

Um Breve Ensaio Sobre os Sentimentos...

Dos Reviews do Spotify

Alisson Rosa

1 Introdução

O Spotify é um serviço digital que dá acesso instantâneo a milhões de músicas, podcasts, vídeos e outros conteúdos de criadores no mundo todo. Pode ser acessado pelo *browser*, sendo possível também baixar o aplicativo, estando disponível para diversas plataformas digitais. Nesse ensaio, vamos analisar os *reviews* feitos na Play Store sobre a versão para *Android*, dessa maneira vamos examinar o comportamento dos reviews (estrelas), junto com o texto deixado pelo autor do mesmo.

1.1 Uma visão Geral das Avaliações

Como é de conhecimento geral, erros geram muito mais comentários que acertos¹, dessa maneira, reclamações de problemas tendem a criar mais agitações que congratulações de acertos, nesse sentido, espera-se uma quantidade considerável de avaliações nota 1 do aplicativo. Pois os usuários de aplicativos são propensos a ter expectativas de sua utilização, entretanto, expectativas não satisfeitas, geram frustrações, e uma maneira de destilar tamanha frustração é lançar toda a fúria no *review* na Play Store. Começamos, portanto, avaliando pela Figura 1, a proporção das avaliações.

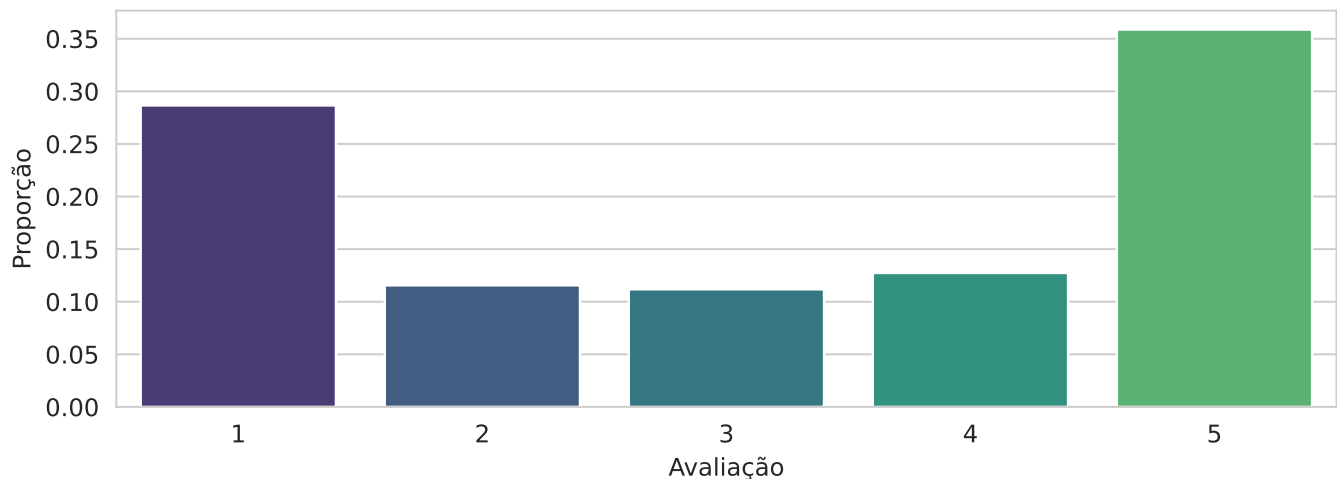


Figura 1: Proporção das Avaliações por Estrelas

¹Se você não dispõe de tal conhecimento, essa análise é a evidência pura e concentra disso.

Notamos assim uma disputa entre amor e ódio em relação ao aplicativo, portanto agora é importante refletir sobre alguns pontos:

- Existe algum padrão no uso de palavras para esses *reviews*?
- Há algum momento no tempo que há mais *reviews* “negativos” que “positivos”?
- É possível classificar o sentimento do autor somente baseado no texto?

Nas seguintes seções vamos aprofundar a análise desses *reviews* em relação ao tempo, também aprofundaremos ainda mais a análise em relação aos *reviews* considerando também o texto deixado pelo autor, dessa maneira, as respostas para essas perguntas vão surgir ao longo do texto.

1.2 Analisando os Review de Maneira Temporal

Vamos nessa subseção averiguar os *review* em termos de contagens e valores médios referentes, a mês, dia e hora.

