Machine Learning Engineer Nanodegree

Twitt Detector Celebrities

Bruno Santos

Março, 18, 2018

Definição

1. Project Overview

Com um mundo cada vez mais conectado, as pessoas utilizam de mídias sociais para expressar suas opiniões e sentimentos. Muitas pessoas utilizam as mídias como fonte de notícias no dia a dia, um dos meios mais populares para isso é o Twitter.

Nos últimos meses vimos como alguns ‘twitts’ quase provocaram guerras, já outros twitts servem para comunicar o andamento do projeto de levar o homem para Marte. Sendo assim, é possível identificar de quem é aquele ‘twitt’ sem nem mesmo saber quem o escreveu?

Sendo assim foram escolhidas 4 celebridades do twitter que se encontram no top 100 com mais seguidores:

- Barack Obama: ex-presidente dos EUA (3° mais seguido do twitter)
- Donald J. Trump: presidente dos EUA (20° mais seguido do twitter)
- Bill Gates: Fundador da Microsoft e filantrópico (22° mais seguido do twitter)
- Elon Musk: Fundador do Paypal, SpaceX e Tesla (92° mais seguido do twitter)

2. Problem Statement

O objetivo desse projeto é criar um reconhecimento de texto para identificação de quem é seu autor, o universo foi reduzido a 4 personalidades, porem pode ser aplicado em larga escala utilizando a mesma técnica mudando apenas o data set. Também para rápida consulta da predição, será criado um endpoint para consumo dessa predição. Para isso acontecer serão seguidos os seguintes passos:

- Consulta a API do twitter buscando twitts dos usuários pré-definidos.
- Tratamento e separação dos campos uteis para esse estudo.
- Preparação do dataset.
- Utilização de mecanismos de varredura de texto como split words.
- Vetorização dos textos e preparação da base de treino e teste.
- Resultado em diferentes classificadores.
- Coleta dos resultados e escolha do melhor classificador.
- Implantação de end-point para consulta da predição em formato json.

3. Metrics

Para cálculo das métricas será utilizada uma matriz de confusão para garantir que os dados de teste estão corretamente sendo realizados.
Como a análise de texto em um âmbito aberto não é garantia de nada, não será esperado um índice de acerto altíssimo. O algoritmo que possuir a média de 80% de acertos será considerado aprovado para esse estudo.

Também serão utilizados dados reais de twitts mais atuais das celebridades escolhidas para esse projeto.

Analysis

1. Data Exploration

Toda a base de dados foi extraído da API do twitter, que disponibiliza dados completos dos últimos 200 twitts do usuário escolhido. Para realização dessa etapa foi necessário criar uma API no twitter (https://developer.twitter.com/en/docs) realizar um HTTP Post para obter um token de acesso, e só assim consultar os dados dos usuários escolhidos (https://api.twitter.com/1.1/statuses/user_timeline.json?screen_name=elonmusk&count=200&tweet_mode=extended) e assim obter a resposta como exemplo abaixo:


```
{
	"created_at": "Fri Mar 09 22:53:25 +0000 2018",
	"id": 972243992153739265,
	"id_str": "972243992153739265",
	"full_text": "Boring Co urban loop system would have 1000’s of small stations the size of a single parking space that take you very close to your destination &amp;amp; blend seamlessly into the fabric of a city, rather than a small number of big stations like a subway",
	"truncated": false,
	"display_text_range": [
	   0,
	   250
	],
	"entities": {
	   "hashtags": [],
	   "symbols": [],
	   "user_mentions": [],
	   "urls": []
	}
```


Todos os dados recolhidos estão anexados nesse trabalho.
Para parse do recebimento do texto foi utilizado um método criado em .NET que realiza a leitura do json e extrai somente o texto e a data que foi publicado, além de realizar limpezas básicas no campo do texto. Esse método está em anexo a esse trabalho.
Sendo assim já temos em um arquivo único o autor, texto e data de cada twitt das celebridades escolhidas separadas por virgula em formato csv.
Com esses dados já é possível começar a criação do algoritmo de machine learning para

2. Exploratory Visualization

Como já foi definido que cada celebridade possui 200 twitts, não existe muita comparação quanto aos textos nesse ponto, porem foi feita uma análise de

como esses 200 twiits estão divididos em uma linha de tempo, que revela a frequência com que cada celebridade costuma publicar na rede social. Outra informação é em quantos dias cada celebridade postou 200 twitts, com uma diferença brutal entre Trump e Obama.

