

Desenvolvimento e implementação do Workflow para realizar a análise de sentimentos como uma possível estratégia comercial

Mateus Borges Rodrigues
Graduando em Engenharia de Software – Uni-FACEF
mateusborges090@gmail.com

Victor Vinicius Sued Flávio
Graduando em Engenharia de Software – Uni-FACEF
visued@gmail.com

Profº Me Geraldo Henrique Neto
Docente – Uni-FACEF
geraldohenriqueneto@gmail.com

Resumo

Com a alta de crescimento de vendas e compras em sites de *e-commerce* do Brasil conforme pesquisas recentes, é comum um aumento no número de reclamações nos produtos adquiridos. Como cada vez mais consumidores utilizam os meios eletrônicos para realizar compras, também utilizam este meio para expressar a sua opinião. Desta forma, identificamos uma oportunidade de analisar o sentimento do consumidor através destas opiniões para que seja possível expressar graficamente o seu sentimento em relação a determinado produto. Com isso, temos uma importante estratégia de mercado para as empresas analisarem o nível de satisfação de seus clientes e para os consumidores visualizarem o nível de satisfação de pessoas que já adquiriram o mesmo produto.

Palavras-chave: Mineração de Dados, Processamento de Linguagem Natural, Análise de Sentimentos.

Abstract

With the high growth of sales and purchases on e-commerce sites in Brazil, according to recent surveys, an increase in the number of complaints about purchased products is common. As more and more consumers use electronic means to make purchases, they will use this means to express their opinion. In this way, we identified an opportunity to analyze consumer sentiment through these opinions so that it is possible to graphically demonstrate their sentiment in relation to a particular product. With this, we have an important market strategy for companies to visualize the level of satisfaction of their consumers and for consumers to also visualize the level of satisfaction they will have when purchasing the same product.

Keywords *Data Mining. Natural Language Processing. Sentiment Analysis.*

1 Introdução

A finalidade de um comércio eletrônico ou *e-commerce* é permitir que o consumidor faça compras de forma rápida e segura, sem a necessidade de se deslocar até uma loja física.

No Brasil, o *e-commerce* segue em expansão e no mês de dezembro de 2020, registraram alta de 53,83% em relação ao mesmo período de 2019, os dados são do índice MCC-ENET (Ecommerce Brasil, 2021).

Segundo dados do ReclameAqui, plataforma utilizada para reclamações contra empresas, produtos e serviços, na *Black Friday* de 2020 houve um total de 140.164 reclamações. O volume é 15,5% maior do que o registrado na edição de 2019 (dados computados até dia 31 de janeiro). O índice geral de solução por parte das empresas também cresceu, ficando em 77,2% (ReclameAqui, 2021).

Observando o crescimento desse mercado, iremos realizar um estudo para verificar a possibilidade de se utilizar a análise de sentimentos como uma estratégia comercial.

2 Referencial Teórico

Nesta seção do artigo abordaremos conceitos de tecnologias e temas que foram utilizados na construção do projeto, tais como a análise de sentimentos, mineração de dados, análise de dados, processamento de linguagem natural e a principal linguagem de programação utilizada no desenvolvimento, R.

2.1 Mineração de Dados

De acordo com Silva *et al.* (2016), a mineração de dados pode ser definida como um processo automático ou semiautomático de explorar analiticamente grandes bases de dados, com a finalidade de descobrir padrões relevantes que ocorrem nos dados e que sejam importantes para embasar a assimilação de informação, suportando a geração de conhecimento.

Apesar do termo mineração de dados ou *Data Mining* ser relativamente novo, antes da popularização da internet se utilizava os seus conceitos, porém havia uma grande limitação devido a tecnologia da época. Somente depois com a acessibilidade da internet e o grande volume de dados disponível que começou a surgir ferramentas capazes de tratar tais informações.

Segundo Laporais (2015), a mineração de dados nada mais é do que reconhecer padrões dentro de uma grande base de dados que sejam relevantes para tomada de decisões.

A mineração de dados pode ser aplicada em diversas situações, tal como a descrita por Silva *et al.* (2016, pg 33) onde temos um funcionário proativo de um restaurante que, além de somar as contas das mesas, também anotava os principais itens que eram consumidos por seus clientes. Desta forma, ele extraiu um padrão comportamental e com isso expressou a seguinte relação:

“[...] quanto maior o tempo de espera pelo preparo de um prato, maior o consumo de bebidas.” (SILVA *et al.*, 2016, pg 33).

Além disso ele observa outra regra derivada da cultura brasileira:

“[...] se filé à parmegiana é servido, então arroz branco e batatas fritas também são servidos.” (SILVA *et al.*, 2016, pg 33).

Tendo como base os dados de número de funcionários, humor dos clientes e dados históricos, através de um algoritmo computacional é possível projetar o valor das gorjetas que serão oferecidas. Existem inúmeros outros casos em que se identificam padrões e com o uso da mineração de dados consegue-se otimizar gastos e entregar melhores resultados para o negócio.