1.2.1 Review Ao longo dos Meses

O que pode se notar pela Figura 2 é um comportamento elevado de *reviews* no mês 4 e o que acarretou em um decréscimo no valor médio dos *reviews* no mesmo mês.

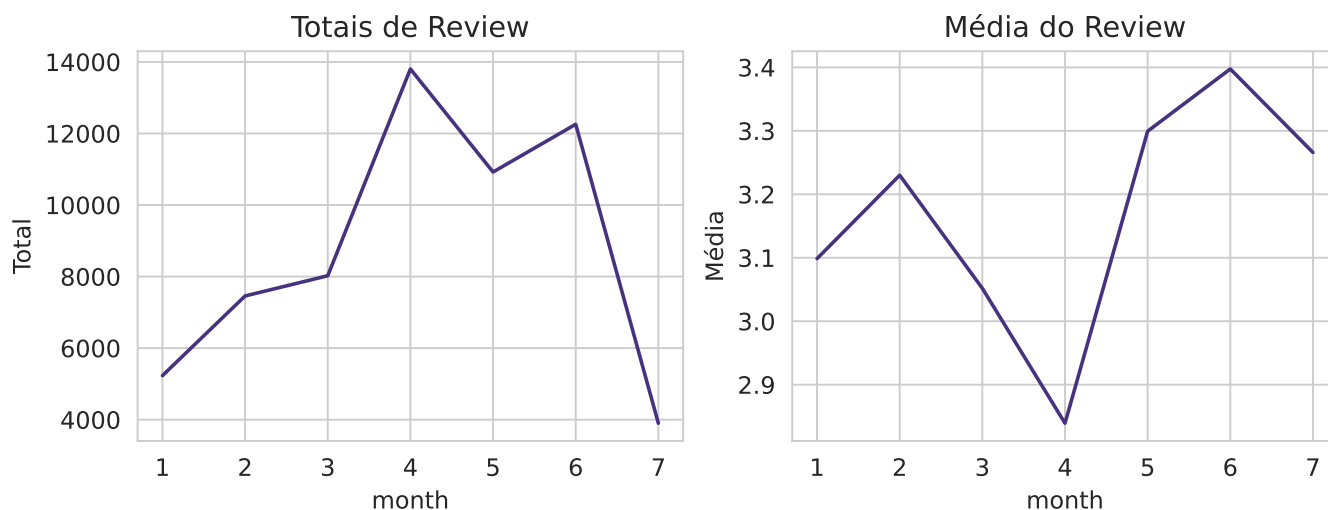


Figura 2: Comportamento dos Reviews ao longo dos meses

1.2.2 Review Ao longo dos Dias

Em relação aos dias temos um comportamento semelhantes aos meses, em que dias que possuem mais *reviews*, possuem avaliação média inferior aos demais, isso indica portanto que o ódio nasce quando as pessoas se reúnem².

²Ou é ao contrário?

