



Universidade do Minho  
Escola de Engenharia

# Scripting no Processamento de Linguagem Natural

## Trabalho Prático 2 - Sentiment Analysis

### Grupo 9

Guilherme Martins PG47225  
Tiago Freitas PG47687



7 de junho de 2022

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Contextualização</b>	<b>2</b>
<b>3</b>	<b>Formulação do problema</b>	<b>3</b>
<b>4</b>	<b>Sentiment Analysis - Reviews de Apps do Google Play</b>	<b>4</b>
4.1	Recolha e tratamento de dados . . . . .	4
4.2	Classificador . . . . .	6
4.2.1	Preparação . . . . .	6
4.2.2	<i>Tokenization</i> . . . . .	6
4.2.3	<i>DataLoader</i> . . . . .	7
4.2.4	Classificador . . . . .	7
<b>5</b>	<b>Análise de resultados</b>	<b>9</b>
<b>6</b>	<b>Conclusão</b>	<b>11</b>

# 1 Introdução

No âmbito da unidade curricular de Scripting no Processamento de Linguagem Natural, foi-nos proposto um trabalho prático onde criássemos um classificador de "sentiment analysis" que é um NLP desenvolvido em Python baseado em regras e dicionários focado num contexto. Neste relatório vamo-nos focar em desmistificar toda a pequena matemática por detrás deste conceito e analisá-la. Após isso, iremos construir um classificador com os conhecimentos adquiridos.

## 2 Contextualização

A Análise de Sentimento é basicamente um classificador de texto que recebe uma frase ou uma lista de frases e descobre se o sentimento por detrás das frases é positivo, negativo ou neutro (que não é tanto usado para podermos estar perante um problema de resposta binária). Para além destas classificações, recentemente é possível também considerar sentimentos como pouco positivo ou pouco negativo.

Vejamos os seguintes exemplos:

- SPLN é uma boa unidade curricular. (sentimento positivo)
- SPLN não é uma boa unidade curricular. (sentimento negativo)
- SPLN é uma unidade curricular. (sentimento neutro)

As frases acima correspondem a observações sobre a unidade curricular de SPLN, no entanto cada uma delas transmite um sentimento diferente.

De um ponto de vista do *machine learning*, esta tarefa enquadra-se nas tarefas de *supervised learning*, uma vez que teremos de fornecer um conjunto de frases, rotuladas com o seu respetivo sentimento, ao modelo de *machine learning* e depois testaremos o modelo em frases ainda não processadas. Neste caso, foi usado um modelo pré-treinado da língua portuguesa, o que faz com, que nós não tenhamos que o "supervisionar", isto é, treinar o modelo, associando sentimentos a frases/palavras inicialmente.

### 3 Formulação do problema

Após surgir a ideia do tema a elaborar neste projeto, faltava apenas o alvo no qual nos iríamos debruçar, ou seja, o *input* que forneceria ao nosso programa para calcular o sentimento. Seria necessário arranjar texto que pudesse ir do melhor sentimento possível até ao pior e, assim, surgiram as opiniões/críticas. A forma mais simples e mais rápida de obter *reviews* está na aplicação que todos usamos que é o Google Play, no qual, todas as aplicações possuem as críticas publicadas pelos seus utilizadores juntamente com um *score*, também por eles atribuído.

O nosso objetivo consiste em calcular o sentimento de uma *review* e depois analisar esse valor, tirando conclusões em relação ao sentimento que pessoalmente atribuímos e, também em relação ao *score* atribuído juntamente com a crítica.

## 4 Sentiment Analysis - Reviews de Apps do Google Play

### 4.1 Recolha e tratamento de dados

Apesar de termos escolhido o Google Play como a fonte da nossa informação, necessitámos de nos focar numa certa área e num certo tipo de aplicações e, através do site *Mobile Action* encontramos as Top 10 aplicações de comida e bebida mais transferidos em Portugal e os seus respetivos ID's. Estes ID's serão úteis para podermos extrair os nossos dados com a ajuda do módulo *Google-Play-Scraper*.