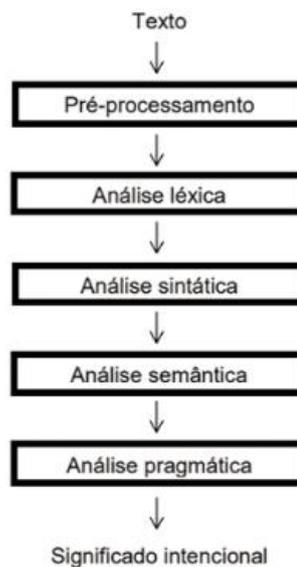
2.2 Análise de Dados

A análise de dados pode ser definida como um processo de refinar dados e transformá-los, para que quando concluído, obter informações relevantes para realizar tomadas de decisões. A análise de dados não é exclusiva da área de Tecnologia da Informação (TI) e muito menos algo novo, porém hoje com o apoio da tecnologia e a sua evolução constante é possível conseguir resultados muito mais eficientes em uma escala muito maior.

2.3 Processamento de Linguagem Natural

O Processamento de Linguagem Natural (PLN) ou do inglês, *Natural Language Processing* (NLP) é uma área dentro da inteligência artificial e seu objetivo principal é conseguir processar a linguagem natural utilizada pelos humanos para a máquina, servindo como uma espécie de tradutor para que seja possível a comunicação entre humano e máquina. Normalmente para que seja possível realizar esse processamento é necessário seguir algumas etapas conforme a apresentado na Figura 1 (DALE, 2010).

Figura 1 - Estágios básicos comuns de análise dentro do processamento de linguagem natural em textos.



Fonte: Adaptada de Dale (2010).

Para iniciar o processo, primeiramente é necessário um texto ou *corpus*. Com isso, é feito o pré-processamento do *corpus* e em seguida diferentes análises que vão extrair aquilo que realmente é útil, isso é necessário porque na linguagem humana existem diversos fatores que a podem tornar confusa e ambígua.

2.4 Análise de Sentimentos

Com a grande expansão das mídias sociais como *Facebook* e sua capacidade de gerar informação em larga escala tornando-se uma mina de ouro para empresas e para sociedade.

Segundo Benevenuto *et al.* (2015), as redes sociais são criação de uma revolução digital. Nela, é permitido se expressar, compartilhar ideias e opiniões sobre produtos ou pessoas.

Com isso, a análise de sentimentos também conhecida como Mineração de Opinião nos ajuda a identificar emoções e opiniões em textos, desenvolveu valor social muito importante tratando de problemas de classificação, sendo comumente utilizada para classificação de textos possuindo aplicabilidade em várias áreas.

Cientistas classificam a execução de análise de sentimentos como classificação de sentimentos, ou ainda, classificação de polaridade dos sentimentos, podendo ser positivo, negativo, neutro (LIU, 2012).

Liu, 2012, explica que a análise de sentimentos é pode ser definido como um conjunto de termos, sendo eles, objetos, componentes, opinião e polaridade. Análise de sentimentos aliada a técnica de mineração de dados torna-se uma ferramenta poderosíssima para empresas que procuram buscar na internet opiniões de pessoas sobre determinado produto que são produzidos por elas ou seus concorrentes podendo realizar tomadas de decisão mais assertivas e com isso agregar mais valor ao seu negócio.

2.5 Terminologias e Conceitos

Observando a popularidade desse tema, surgiram vários conceitos relacionados à análise de sentimentos, por exemplo a polaridade que demonstra o grau de positividade e negatividade de um texto que resulta em um binário (positivo ou negativo), ou ternário (positivo, negativo ou neutro), exemplificado por Benevenuto.

Por exemplo, a frase “Como você está bonita hoje” é positiva e a frase “Hoje é um péssimo dia” é negativa, já a frase “Hoje é 21 de Outubro” não possui polaridade e normalmente é classificada como neutra (BENEVENUTO, RIBEIRO, ARAÚJO, 2015).

Segundo Benevenuto (2015), para se representar a intensidade de um sentimento ou polaridade, normalmente devemos utilizar um ponto flutuante entre (-1 e 1), ou até entre $-\infty$ e $+\infty$, em muitos dos casos será necessário utilizar um *threshold* para conseguir identificar a neutralidade.

Há trabalhos que por exemplo medem a força de sentimentos nos títulos das notícias como o *Magnetic News* [Reis *et al.*, 2014] [Reis *et al.*, 2015b], capaz de separar eficientemente para o usuário notícias boas de notícias ruins (BENEVENUTO, RIBEIRO, ARAÚJO, 2015).

Para se representar o sentimento específico é verificado a presença de algumas emoções como raiva, surpresa, felicidade no texto existe alguns métodos que apresentam interpretações capazes de identificar qual sentimento a sentença apresenta.

Como exemplo a abordagem léxica Emolex (Mohammad and Turney, 2013), a qual é baseada a partir da avaliação de milhares de sentenças em inglês para 9 sentimentos diferentes: *joy, sadness, anger, fear, trust, disgust, surprise, anticipation, positive, negative* (BENEVENUTO, RIBEIRO, ARAÚJO, 2015).

Uma sentença se torna objetiva quando contém um fato ou informação, quando do contrário, podem expressar opiniões pessoais, alguns métodos utilizam análise de objetividade para verificar se é viável realizar a análise de sentimentos.

Cabe ressaltar que textos informais (ex.: coletados de redes sociais) tendem a ser mais subjetivos que textos formais (ex.: coletados de notícias) (BENEVENUTO, RIBEIRO, ARAÚJO, 2015).

Para que uma sentença seja objetiva ela precisa apresentar fatos ou informações, pois sentenças subjetivas apresentam opiniões pessoais e para que os resultados sejam satisfatórios é preciso que o conjunto de dados possuam mais sentenças objetivas que subjetivas.