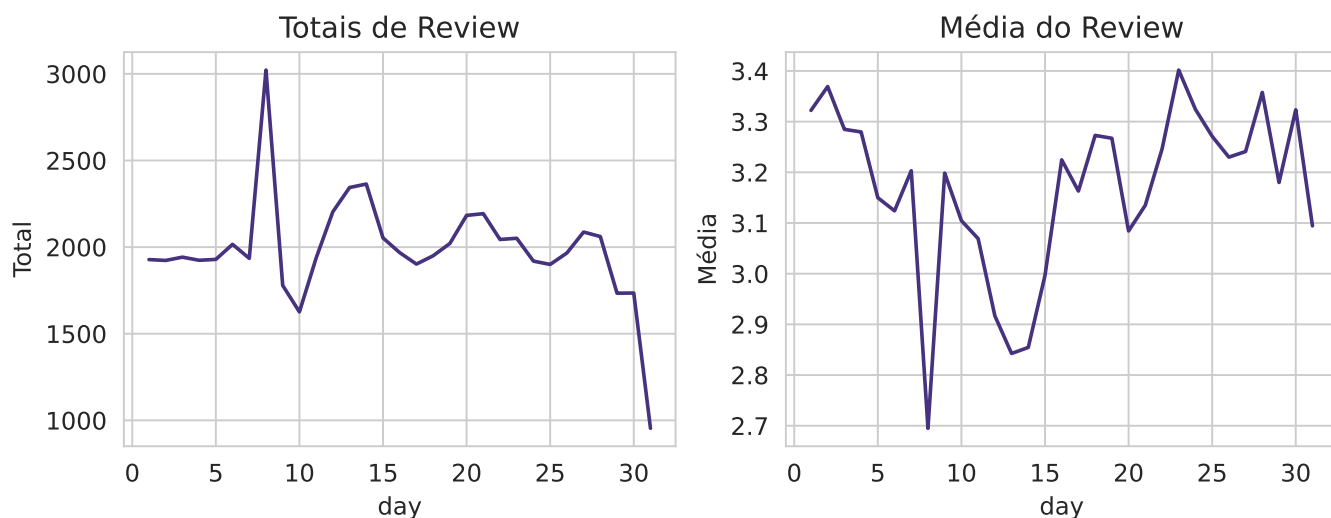


Figura 3: Comportamento dos Reviews ao longo dos dias

1.2.3 Review Ao longo das Horas

Em relação as horas temos uma grande quantidade de *reviews* no horário das 10-15h e depois o pico acontece as 18h, assim como o menor valor de *review* também acontece as 18h. Vale ressaltar que a média em relação as horas tende a manter-se estacionária até as 15h, um comportamento não visto na média dos meses e dias.

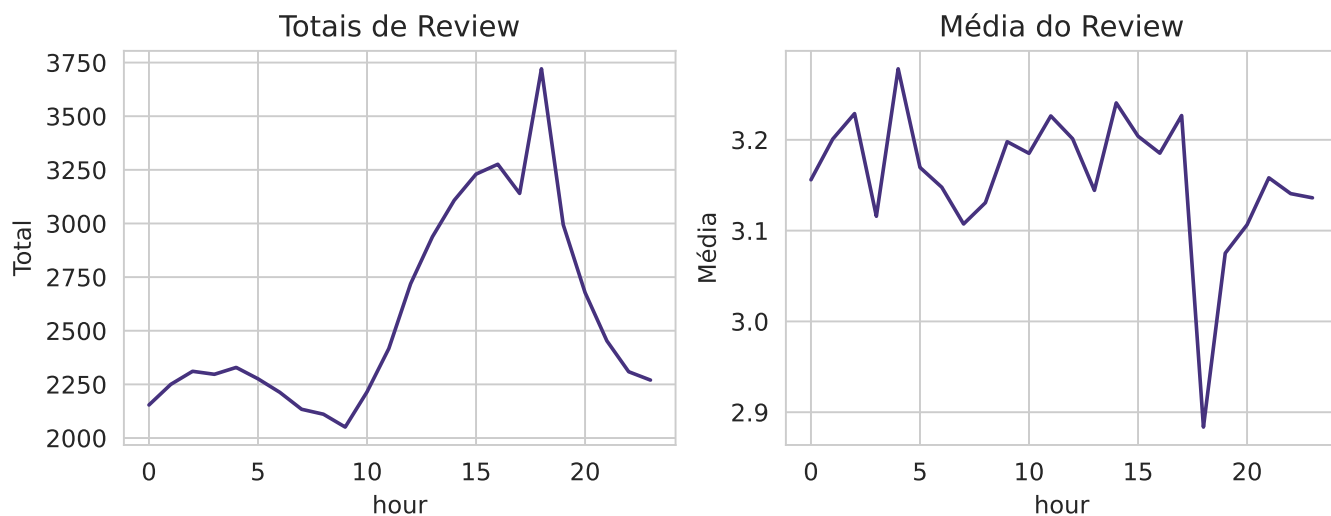


Figura 4: Comportamento dos Reviews ao longo das horas

2 Texto e seus ajustes

Em toda análise textual é necessária um bom pré processamento dos dados, portanto é isso que vamos tratar nessa seção. Realizaremos:

- Remoção de Stop Words³
- Lematização⁴
- Remoção de Espaços inde

Porém antes do processamento, vejamos o comportamento da frequência das palavras: Para o casos de 5 estrelas notamos palavras positivas como “love”, “great” e “easy” em destaque.



Figura 5: Wordcloud para Reviews de 5 estrelas

E para o caso de reviews de 1 estrela nota-se palavras referentes a problemas como “fix”, “even” e “issue”.



Figura 6: Wordcloud para Reviews de 1 estrela

Nota-se em ambos os casos que palavras como “Spotify”, “song”, “app” e “music” possuem uma alta frequência de exibição, entretanto não agregam muita informação para a análise, assim é interessante a remoção delas.

³Stop Words são palavras as quais possuem uma alta taxa de frequência no texto, mas carregam pouca informações para a análise a ser realizada.