#### Top 10 Food and Drinks Apps in Portugal

1. Burger King - Portugal  $\Rightarrow$  com.bk.pt
2. Uber Eats: Entrega de comida  $\Rightarrow$  com.ubercab.eats
3. McDonald's  $\Rightarrow$  com.mcdonalds.mobileapp
4. Too Good To Go  $\Rightarrow$  com.app.tgtg
5. TheFork - Restaurantes  $\Rightarrow$  com.lafourchette.lafourchette
6. Bolt Food  $\Rightarrow$  com.bolt.deliveryclient
7. Zomato Portugal  $\Rightarrow$  com.outdarelab.zomato
8. Moulinex, receitas e mais...  $\Rightarrow$  com.groupeseb.moulinex.food
9. H3  $\Rightarrow$  pt.yunit.mobile.android.h3
10. Telepizza Refeições ao Domicílio  $\Rightarrow$  com.telepizza

Figura 1: Top 10 aplicações comida e bebida - Portugal

Depois de instalar o *Google-Play-Scraper* (*pip install*), transferimos, filtrando pelas mais relevantes, o mesmo número de *reviews* correspondente a cada um dos 5 possíveis *scores*. No excerto de código que temos em baixo transferimos 50 *reviews* da aplicação Uber Eats.

```
apps_ids = ['com.bk.pt', 'com.ubercab.eats', 'com.mcdonalds.mobileapp', 'com.app.tgtg',
            'com.lafourchette.lafourchette', 'com.bolt.deliveryclient', 'com.outdarelab.zomato', 'com.
            groupeseb.moulinex.food', 'pt.yunit.mobile.android.h3', 'com.telepizza']

ids = [apps_ids[1]]

apps_reviews = []

for id in ids:
    for score in list(range(1, 6)):
        rvs, _ = reviews(id, lang='pt', country='pt',
                          sort=Sort.MOST_RELEVANT, count=10,
                          filter_score_with=score)
        for r in rvs:
            r['appId'] = id
        apps_reviews.extend(rvs)
```

Figura 2: Gerar *reviews*

Para obtermos resultados mais objetivos e para mitigar possíveis incoerências e desvios, decidiu-se ajustar os 5 possíveis *scores* em 3 classes (sentimento positivo, sentimento negativo e neutro) da seguinte forma:

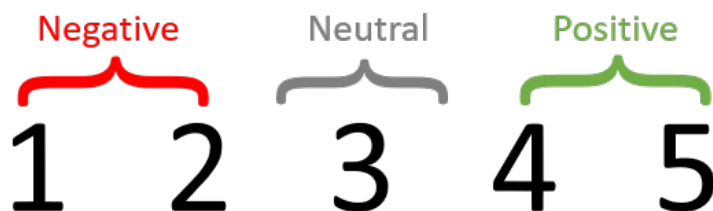


Figura 3: Sentimentos

De seguida, e após criar um novo atributo no nosso *dataset* (*sentiment*), guardamos estas informações num *csv* que servirá como *input* do classificador que desenvolvemos.

## 4.2 Classificador

Este projeto foi desenvolvido utilizando uma *framework* de *machine learning* o Torch (Pytorch) e o BERT (Bidirectional Encoder Representations from Transformers), que é basicamente um modelo de representação de linguagem desenvolvido por pesquisadores do *Google AI* e é uma técnica de Processamento de Linguagem Natural. O modelo BERT pode ser útil para identificar sentimentos de frases, que é o que pretendemos neste caso, e para adaptá-lo precisamos de criar um classificador e realizar algumas mudanças durante o treino do modelo. Com isto, nós fizemos uso do modelo BERT pré-treinado para a língua portuguesa.

