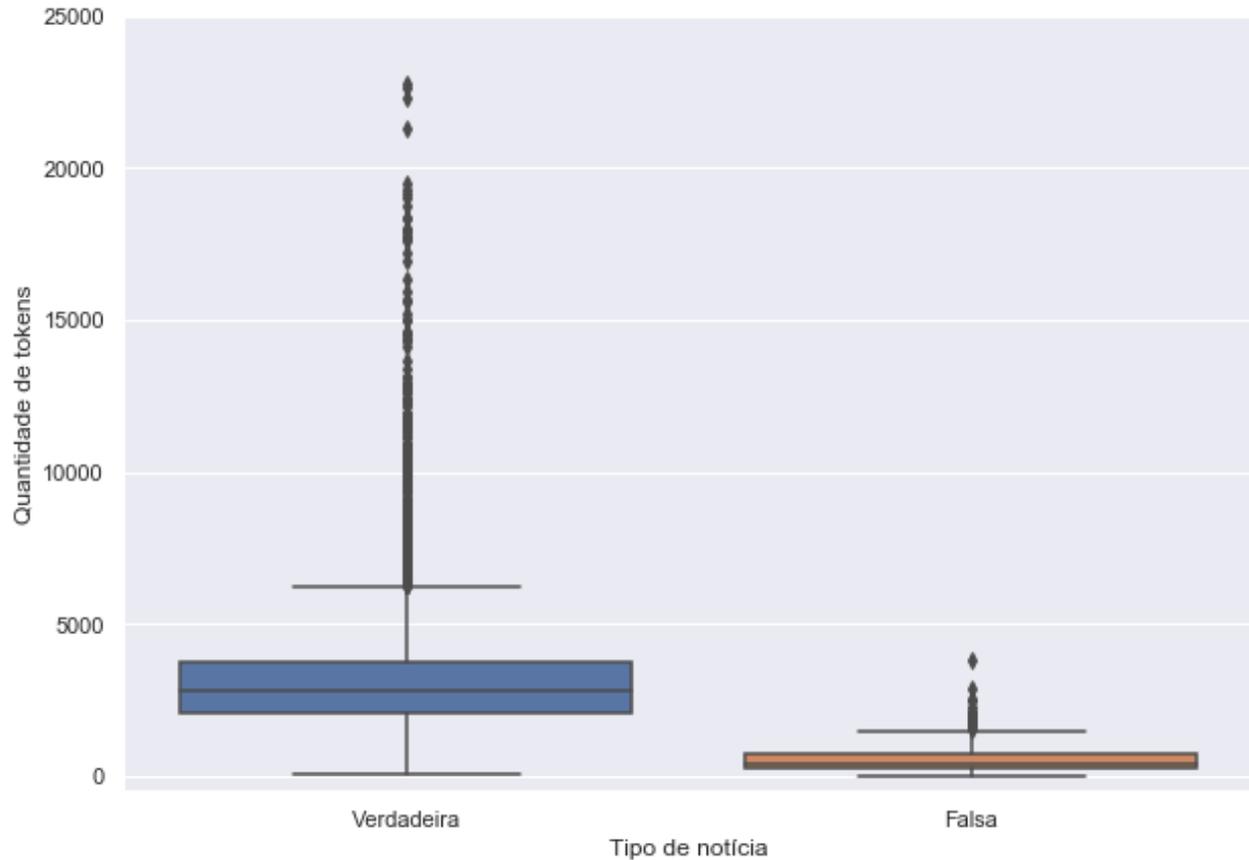
# ANÁLISE E PREP. DOS DADOS

## CONJUNTO PRIMÁRIO

- Remoção de notícias sem texto
  - Possuíam apenas vídeos ou imagens
- Definição de um intervalo temporal
  - 01/2016 a 12/2018
- Análise dos tamanhos dos textos
  - Textos de notícias verdadeiras muito maiores que os textos de notícias falsas
  - Viés no treinamento dos modelos