2.6 A Linguagem R

A linguagem R é uma linguagem de programação destinada para análise e visualização de dados estatísticos. O nome deriva da inicial do nome de seus criadores, Ross Ihaka e Robert Gentleman e de uma expressão figurada com a linguagem S, que foi baseada para a criação do R.

R é um *software* livre e possui a licença *Free Software Foundations General Public License*, o que permite a sua redistribuição de forma gratuita e proíbe a sua comercialização.

Segundo Ihaka (2017) R é uma linguagem baseada em expressões:

- Usuários digitam a expressão no *prompt* do R .

- Essas expressões são avaliadas pelo interpretador do R.
- As expressões computadas são impressas para o usuário.
E é extensível:
- Usuários podem implementar novas funcionalidades em forma de funções.
- Desenvolvedores podem implementar pacotes que incluem novas funcionalidades, estendendo o que o R já oferece.

3 Análise do Problema e Implementação

Geralmente antes de iniciar o processo de análise de sentimentos o Cientista de Dados precisará estruturar os dados que serão trabalhados no processo de análise, e para isso ele precisa utilizar algumas técnicas de limpeza dos dados estruturando-o e melhorando a rotina de análise de sentimentos.

Nos tópicos a seguir, construiremos todos os passos seguidos para implementação do modelo de análise de sentimentos definido, aplicados a partir de uma base de dados constituída de comentários de produtos adquiridos por uma das principais plataformas de *e-commerce* do Brasil à Magazine Luiza.

3.1 Processo de Aquisição e Importação dos Dados

Foi realizada a captura dos dados de produtos e seus respectivos comentários do *e-commerce* da empresa Magazine Luiza, e para isso foi utilizada a técnica de *web scraping* que é muito comum quando se precisa acessar páginas HTML e extrair dados específicos dessa estrutura.

Para realizar esse trabalho utilizamos a linguagem *Python* e o *framework Scrapy*.

Com o apoio do *framework Scrapy* desenvolvemos um *Spider* com a função de acessar todas as páginas do *e-commerce* e extrair os dados necessários.

Um trecho de código Python, é apresentado na Figura 2, exemplificando uma função que tem como objetivo extrair dados do produto e os respectivos comentários, persistindo os dados em um sistema de gerenciamento de banco de dados *MySQL*.

Todo o código fonte está disponível no *Github* do projeto, o link está descrito no final do artigo nas Referências.

Figura 2 - Trecho de código Python

```
def extrair_dados(self, response, **kwargs):
    item = MagazineLuizaItem()
    categoria = response.xpath('..//nav/ul[@class="breadcrumb"]/li[3]/a/text()').get().strip()
    segmento = response.xpath('..//nav/ul[@class="breadcrumb"]/li[2]/a/text()').get().strip()
    nome_produto = response.xpath('..//h1[@class="header-product__title"]//text()').get().strip()
    codigo_produto = re.findall('^Código{.}', response.xpath('..//small[@class="header-product__code"]//text()').get().strip())[0]
    url_comentarios = 'https://www.magazineluiza.com.br/review/{}/?page=1'.format(codigo_produto.strip())

    item['codigo_produto'] = codigo_produto
    item['categoria'] = categoria
    item['segmento'] = segmento
    item['nome_produto'] = nome_produto

    requisicao = requests.get(url_comentarios, headers=self.headers)
    if(requisicao.status_code == requests.codes.ok and requisicao.headers['content-type'] == 'application/json; charset=utf-8'):
        for i in range(requisicao.json()['data']['pages']):
            requisicao = requests.get(url_comentarios[:-1]+str(i+1))
            if(requisicao.status_code == requests.codes.ok and requisicao.headers['content-type'] == 'application/json; charset=utf-8'):
                resultado = requisicao.json()['data']['objects']

                for obj in resultado:
                    item['nome_consumidor'] = obj['customer_name']
                    item['data'] = obj['date']
                    item['rating'] = obj['rating']
                    item['likes'] = obj['likes']
                    item['dislikes'] = obj['dislikes']
                    item['titulo'] = obj['title']
                    item['comentario'] = obj['review_text']
                    item['localizacao'] = obj['location']

    yield item
```

Fonte: Autores.

Após a extração dos dados eles foram inseridos na base de dados, e, na sequência, foi gerado um arquivo CSV (*Comma Separated Values*) para ser importado na ferramenta RStudio, conforme podemos visualizar na Figura 3 a seguir.

Figura 3 - Importando csv no RStudio

```
magazine <- read.csv(file = "c:/temp/magazineLuiza_original.csv", sep = ';',
                    dec = '.', encoding = 'UTF-8')[,c('id', 'codigo_produto',
                    'categoria', 'segmento', 'nome_produto', 'comentario')]
```

Fonte: Autores.

Posteriormente à realização do processo de carga dos dados na *IDE RStudio*, damos início a etapa pertinente ao Pré-Processamento.

3.2 Pré-Processamento

A etapa de pré-processamento é crucial para separarmos os dados relevantes daqueles não relevantes, isso porque no processo de obtenção, o computador não será capaz de filtrar os dados que serão insignificantes para o projeto. É nesta etapa que os comentários dos produtos de *e-commerce* serão decompostos e os termos menos significantes serão descartados. O resultado do pré-processamento é a *Bag of Words*, também conhecida como *BOW*. É uma matriz mostrando as palavras e a sua frequência.

3.2.1 Removendo Tags Indesejáveis

Nesta primeira etapa é necessário remover da estrutura os valores que não apresentam nenhuma informação, ou seja, valores nulos conforme demonstrado

na Figura 4 nas linhas 134, 136 e 137. Se não forem removidos, o resultado da análise poderá ser impactado negativamente.