⁴Tarefa de determinar se duas palavras tem a mesma raiz, apesar de diferença de “estrutura”

2.0.1 Um ajuste dos Reviews

Vamos aqui nessa subseção recategorizar a variável numérica da avaliação deixando da seguinte maneira:

- 1 a 2 Estrelas: Negativo
- 3 Estrelas: Neutro
- 4 a 5 Estrelas: Positivo

Portanto a partir desse momento, quando refere-se a review negativos, estamos nos referenciando a 1 a 2 estrelas e assim por diante.

2.1 Análise Sem as Stop Words

Nessa breve subseção remover as **stops words**, fazer algumas análises em relação aos *reviews* agora categorizados e também refazer algumas análises da seção anterior. Além das stop words “tradicionais” do inglês, serão removidas as seguintes palavras: app, song, music, e spotify.

Vejam os a título de curiosidade a wordcloud para *reviews* de 5 estrelas, agora sem stop words.



Figura 7: Wordcloud para Reviews de 5 estrelas sem Stop Words

Nota-se, agora um destaque para palavras positivas como: “love”, “great” e “good”.

2.2 Palavras em Review

Nessa subseção vamos averiguar a quantidade de palavras pelos *reviews*, será que *reviews* positivos possuem mais palavras que negativos? Será que o comportamento da quantidade das palavras se mantém constante ao longo do tempo? As respostas para essas perguntas e mais serão respondidas nessa seção.

2.2.1 Para Reviews Negativos

Para o caso de *reviews* negativos, notamos uma média de aproximadamente 38 palavras, um mínimo de 3 palavras, e um *review* que incrivelmente atinge 700 palavras!

Vale que salientar que apesar da existência de reviews bastante tagarelas, a quantidade de palavras tende a se comportar a sua maior parte até 100, como podemos ver tanto pela Tabela 1 e pela Figura 8.

Tabela 1: Análise descritiva para a quantidade de palavras em sentimentos negativos

	count	mean	std	min	25%	50%	75%	max
Total de Palavras	24771	37.754	23.9029	3	20	32	50	699

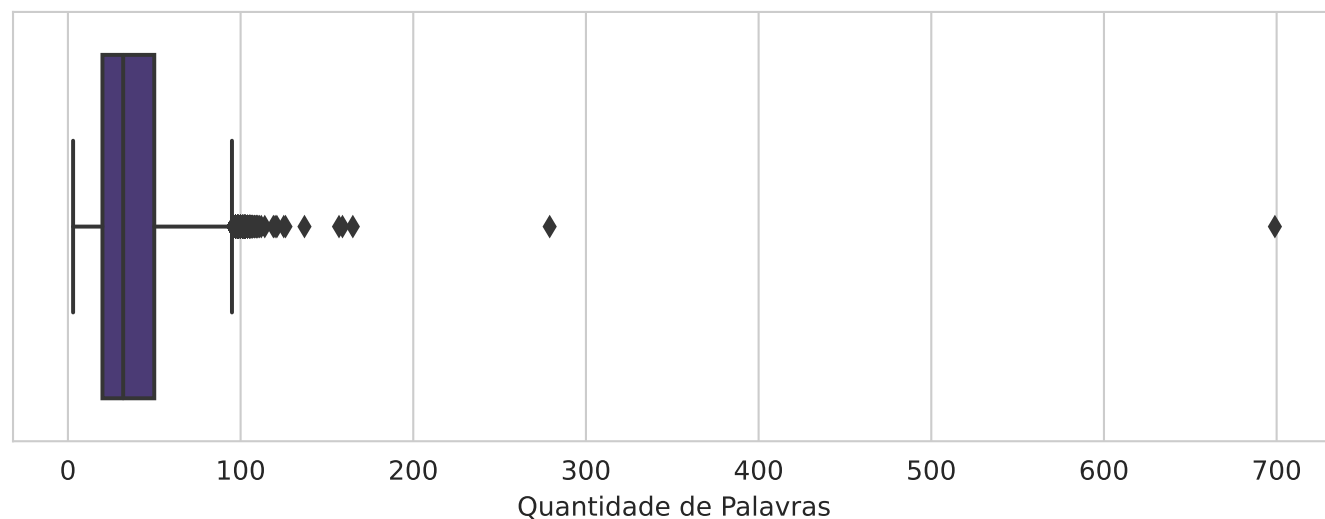


Figura 8: Quantidade de palavras para sentimentos negativos

2.2.2 Para Reviews Positivos

Referente aos *reviews* positivos temos um hábito bastante diferente do que os negativos, pois temos o quantil de 75% de somente 31 palavras, enquanto no negativo eram 50, a mediana é de 18 para casos positivos enquanto para negativos era de 32. Pela Figura 9 podemos reparar muitos *reviews* categorizados como *outliers* isso se deve ao fato de que a maior parte possui poucas palavras.