# ANÁLISE E PREP. DOS DADOS

CONJUNTO PRIMÁRIO



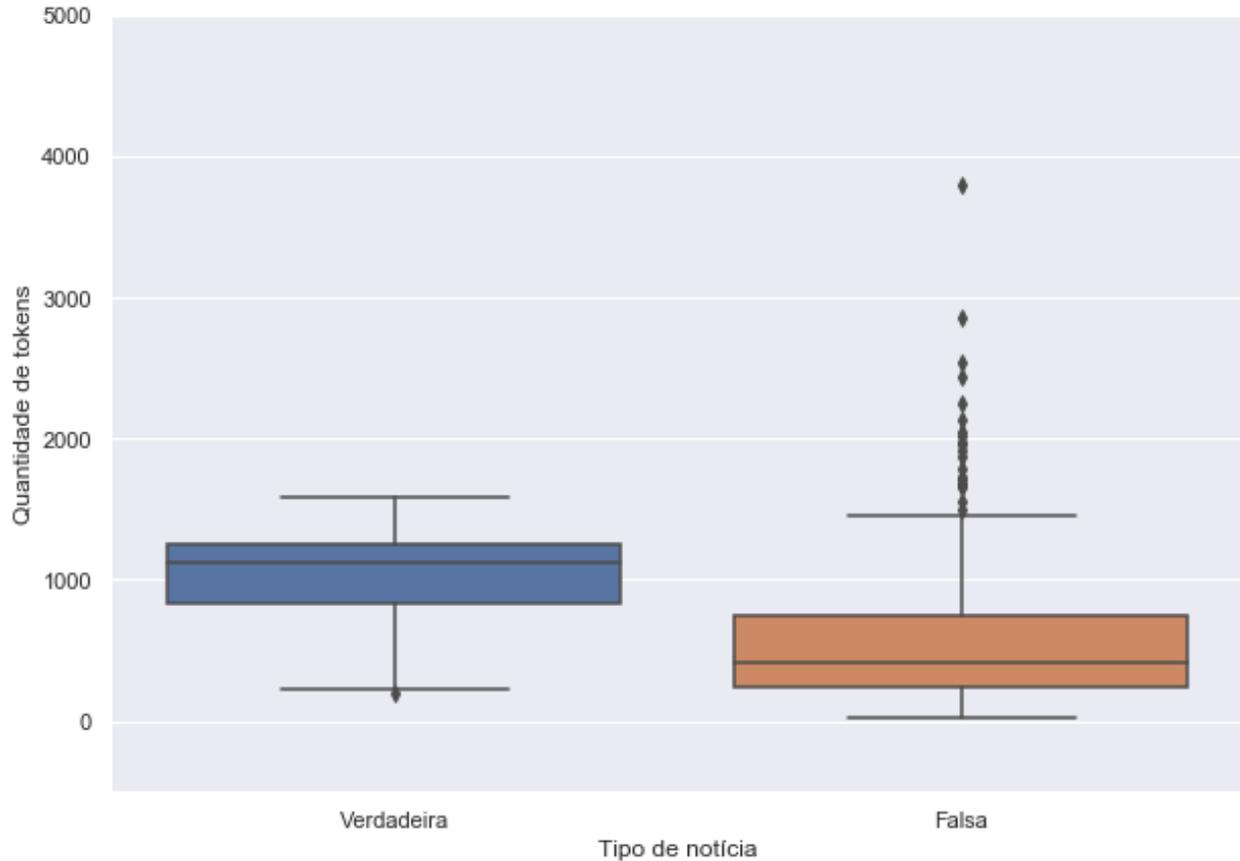
# ANÁLISE E PREP. DOS DADOS

## CONJUNTO PRIMÁRIO

- Foram selecionados os  $n$  textos mais curtos do conjunto de notícias verdadeiras, de forma que ficasse com a mesma quantidade de notícias falsas
  - Resolveu o desbalanceamento a nível de quantidade de documentos
  - Atenuou o desbalanceamento quanto ao tamanho dos textos