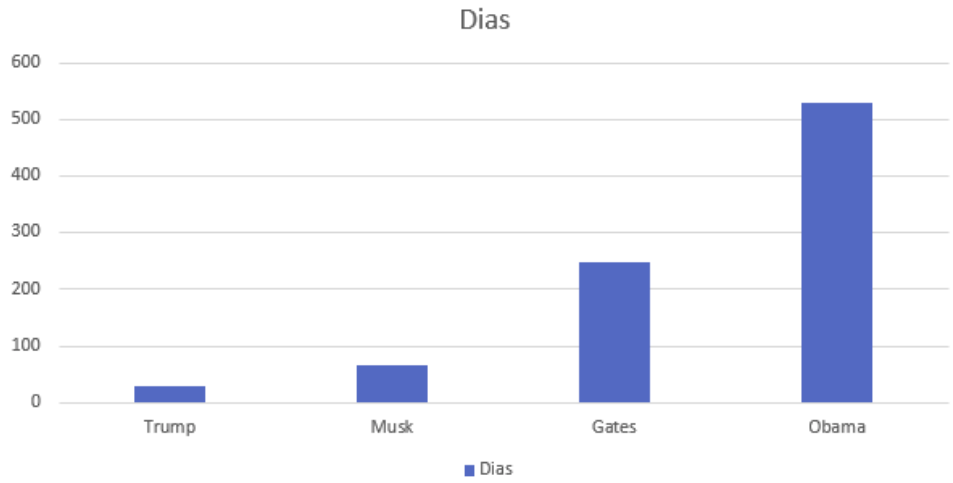

Imagem 1: Dias usados para postar 200 twitts

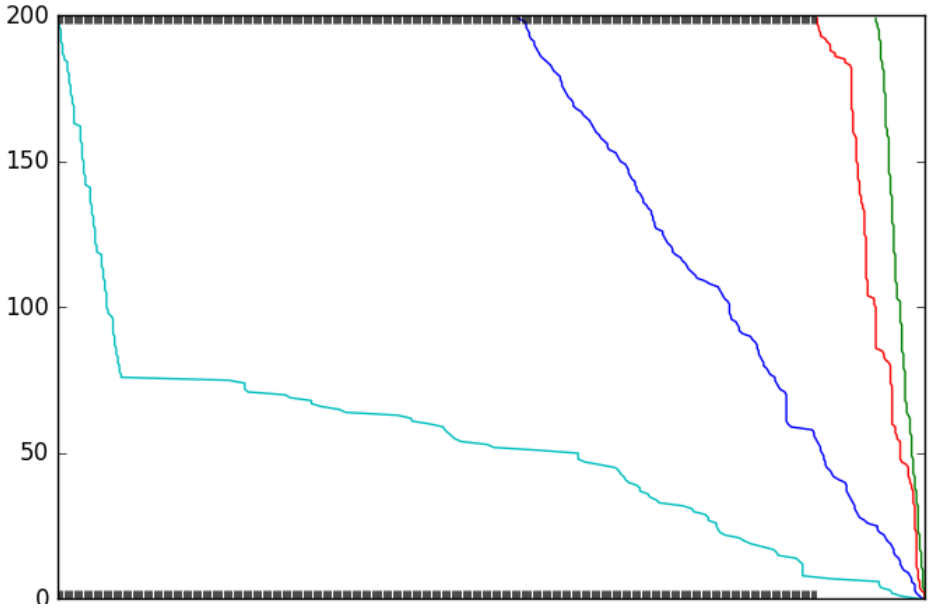

Imagem 2: De 0 a 200 twitter em dias

Outro ponto a se observar é a nuvem de palavras usadas:

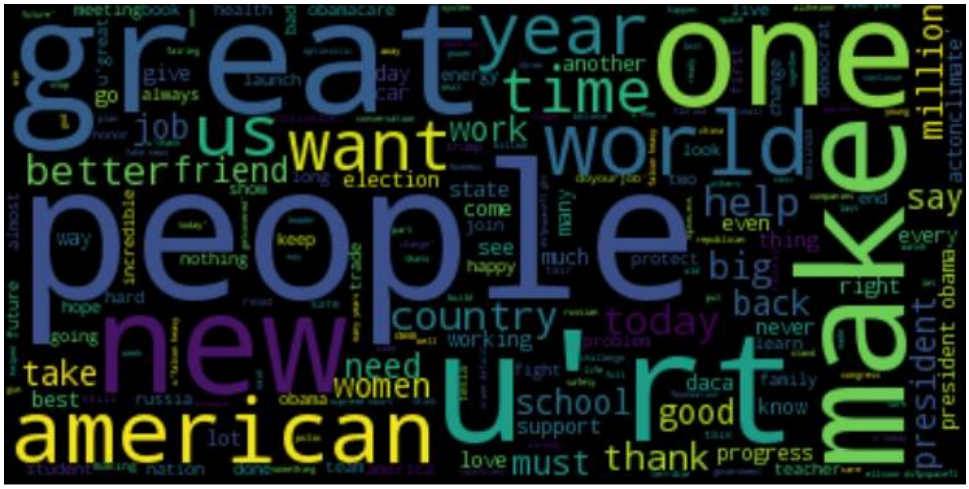

Imagem 3: Word Cloud geral

3. Algorithms and Techniques

Para análise de texto é de boa pratica primeiramente utilizar a limpeza de texto chamado de “stop words” que são palavras auxiliares, diferentes em cada língua. Em português “para, que, até” são consideradas stop words, e assim removidas do texto original para não causar erros de análise de palavras que são comuns para qualquer universo.
Com isso foi utilizado uma biblioteca em python chamada de “nltk.corpu”, que realiza a manutenção de “stop words” em diversas línguas.
Para realização de gráficos rápidos foi gerada uma classe auxiliar para criação da matriz de confusão. Com isso rapidamente após o treino de um classificador, era possível visualmente saber se estava no caminho correto.
Foi utilizado a página do scikit-learn (http://scikit-learn.org/stable/tutorial/machine_learning_map/) que auxilia na escolha do melhor classificador para cada caso, com isso foram analisados 4 tipos.

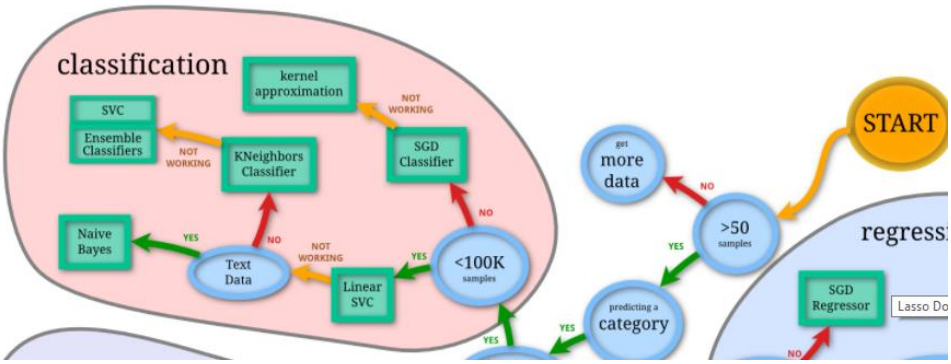

Imagem 4 : Scikit-learn :: Choosing the right estimator

Sendo assim foram analisados os métodos para realização da aprendizagem supervisionada:

- LogisticRegression
- GaussianNB
- NearestCentroid
- LinearSVC

4. Benchmark

Para análise dos resultados foram incluídas as matrizes de confusão, sendo assim de uma forma simples e fácil é possível entender se o método estava obtendo sucesso na predição.
Conforme já mencionado, se a média da matriz de confusão for de 80% já considero esse projeto um sucesso, pela quantidade de dados iniciais, além da quão abrangente análise de texto em um ambiente sem restrições como o twitter pode ser.