Tabela 2: Análise descritiva para a quantidade de palavras em sentimentos positivos

	count	mean	std	min	25%	50%	75%	max
Total de Palavras	29937	23.8083	19.1843	2	10	18	31	222

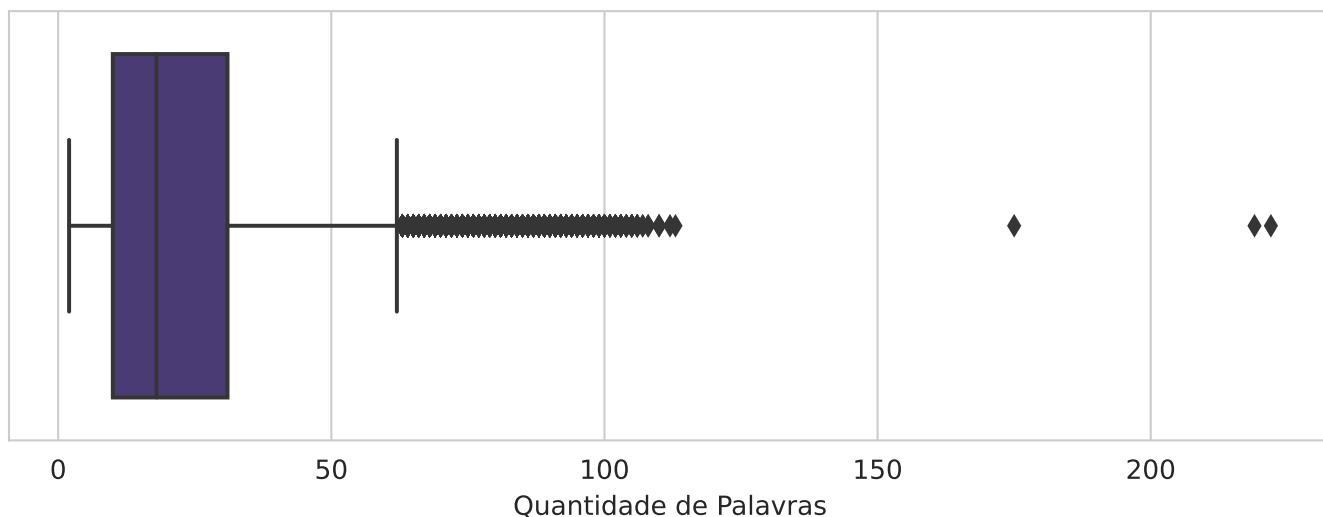


Figura 9: Quantidade de palavras para sentimentos positivos

2.3 Revisitando o tempo

Vamos nessa subseção averiguar o comportamento da quantidade de palavras por sentimento ao longo do tempo. Assim o que podemos notar pela Figura 10 é que os *reviews* classificados como positivos tem uma quantidade de palavras bem inferior aos negativos não somente no sentido global como visto pela Tabela 1 e Tabela 2 mas também no sentido temporal.

Em relação ao comportamento em relação aos dias, tem-se que para os *reviews* positivos visualmente uma estacionariedade em relação a quantidade de palavras, o que não acontece para o caso de *reviews* neutros e negativos. Para o caso dos meses é visível a não estacionariedade, tem-se que para meses iniciais do ano, os *reviews* possuem mais palavras que os situados no meses centrais do ano.