Figura 4 - Valores nulos nos comentários

[134] ""

[135] "O produto chegou em perfeito estado e antes do prazo. Funcionamento perfeito até o momento."

[136] ""

[137] ""

Fonte: Autores.

A remoção dos valores nulos é relativamente simples, conforme demonstrado na Figura 5, que, por meio do operador lógico de negação, negamos a função *is.na* para retornar a coluna de comentários já com os valores não nulos dentro da variável *magazine_dataset*.

Figura 5 - Remoção de valores nulos dos comentários

```
magazine_dataset <- magazine[!(is.na(magazine$comentario) | magazine$comentario==""), ]
```

Fonte: Autores.

Com isso, não temos mais tuplas com valores ausentes, como demonstrado na Figura 6:

Figura 6 - Comentários nulos removidos

[134] "a minha chegou antes do prazo estipulado .entrega super rápida e a máquina é excelente em relação ao pega fiapos na descrição da máquina já dá pra saber que não vem antes de terminar a compra . simplesmente maravilhosa ótimo custo e super silenciosa dá pra usar qualquer hora e as roupas sai super limpinhas.."

[135] "a máquina em si e Boa mas a minha tem 6 meses e tá apresentando defeito tento contato com a magalu e não tenho retorno a espera de um retorno da magalu"

[136] "Ela é ótima! Quem tá reclamando só pode ter recebido um produto com defeito, pq ela lava bem, é bem grande, seca bem, fácil de usar, silenciosa. Uso ela sempre a noite e pela manhã só faço estender. Amo!"

[137] "O valor está completamente razoável pelo que ela entrega. Eficiente, gasta pouco, as roupas saem quase seca e sem muitas bolinhas."

Fonte: Autores.

3.2.2 Removendo Stop Words

As *stop words* são palavras que não possuem nenhum valor semântico na frase, ou seja, removendo-as não tiraria o sentido da frase, por exemplo a frase artigo sobre o *e-commerce* tem o mesmo sentido que artigo *e-commerce*, ou seja, mesmo removendo as palavras sobre e o, não tirou o sentido da frase.

Partindo dessa ideia, vamos remover dos nossos comentários essas *stop words*, conforme apresentado na Figura 7.

Figura 7 - Remoção das *stop words*

```
corpus_fogao_FI <- tm_map(corpus_fogao_FI, removeWords, stopwords_atualizada)
```

Fonte: Autores.

Utilizamos a função *tm_map*, do pacote *tm*, que trabalha com três parâmetros, o primeiro é o *corpus*, o segundo parâmetro é ação ou função de *callback*, que nesse caso será *removeWords* e o terceiro parâmetro é um *array* de *stop words* que extraímos da internet na linguagem português do Brasil.

3.2.3 Removendo Pontuações

As pontuações não terão relevância para nós pois não queremos visualizar a concordância textual dos comentários, portanto, também faremos a remoção das mesmas conforme podemos visualizar na Figura 8.

Figura 8 - Remoção da pontuação

```
corpus_fogao_FI <- tm_map(corpus_fogao, removePunctuation)
```

Fonte: Autores.

Utilizando a função *tm_map* da biblioteca *tm*, passamos o *corpus* do nosso texto *corpus_fogao* que são todos os comentários relacionados ao produto fogão e a função de *callback*, *removePunctuation*.

3.2.4 Removendo Números

É necessário também remover os números, sobretudo por eles não possuírem nenhuma informação relevante para o nosso processo, portanto, e a função responsável utilizada pela remoção da representação numérica, ora visualizado pela Figura 9.

Figura 9 - Remoção de números

```
corpus_fogao_FI <- tm_map(corpus_fogao_FI, removeNumbers)
```

Fonte: Autores.

Nesse processo também foi utilizado o método *tm_map* que recebe dois parâmetros, no primeiro parâmetro está recebendo o *corpus* e no segundo é a ação ou função de *callback* que nesse caso será *removeNumbers*.

3.2.5 Removendo Espaços em Branco

Ao executar as etapas anteriores é natural que sobre alguns espaços em branco entre as palavras.

Portanto, para os dados ficarem organizados adequadamente, realizamos o processo para remover os espaços indesejáveis, conforme apresentado pela Figura 10.

Figura 10 - Remoção de espaços em branco

```
corpus_fogao_FI <- tm_map(corpus_fogao_FI, stripwhitespace)
```

Fonte: Autores.

Mais uma vez foi utilizado o método *tm_map* recebendo dois parâmetros, o primeiro recebe o *corpus* e o segundo a função de *callback* ou ação a ser realizada que será *stripWhitespace*.

3.3 Criando uma Matriz de Frequência de Termos

Com a etapa de pré-processamento concluída, geramos uma representação com base na frequência de termos. A Figura 11 apresenta o uso da função “*TermDocumentMatrix()*” do pacote *tm* utilizada para atender esse propósito.

Figura 11 - Criando um documento de frequência de termos

```
corpus_fogao_tf <- TermDocumentMatrix(corpus_fogao_FI,  
                                     control =  
                                     list(minwordLength = 1,minDocFreq=1))
```

Fonte: Autores.

Esta função trabalha com dois parâmetros, no primeiro é informado o *corpus* que já foi estruturado na etapa 3.2 e no segundo parâmetro é implementado um filtro que permite selecionar a quantidade mínima de caracteres e a frequência mínima de palavras.