# ANÁLISE E PREP. DOS DADOS

## CONJUNTO PRIMÁRIO



# ANÁLISE E PREP. DOS DADOS

## CONJUNTO PRIMÁRIO

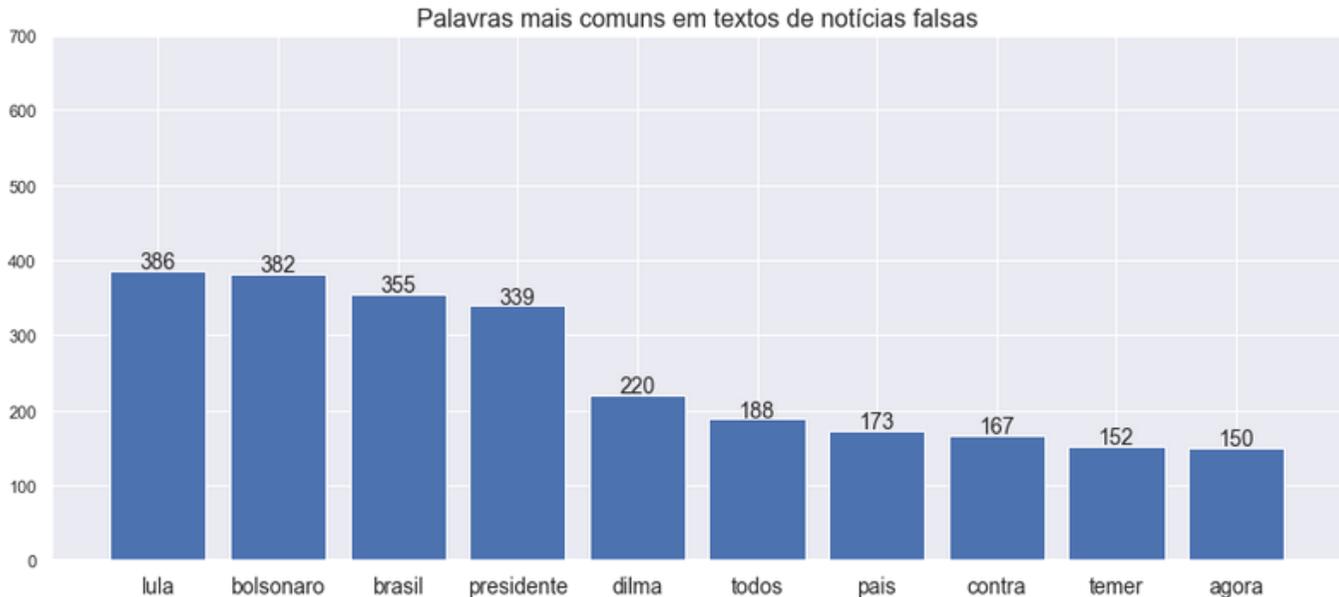
- Análise das palavras mais frequentes



# ANÁLISE E PREP. DOS DADOS

## CONJUNTO PRIMÁRIO

- Features usadas no modelo são as palavras



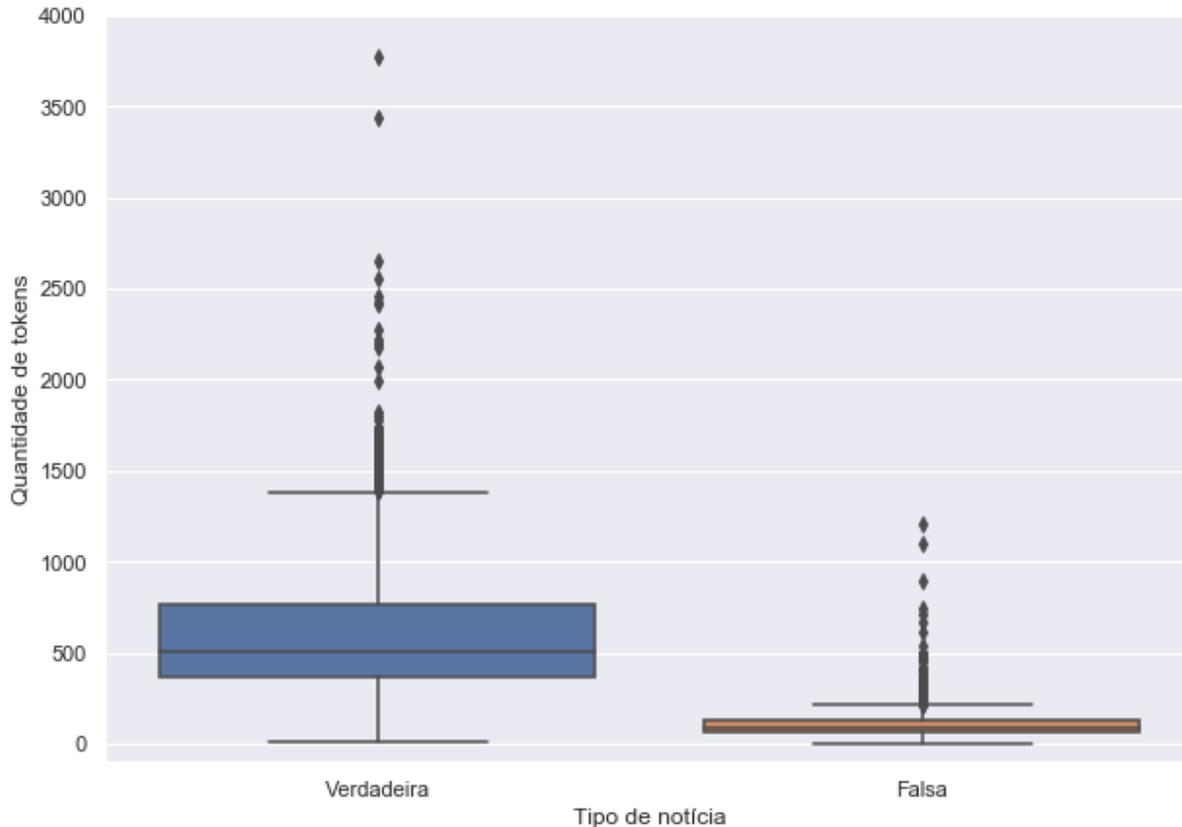
# ANÁLISE E PREP. DOS DADOS

CONJUNTO FAKE.BR CORPUS

- Recorte para pegar apenas notícias relacionadas à política
- Balanceadas: 2089 textos de cada classe
- Correspondem a um mesmo intervalo de tempo
- Desbalanceamento no tamanho dos textos
  - Textos truncados