### 4.2.1 Preparação

Já com o *csv* gerado importado no programa, começamos por converter o texto das *reviews* em *tokens* utilizando o *tokenizer* do BERT já adaptado à língua portuguesa, apenas com o objetivo de achar a *review* com o maior número e assim definirmos um tamanho máximo, o que vai facilitar o "verdadeiro" processo de *tokenization*.

```
token_lens = []
for txt in df.content:
    tokens = tokenizer.encode(str(txt), max_length=512, padding='max_length')
    token_lens.append(len(tokens))

MAX_LEN = max(token_lens)
```

Figura 4: Tamanho máximo

### 4.2.2 Tokenization

O processo de *tokenization* do BERT divide o texto em *tokens* associando cada *token* a um ID, utilizando *word embedding* e o vocabulário por nós inserido. Existem também *tokens* especiais dos quais se destacam:

- SEP (102) - Marca o fim de uma frase.
- CLS (101) - Deve ser colocado no início de cada frase para o BERT saber que se trata de um problema de classificação.
- PAD (0) - Tokens de valor 0 que devem ser adicionados às sentenças para garantir que todas tenham o mesmo tamanho.

Este *encoding* também retorna um conjunto de *attention masks* que serve, simplesmente, para sinalizar quais tokens são importantes, uma vez que o tamanho do *batch* calculado vai ser maior do que a maior parte do tamanho de cada uma das *reviews*, o que faz com que exista *tokens* PAD, que, no fundo, não são importantes, no final de cada até completar esse tamanho.





```

class SentimentClassifier(nn.Module):
    def __init__(self, n_classes):
        super(SentimentClassifier, self).__init__()
        self.bert = BertModel.from_pretrained(PRE_TRAINED_MODEL_NAME, return_dict=False)
        self.drop = nn.Dropout(p=0.3)
        self.out = nn.Linear(self.bert.config.hidden_size, n_classes)

    def forward(self, input_ids, attention_mask):
        _, pooled_output = self.bert(
            input_ids=input_ids,
            attention_mask=attention_mask
        )
        output = self.drop(pooled_output)
        return self.out(output)

```

Figura 7: Classificador

## 5 Análise de resultados

Os resultados obtidos são bastante bons, o que prova a eficácia do nosso modelo. Podemos destacar que quando o sentimento previsto não corresponde ao *score* atribuído pelo utilizador da aplicação, a maior parte das vezes acontece devido à incoerência do próprio utilizador entre o seu comentário e a sua avaliação. De seguida apresentamos alguns dos resultados obtidos para a aplicação Uber Eats.

- Review: O exemplo perfeito de desleixo e falta de profissionalismo. Quando entraram no mercado e eram os únicos no ramo, o serviço prestado era sem dúvida de qualidade. Atualmente, está péssimo, horrível, amador! Códigos promocionais de 8€ que não são aplicados, (quando no check out do total do pedido já é apresentado o desconto e o preço final com o mesmo) e depois debitam sempre esses 8€ que dizem que vão descontar. Isso é roubo, amigos. Menos um cliente. Vamos dar voz a quem realmente merece!  
Prediction: negative  
Real value: negative
- Review: Boa tarde,a entrega é boa porém achei horrível o fato de não conseguir mudar o número do cartão , ou seja se tenho outro cartão e quero utilizar não posso ? Tem a opção de excluir e editar mais o vosso aplicativo não deixa mudar . Então revejam esta situação porque é desagradável. Pois diz que não é possível excluir a forma de pagamento, já que só tenho uma cadastrada.  
Prediction: neutral  
Real value: negative
- Review: App problemático, cheio de erros, não fixa meu endereço, mostra um endereço errado, por mais que eu corrija, volta para o errado. Quando tento entrar pra escolher restaurante, da erro. Perda de tempo e desgaste. Há muito tempo deixei de usa pra fazer pedido em restaurante pela dificuldade e instabilidade. Horrível, Ainda insisti só pra me aborrecer. Chega!  
Prediction: negative  
Real value: negative
- Review: Tem funcionado bem, no entanto, julgo que fazem muitas promoções que depois não são aplicáveis ou aos restaurantes que preferimos ou nos produtos preferidos. Menus deveriam ter uma apresentação mais tátil com imagens, em vez de lista. Aplicação lenta e com dificuldade em cumprir prazos estipulados assim como falha no contacto directo com os parceiros, não tendo um método eficaz de resolução na hora de problemas relacionados com entregas  
Prediction: neutral  
Real value: negative
- Review: A aplicação é boa, as pessoas das entregas são super cuidadosas e simpáticas. Porém acho que devia haver mais cupões de desconto, para quem encomenda varias vezes do mesmo restaurante. A taxa de entrega esta cada vez mais caro. Podiam pensar pelo menos em um desconto por x vezes que se encomenda por exemplo. Assim o mais certo e deixa de usar a app, torna-se muito caro um simples lanchinho.  
Prediction: neutral  
Real value: neutral
- Review: A aplicação é muito boa, sim. Mas os preços de entrega de produto são ridiculos. Se uma pessoa quiser pedir algo barato, quase que paga mais no preço de entrega que no produto em si. E sim, eu percebo que tenha de haver um lucro por parte da uber e por parte do distribuidor, mas os preços