Para visualizar o resultado de termos *versus* frequência podemos atribuir o *corpus* na função *as.matrix* para transformar o documento em uma matriz e então utilizar as funções de *sort* e de *data.frame* para melhor visualização, conforme demonstrado na Figura 12.

Figura 12 - Criando matriz de frequência de termos

```
m <- as.matrix(corpus_fogao_tf)  
v <- sort(rowSums(m),decreasing=TRUE)  
d <- data.frame(word = names(v),freq=v)  
colnames(d) <- c('word','freq')
```

Fonte: Autores.

O resultado do processamento dos comandos da Figura 12 é demonstrado na figura 13:

Figura 13 - Matriz de frequência de termos

word	freq
lindo	462
super	399
recomendo	317
ótimo	311
prazo	280
bonito	267
comprei	242
excelente	240
qualidade	234
fácil	218
rápido	213
amei	186
chamas	180
limpar	173
gostei	172
rápida	158
perfeito	154
maravilhoso	136
expectativas	133
compra	123
satisfeita	107
porém	106
parabéns	99
atendeu	94

Fonte: Autores.

3.3.1 Text Mining

Nessa etapa, realizamos a estruturação e o agrupamento dos termos com o objetivo de gerar o *corpus*.

O processo de agrupamento dos comentários foi calcado exclusivamente pela identificação de determinados itens (produtos), conforme representado pela Figura 14.

Figura 14 - Agrupando por categoria

```
library(dplyr)
magazine_group <- magazine %>%
  group_by(categoria) %>%
  count(categoria, sort = TRUE) %>%
  filter(n >= 100) %>%
  ungroup()

magazine_grouped <- magazine_dataset[magazine_dataset$categoria %in%
  magazine_group$categoria,]
```

Fonte: Autores.

A Figura 15, apresenta a sequência de comandos utilizados para a criação de uma estrutura de dados adequada (mais conveniente) para o processo de análise de dados.

Figura 15 - Agrupando por categoria de produtos

```
fogao <- magazine_grouped[magazine_grouped$categoria %in% c("Fogão"),] # 3074
maquina_lavar <- magazine_grouped[magazine_grouped$categoria %in% c("Máquina de Lavar"),] # 1225
cooktop <- magazine_grouped[magazine_grouped$categoria %in% c("Cooktop", "Cooktop 5 Bocas"),] # 1188
geladeira <- magazine_grouped[magazine_grouped$categoria %in% c("Geladeira / Refrigerador"),] # 815
depurador <- magazine_grouped[magazine_grouped$categoria %in% c("Depurador de Ar"),] # 775

smart_tv <- magazine_grouped[magazine_grouped$categoria %in% c("Smart TV"),] # 1258
smartphone_samsung <- magazine_grouped[magazine_grouped$categoria %in% c("Samsung Galaxy", "Galaxy A31",
  "Galaxy A51", "Galaxy A10s",
  "Galaxy A01", "Galaxy A11",
  "Galaxy M31"),] # 602
smartphone_lg <- magazine_grouped[magazine_grouped$categoria %in% c("Celular LG", "LG K51s", "LG K8"),]
smartphone_motorola <- magazine_grouped[magazine_grouped$categoria %in% c("Motorola", "Moto E6S",
  "Moto G9 Play"),]
smartphone_iphone <- magazine_grouped[magazine_grouped$categoria %in% c("iPhone 6s e 6s Plus"),]
```

Fonte: Autores.

Na Figura 16, é possível visualizar as instruções responsáveis pela instalação dos pacotes utilizados para a criação do corpus.

Figura 16 - Instalando pacotes

```
install.packages('devtools')
install.packages("slam")
install.packages("NLP")
install.packages("tm", dependencies = TRUE, repos="http://R-Forge.R-project.org")
```

Fonte: Autores.

Já na Figura 17, é possível visualizar o comando responsável pela criação do *corpus* da categoria fogão, contemplando as principais marcas e modelos.

Figura 17 - Gerando *corpus*

```
corpus_fogao <- Corpus(VectorSource(tolower(fogao$comentario)))
```

Fonte: Autores.

3.3.2 Text Visualization

Posteriormente a geração da matriz de frequência de termos é possível gerar uma *Word Cloud*, ora visualizada pela Figura 18 referente ao produto fogão.

Figura 18 - Gerando uma nuvem de palavras do produto fogão (WordCloud).



Fonte: Autores.

Uma nuvem de palavras é uma representação gráfica da frequência de palavras em um texto, sendo que quanto maior a frequência da palavra, mais chamativa ela será no gráfico. Com isso, podemos concluir que para a categoria de produto fogão, os consumidores avaliaram de forma muito positiva.

4 Resultados e Discussões

Com a nuvem de palavras conseguimos ter uma boa visão sobre a percepção do consumidor em relação a determinado produto, podemos melhorar essa visão através da análise de sentimentos.

Para isso, utilizamos a biblioteca *syuzhet*. Através dela é possível obter *scores* para cada palavra que representa alguma emoção, essas emoções foram definidas de acordo com o modelo de Plutchik, criado pelo psicólogo com o mesmo nome, Robert Plutchik. Ele define que existem 8 emoções básicas: alegria, confiança, medo, surpresa, tristeza, expectativa, raiva e desgosto.