Methodology

1. Data Preprocessing

Como já descrito, após HTTP GET diretamente do twitter, o resultado foi incluído em um arquivo json, parser via código em .NET com limpeza de caracteres especiais e virgulas. A junção em um documento único foi feita manualmente. A limpeza por stop words utilizando python e remoção de endereços http para o máximo de limpeza no campo de texto.
Outro dado que precisou de pré-processamento foi o campo de data, já que estava em formato não reconhecido facilmente pelo datetime do python, então foi feito um parser manual para adequação desse formato.

2. Implementation

Para a implementação completa um dos requerimentos era a criação de uma API com end-point para consulta dos resultados dessa predição. Para isso foi utilizado o microframework Flask, utilizado para transformar códigos python em ambientes consultivos via web.
Outro diferencial da implementação foi o encapsulamento da predição final, para isso foi utilizado o serializador dill, que é um modulo baseado em pickle para serialização e deserialização de objetos. Com isso todo um ambiente produtivo consultivo desse algoritmo fica extremamente rápido e fácil de implantação.

3. Refinement

Todo o trabalho em refinamento de algoritmo parecia refletir em breves pontos do resultado na matriz de confusão, mas assim que era alterado o Randon state ou o tamanho da base de dados os refinamentos e tunnings não faziam tanto sentido, por isso todos classificadores foram utilizados em suas chamadas padrões.

Results

1. Model Evaluation and Validation

O classificador feito utilizando o método de Regressão Logística foi o que mais chamou a atenção devido a média de acertos que realizou na predição na base de teste, por esse motivo foi escolhido para representar o projeto, conforme mostra a Imagem 5.