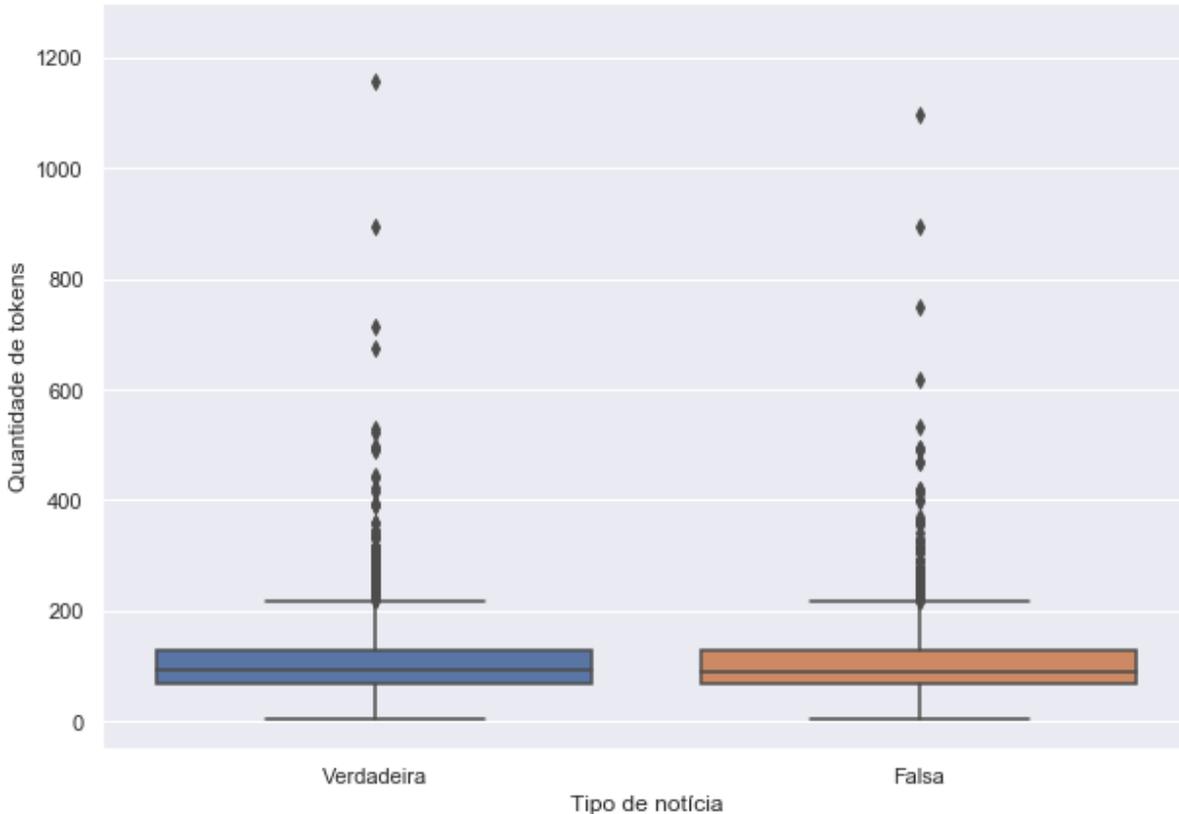
# ANÁLISE E PREP. DOS DADOS

CONJUNTO FAKE.BR CORPUS



# ANÁLISE E PREP. DOS DADOS

CONJUNTO FAKE.BR CORPUS



# ANÁLISE E PREP. DOS DADOS

CONJUNTO FAKE.BR CORPUS

- Análise das palavras mais frequentes



# ANÁLISE E PREP. DOS DADOS

CONJUNTO FAKE.BR CORPUS



# ANÁLISE E PREP. DOS DADOS

## PRÉ-PROCESSAMENTO GERAL

- Caracteres minúsculos
- Stemming: reduzir palavras ao radical
  - "corrida" e "correr" viram "corr"
  - agrupam palavras com mesmo sentido
- Remoção de acentos, caracteres não alfabéticos e palavras muito curtas (2 ou menos letras)

# ANÁLISE E PREP. DOS DADOS

## PRÉ-PROCESSAMENTO GERAL

- Remoção de stopwords: palavras "vazias" que não agregam à diferenciação
  - Ex.: "de", "e" e "em"
- Vetorização (biblioteca Scikit-Learn):
  - Bag-of-words: CountVectorizer
  - TF-IDF: TfidfVectorizer

# TREINAMENTO E TESTE

- 70% do conjunto para o treino
- 30% do conjunto para teste
- Os conjuntos de dados primário e Fake.Br corpus foram utilizados separadamente
- Utilizadas todas as combinações de vetorização + algoritmos de aprendizado de máquina

# TREINAMENTO E TESTES

## CONJUNTO PRIMÁRIO

- Todas as combinações de vetorização + modelo treinadas com a partição de treinamento e testadas com a partição de teste do conjunto primário

<b>Vetorizador</b>	<b>Modelo</b>	<b>Acurácia</b>
CountVectorizer	GaussianNB	0.799458
CountVectorizer	LogisticRegression	0.915989
CountVectorizer	RandomForestClassifier	0.888889
CountVectorizer	SVC (kernel linear)	0.905149
TfidfVectorizer	GaussianNB	0.769648