De acordo com Jockers, (2014), no site oficial do pacote *syuzhet* e com o dicionário de emoções definido por Mohammad, (2010), este pacote hoje suporta mais de 40 linguagens, incluindo o português, sendo traduzido com a ajuda do *Google Tradutor*. Na Figura 19, temos um pequeno exemplo de como a biblioteca consegue fazer uma análise de uma palavra:

Figura 19 - Utilizando a biblioteca *syuzhet*

```
get_nrc_sentiment("bonita", lang = "portuguese")
```

Fonte: Autores.

Utilizando a *string* “bonita” como primeiro parâmetro e a linguagem português, conseguimos obter o resultado demonstrado na Figura 20 ao executar a função:

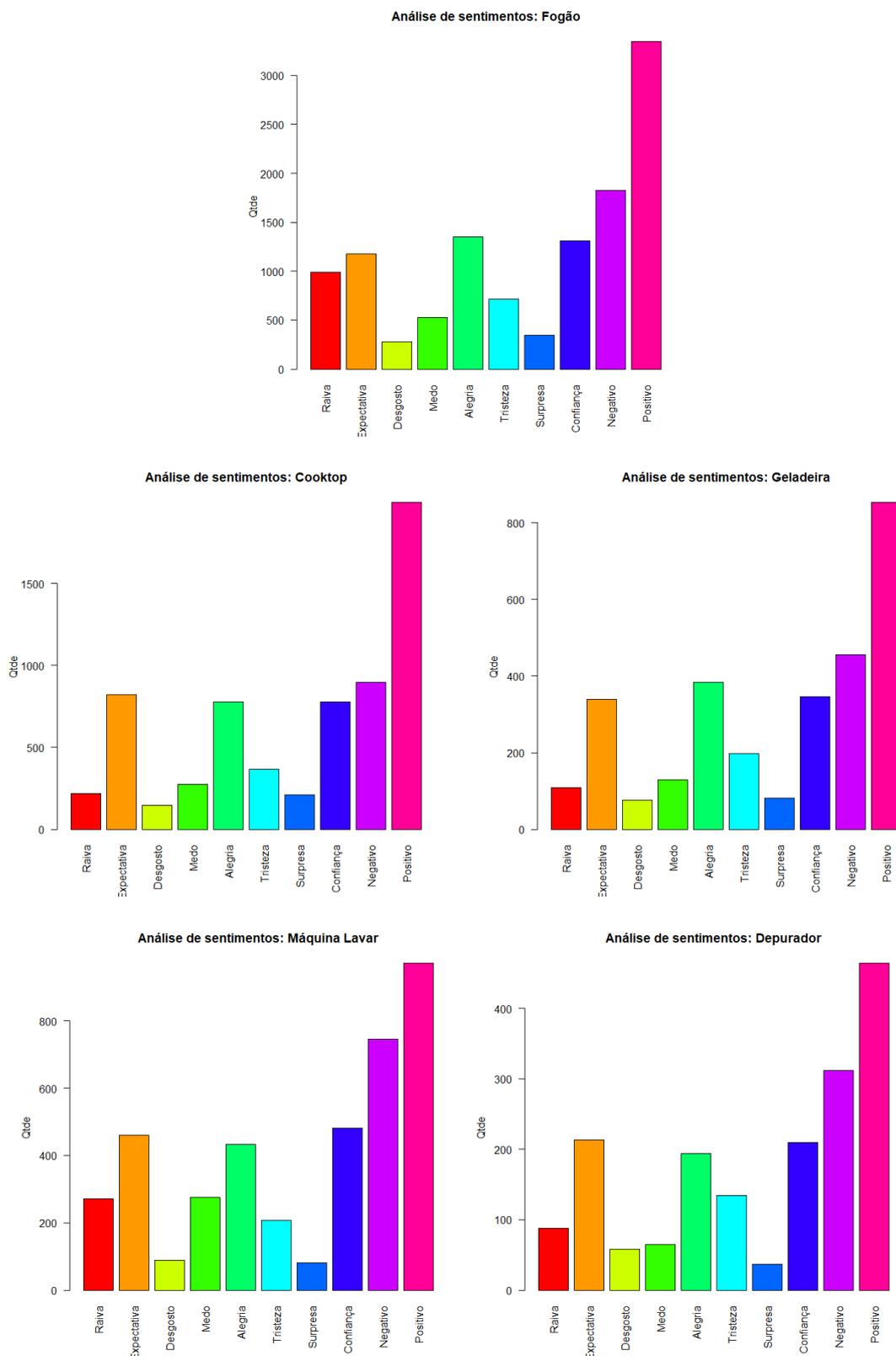
Figura 20 - Resultado retornado após execução da função *get_nrc_sentiment*

```
get_nrc_sentiment("bonita", lang = "portuguese")  
anger anticipation disgust fear joy sadness surprise trust negative positive  
0 1 0 0 2 0 0 1 0 2
```

Fonte: Autores.