são excessivamente altos, e não é assim que trazem o "futuro" para as grandes massas.

Prediction: neutral

Real value: neutral

- Review: Numa altura de pandemia acho uma ótima opção e segura! Parece-me ser uma aplicação acreditada e de confiança. Contudo considero algumas taxas excessivas, chega a tornar-se impeditivo de pedir mais vezes ou determinados restaurantes por causa disso. É apelativa, mas de vez em quando tem quebras/-falhas no sistema que não me permite dar a gratificação aos estafetas.

Prediction: positive

Real value: positive

- Review: Já tenho encomendado várias vezes refeições e muitas vezes corre bem. No entanto já numa ocasião a encomenda foi cancelada sem explicação adicional. Outras vezes os restaurantes de onde tinha encomendado não estão disponíveis, estando abertos. O motor de busca é frustrante pois mostra tudo, se não houver o prete dido (por exemplo "mexicano": mostra em primeiro lugar os 2 que existem para a zona, mas seguidamente todos os outros restaurantes por ordem aleatória; se estes não estiverem disponíveis, mostra só restaurantes ao calhas). Hoje encomendei o jantar, cuja previsão de entrega foi sendo sucessivamente adiada perto da hora de entrega, acrescentando dois ou três minutos de cada vez, até se passar bem mais de uma hora. Não há qualquer hipótese de esclarecer a situação com ninguém, só respostas automatizadas e impessoais que não resolvem a situação. Foi, muito provavelmente, a última vez que encomendo com a Uber Eats

Prediction: negative

Real value: positive

- Review: Muitas opções diversificadas!!! Fácil de fazer os pedidos e são recebidos a tempo e em condições. Uso bastante!!!

Prediction: positive

Real value: positive

- Review: App bastante fácil a nível de navegação. Sempre prontos para ajudar o cliente quando algo corre menos bem.

Prediction: positive

Real value: positive

O programa leva a algum tempo a correr pois o grupo não se focou em otimizar o tempo que o modelo leva a prever os sentimentos das *reviews*.

## 6 Conclusão

Dado por concluído o último trabalho da Unidade Curricular de Scripting no Processamento de Linguagem Natural, o grupo encontra-se realizado e feliz com o trabalho desenvolvido.

Achamos que o trabalho serviu para nos esclarecer que a análise de sentimento resolve vários problemas de negócio, já que ajuda a prever o comportamento do cliente em relação a um produto, ajuda a testar a adaptabilidade de um produto, automatiza a tarefa de relatórios de preferência do cliente e, vendo o primeiro exemplo do nosso trabalho, pode automatizar facilmente o processo de determinar o desempenho de um filme analisando os sentimentos por trás das críticas do filme em várias plataformas.

Aprendemos também a trabalhar com ferramentas e modelos que manipulem a língua portuguesa, analisando sentimentos de frases através de um treino com avaliações reais de portugueses em aplicações da Play Store.