TfidfVectorizer	LogisticRegression	0.899729
TfidfVectorizer	RandomForestClassifier	0.886179
TfidfVectorizer	SVC (kernel linear)	0.891599

# TREINAMENTO E TESTES

CONJUNTO FAKE.BR CORPUS

- Todas as combinações de vetorização + modelo treinadas com a partição de treinamento e testadas com a partição de teste do conjunto Fake.Br corpus

<b>Vetorizador</b>	<b>Modelo</b>	<b>Acurácia</b>
CountVectorizer	GaussianNB	0.802233
CountVectorizer	LogisticRegression	0.880383
CountVectorizer	RandomForestClassifier	0.860447
CountVectorizer	SVC (kernel linear)	0.870813
TfidfVectorizer	GaussianNB	0.794258
TfidfVectorizer	LogisticRegression	0.881180
TfidfVectorizer	RandomForestClassifier	0.855662
TfidfVectorizer	SVC (kernel linear)	0.855662

# TREINAMENTO E TESTES

## CONJUNTO DE TESTES GERAL

- A fim de realizar um teste mais justo e comparável, criou-se uma partição de teste a partir das partições de teste de cada conjunto de dados
  - 369 amostras do conjunto primário
  - 369 amostras do conjunto Fake.Br corpus

# TREINAMENTO E TESTES

CONJ. PRIMÁRIO + CONJUNTO DE TESTES GERAL

- Melhor acurácia: 0.810298
  - Vetorização: TF-IDF
  - Modelo: Regressão Logística

Vetorizador	Modelo	Acurácia
CountVectorizer	GaussianNB	0.714092
CountVectorizer	LogisticRegression	0.800813
CountVectorizer	RandomForestClassifier	0.785908
CountVectorizer	SVC (kernel linear)	0.791328
TfidfVectorizer	GaussianNB	0.708672
TfidfVectorizer	LogisticRegression	0.810298
TfidfVectorizer	RandomForestClassifier	0.791328
TfidfVectorizer	SVC (kernel linear)	0.798103