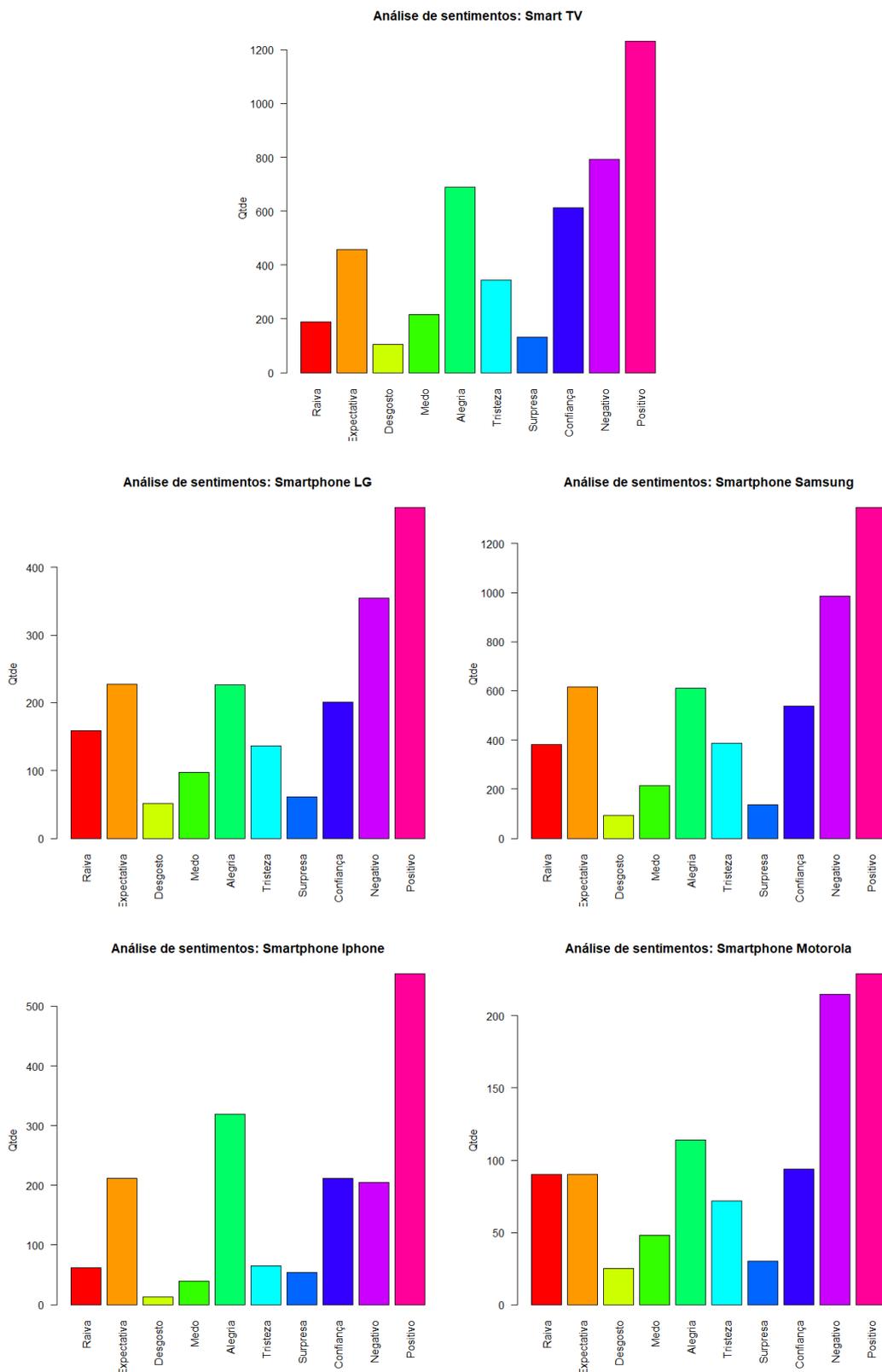
Com isso é possível perceber que a palavra “bonita” representa algo positivo e sentimentos como expectativa, alegria e confiança. A seguir nas Figuras 21 e 22, vamos mostrar os gráficos gerados com a ajuda da biblioteca de análise de sentimentos, para isso utilizamos os comentários para as categorias de produtos selecionadas (fogão, máquina de lavar, *cooktop*, geladeira, depurador, *smart tv* e *smartphones* das fabricantes LG, Samsung, Motorola e Iphone).

Figura 21 - Gráficos resultantes da análise de sentimentos de eletrodomésticos



Fonte: Autores.

Figura 22 - Gráficos resultantes da análise de sentimentos de eletrônicos

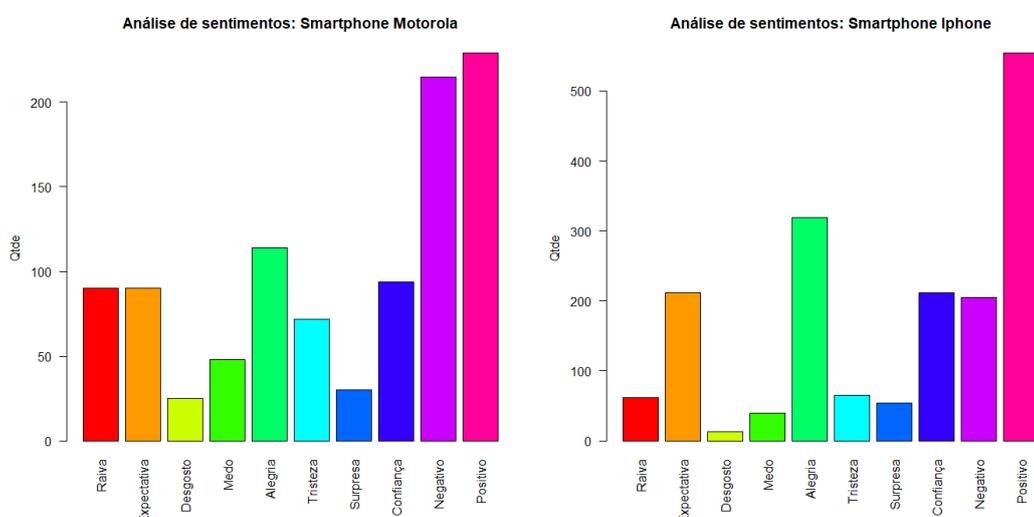


Fonte: Autores.