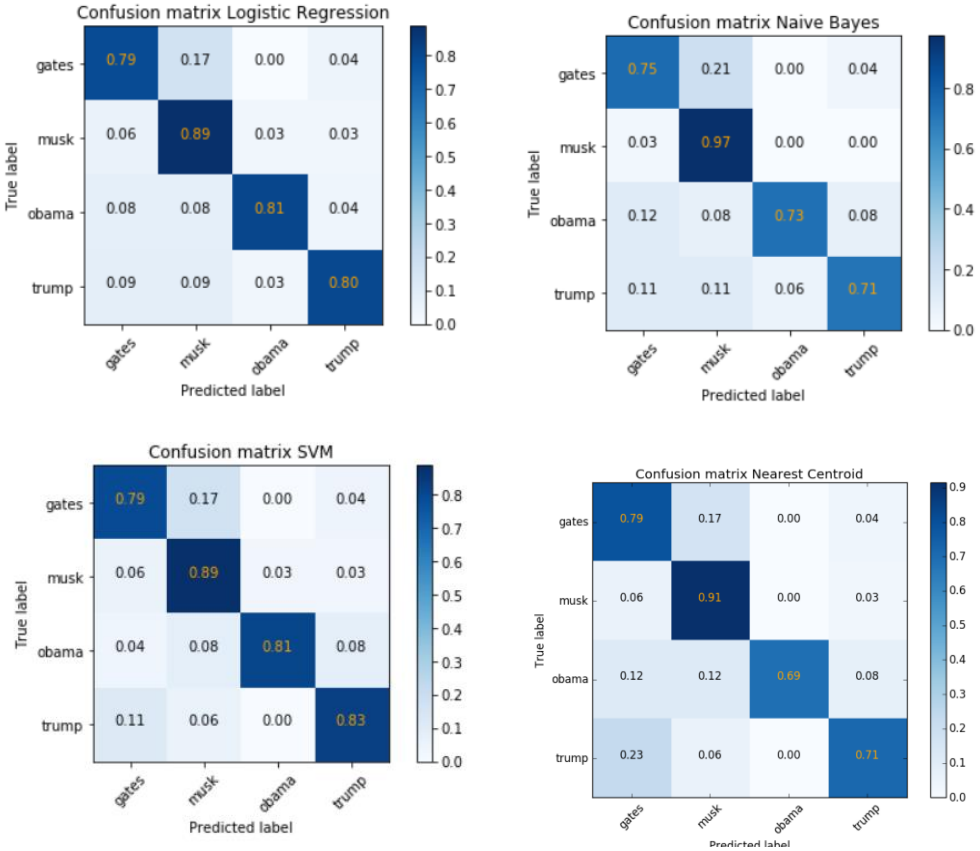

Imagem 5 : Matriz de confusão

2. Justification

Considerando a média de 80% de sucesso estabelecida e utilizando todo o cenário funcionando acredito que o que foi estabelecido antes da finalização do projeto foi alcançado, mesmo estando passível a diversas melhorias e implementações para melhoria.

Conclusion

1. Free-Form Visualization

Realizando um teste “ao vivo” com o último twitt de cada uma da celebridade (não presente no treino ou teste) para validar se a predição está correta:

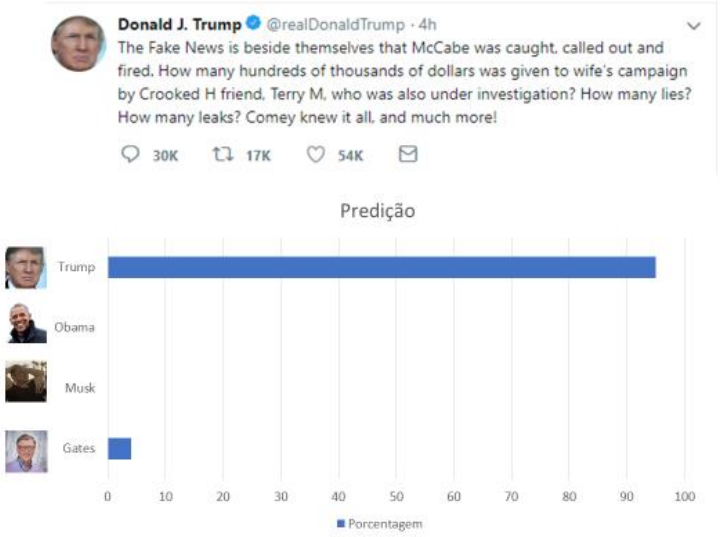

Imagem 6: Predição Donald Trump

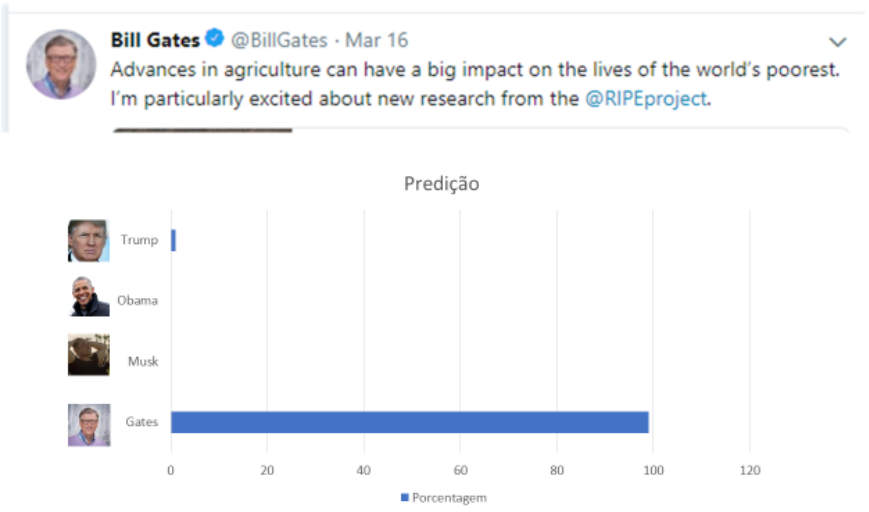

Imagem 7: Predição Bill Gates

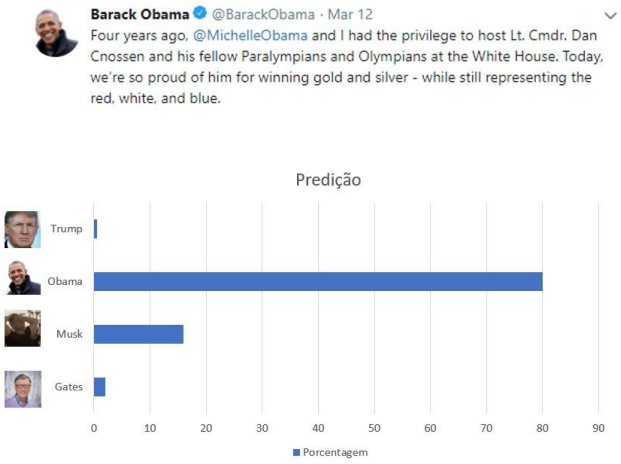

Imagem 8: Predição Obama

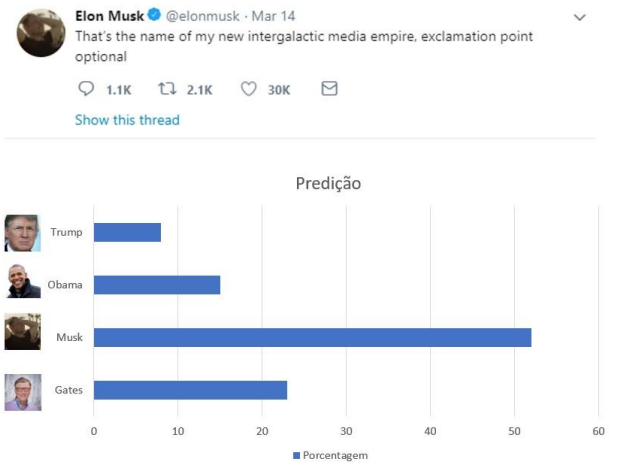

Imagem 9: Predição Elon Musk

Como já mencionado, com acerto em 80% não irá acertar todas, principalmente algumas mensagens sem muita informação ou palavras chaves usadas por outras celebridades. Como no exemplo a palavra "stars" é diretamente relacionada a Elon Musk, mas na verdade é uma mensagem de adeus de Obama para Stephen Hawking.

Imagem 10: Twitter Obama adeus a Hawking

2. Reflection

O aspecto mais complicado desse projeto foi a realização da ferramenta externa de stopwords, não a ferramenta em si, mas em uma forma de integrar a solução proposta.
A realização de criar um ambiente consultivo com flask e dill também exigiu toda uma diferença de particuliradidades e estudos para entender como esse ambiente deve funcionar de forma rápida e funcional.
Interessante pensar em como um estudo simples desse abre ideias para novas soluções, principalmente em relação a análise de texto e comparação com diversos cenários, e como esse algoritimo pode ser usado em todos esses cenarios com poucas mudanças.

3. Improvement

Para qualquer tipo de melhorias o primeiro passo seria automatizar a etapa inicial de recuperar os twitts via python para agilizar a etapa de preparação dos dados.
Para ficar mais interessante, incluir diversas celebridades no estudo, e não apenas as 4 escolhidas.
Para colocar essa ferramenta na web, colocar o flask em abiente produtivo, configurando com ngnix, supervisor e flask em paralelo para atender a esse ambiente consultivo.

References

- The top 100 people in Twitter : http://friendorfollow.com/twitter/most-followers/

- Flask : http://flask.pocoo.org/
- Obama Twitter : https://twitter.com/barackobama
- Trump Twitter : https://twitter.com/realdonaldtrump
- Elon Musk Twitter : https://twitter.com/elonmusk
- Bill Gates Twitter : https://twitter.com/billgates
- Twitter API : https://developer.twitter.com/en/docs
- Dill documentation : https://pypi.python.org/pypi/dill
- scikit-learn : http://scikit-learn.org/stable/
- Natural Language Toolkit : https://www.nltk.org/