# TREINAMENTO E TESTES

CONJ. FAKE.BR CORPUS + CONJUNTO DE TESTES GERAL

- Melhor acurácia: 0.861789
  - Vetorização: TF-IDF
  - Modelo: Regressão Logística

<b>Vetorizador</b>	<b>Modelo</b>	<b>Acurácia</b>
CountVectorizer	GaussianNB	0.742547
CountVectorizer	LogisticRegression	0.838753
CountVectorizer	RandomForestClassifier	0.846883
CountVectorizer	SVC (kernel linear)	0.822493
TfidfVectorizer	GaussianNB	0.726287
TfidfVectorizer	LogisticRegression	0.861789
TfidfVectorizer	RandomForestClassifier	0.837398
TfidfVectorizer	SVC (kernel linear)	0.833333

# OTIMIZAÇÃO DE HIPER-PARÂM.

CONJ. FAKE.BR CORPUS + CONJUNTO DE TESTES GERAL

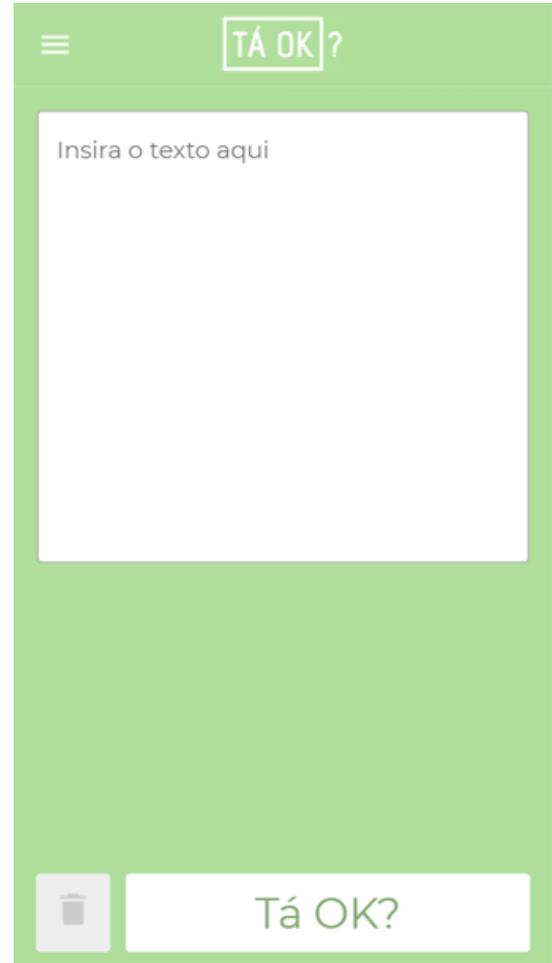
- GridSearchCV (Scikit-Learn)
  - Método de força bruta que varia parâmetros do modelo buscando otimizar uma certa métrica
- Elevou a acurácia da predição a 0.876693
- O modelo foi exportado para um arquivo

# DISPONIBILIZAÇÃO DO MODELO

- API REST com a biblioteca Flask, carregando o modelo exportado
- Recebe um JSON contendo o texto da notícia
- Retorna se esta é verdadeira ou falsa e a confiança associada à predição
- API hospedada na plataforma PythonAnywhere

# DESENVOLVIMENTO DO APP

- Desenvolvido utilizando Ionic
- Faz uma requisição à API e exibe a resposta ao usuário



# CONCLUSÃO

- Desenvolvimento de um pipeline para identificação de fake news
- Aplicação mobile para educar as pessoas para não acreditar cegamente em tudo que lê
- Pode-se treinar com outros conjuntos de dados, bastando seguir as premissas deste trabalho
- Emprego de tecnologias não aprendidas na faculdade e que são bastante utilizadas no mercado de trabalho

# TRABALHOS FUTUROS

- Melhorar a coleta dos dados
- Usar outros modelos e características

# REFERÊNCIAS

[1] BIRD, S.; KLEIN, E.; LOPER, E. Natural language processing with Python: analyzing text with the natural language toolkit.

[2] MONTEIRO, R. A.; SANTOS, R. L. S.; PARDO, T. A. S.; ALMEIDA, T. A. de; RUIZ, E.E. S.; VALE, O. A. Contributions to the study of fake news in portuguese: New corpus and automatic detection results.

**MUITO  
OBRIGADO!**