4.1 Análise de Sentimentos: *Motorola* versus *Iphone*

Observando o resultado da análise dos consumidores que adquiriram *smartphones* da marca *Motorola* e *Iphone* na Figura 23, podemos identificar que os consumidores do *Iphone* ficaram mais felizes e satisfeitos com a compra em relação aos que adquiriram *smartphones* da marca *Motorola*, porque as barras dos sentimentos Negativo e Positivo da marca *Motorola* quase se encontram, ou seja, houve quase que a mesma quantidade de consumidores que avaliaram positivamente ou negativamente o produto, já no *smartphone* da *Apple*, o *Iphone*, a barra do sentimento Negativo fica bem abaixo da metade da barra do sentimento Positivo, indicando uma avaliação bastante satisfatória acerca do produto.

Figura 23 - Comparação *Motorola* versus *Iphone*



Fonte: Autores.

5 Conclusão

Observar os resultados obtidos do processo de análise de sentimentos dos produtos mostrados anteriormente, pode auxiliar os consumidores na decisão de compra, por exemplo analisando o sentimento de quem comprou o *smartphone Iphone* da *Apple* a avaliação negativa é menor que a metade da avaliação positiva, claramente os consumidores que adquiriram este produto ficaram bastante satisfeitos com a aquisição.

Portanto acreditamos que esse tipo de análise pode auxiliar consumidores a visualizarem o sentimento de outros consumidores que adquiriram o produto que se deseja antes de efetuar a compra e assim atingir o nível de satisfação que esperamos obter após adquirir o produto desejado.

5.1 Implementações Futuras

5.1.2 Buscador

Com base no conhecimento obtido, idealizamos um buscador onde o consumidor poderia consultar, inserir o produto desejado, inclusive filtrar por marca, modelo, fabricante e assim visualizar os resultados da análise de sentimentos desses produtos que seriam baseados em comentários extraídos dos principais *e-commerce* do Brasil.

5.1.3 Análise Preditiva

Realizar a implementação de um modelo preditivo, que a partir de um comentário de um determinado produto, poderíamos prever o sentimento atual do consumidor.

Referências

BENEVENUTO, F.; RIBEIRO, F.; ARAÚJO, M. **Métodos para Análise de Sentimento sem Mídias Sociais.** Disponível em: <<http://homepages.dcc.ufmg.br/~fabricio/download/webmedia-short-course.pdf>>. Acesso em: 4 de abril de 2021.

DALE, R. **Classical approaches to natural language processing.** In: INDURKHYA, N.; DAMERAU, F. J. (Ed.) Handbook of natural language processing; Chapman & Hall/CRC; 2ª edição. 2010.

DIAZ, Gene. **Repositório com a coleção de stopwords utilizada no projeto.** Disponível em: <<https://github.com/stopwords-iso/stopwords-pt>>. Acesso em 10 de outubro de 2021.

ECOMMERCE BRASIL. **E-commerce brasileiro cresce 73,88% em 2020, revela índice MCC-ENET.** Disponível em: <<https://www.ecommercebrasil.com.br/noticias/e-commerce-brasileiro-cresce-dezembro/>>. Acesso em: 4 de abril de 2021.

IHAKA, Ross. **The R Project: A Brief History and Thoughts About the Future.** Disponível em: <<https://www.stat.auckland.ac.nz/~ihaka/downloads/Otago.pdf>>. Acesso em: 4 de abril de 2021.

JOCKERS. **Introduction to the Syuzhet Package.** Disponível em: <<https://cran.r-project.org/web/packages/syuzhet/vignettes/syuzhet-vignette.html>>. Acesso em 3 de outubro de 2021.

LAPORAIS, Bruno. **Técnica computacional pode auxiliar instituições em tomadas de decisão e criação de estratégias mais assertivas.** Revista PUC Minas, 2015. Disponível em: <<http://www.revista.pucminas.br/materia/mineracao-de-dados/>>. Acesso em : 25 de março de 2021.

LIU, B. **Sentiment Analysis and Opinion Mining, Synthesis Lectures on Human Language Technologies**. Morgan & Claypool Publishers, v. 5, n. 1, p. 1–167, 2012. Disponível em: <<https://www.morganclaypool.com/doi/abs/10.2200/s00416ed1v01y201204hlt016>>. Acesso em: 15 de abril de 2021.

MOHAMMAD, Saif. **NRC Word-Emotion Association Lexicon**. Disponível em: <<https://saifmohammad.com/WebPages/NRC-Emotion-Lexicon.htm>>. Acesso em 3 de outubro de 2021.

RECLAMEAQUI. **Black Friday 2020 encerra com mais de 140 mil reclamações e crescimento na solução dos problemas**. Disponível em: <https://noticias.reclameaqui.com.br/noticias/black-friday-2020-encerra-com-mais-de-140-mil-reclamacoes-e_4119/>. Acesso em: 4 de abril de 2021.

RODRIGUES, Mateus Borges; FLÁVIO, Victor Vinicius Sued. **Repositório contendo a base de dados e código-fonte realizados no projeto**. Disponível em: <<https://github.com/visued/Analise-Sentimentos-Ecommerce>>. Acesso em 10 de outubro de 2021.

SILVA, et al. **Introdução à Mineração de Dados - Com Aplicações em R**. BARUERI, GEN LTC; 1ª edição. 2016.