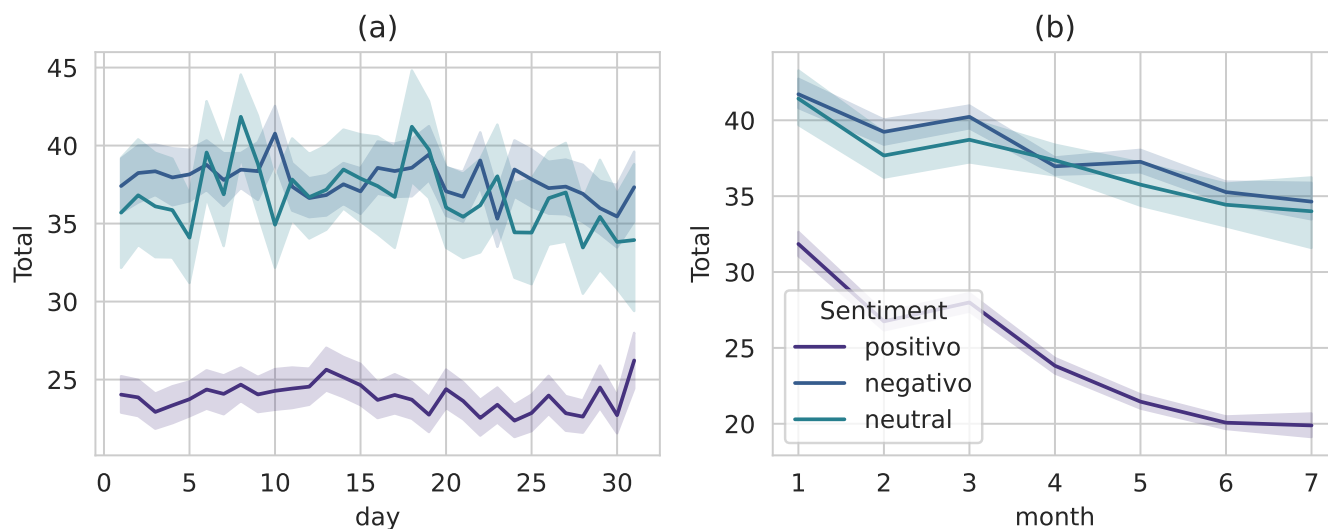


Figura 10: ?(caption)

3 Modelagem

Vamos nessa seção criar um modelo preditivo que fornece o sentimento do autor do *review* baseado no texto deixado pelo mesmo. Reiterando que um sentimento positivo nesse contexto, significa *review* numérico de 4 a 5 estrelas, e um sentimento negativo *reviews* de 1 a 2 estrelas na Play Store.

3.1 Desenvolvendo as Covariáveis

Como vimos, não temos covariáveis “automáticas” no **dataset**, assim faz-se necessário a criação baseando-se no *review* escrito, vamos aqui utilizar de duas técnicas, a saber Vetores de Contagens⁵ e TF-IDF. Dessa forma pretendemos avaliar o sentimento do usuário ao fazer o *review* escrito, por questões técnicas⁶ vamos utilizar somente os *reviews* que são positivos e negativos, dessa maneira removendo os classificados como neutros, tornando-se um problema de classificação binária. Vejamos pela Figura 11 a proporção de *reviews* agora categorizados como negativos e positivos.

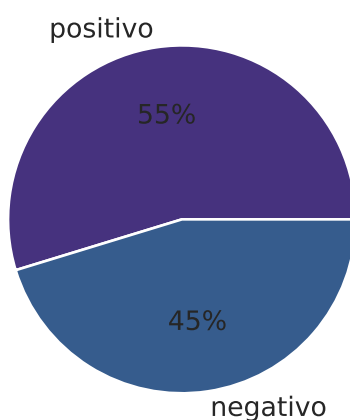


Figura 11: Proporção das Avaliações por Sentimento

Para a tristeza daqueles que tendem a inclinar-se a suspeitar mais do mal que do bem, temos agora uma quantidade levemente superior para casos de *reviews* de sentimentos positivos, dessa forma em termos de modelagem, é dito um problema de classes⁷ balanceadas.

3.1.1 Count Vectorization

A ideia é bastante simples, vamos criar uma coluna/vetor para cada palavra única no banco de dados e associar a quantidade de vezes que aparece no *review* escrito, um exemplo didático pode ser visto a seguir.

⁵Count Vectorization

⁶Técnicas aqui quer dizer: Não foi apresentado no curso a qual esse trabalho esta sendo apresentado

⁷Onde classes refere-se : Positivo e Negativo

Training examples	Features		
	love	programming	also
1 → I love programming	1	1	0
2 → Programming also loves me	1	1	1

Figura 12: Exemplo de CV

3.1.2 TF-IDF

O termo TF-IDF significa “Term Frequency - Inverse Document Frequency”, é uma técnica assim como Count Vectorizer (CV) para contar a quantidade de palavras em um documento⁸, ao contrário do CV, aqui utilizamos *scores* para cada palavra, que, em geral, refere-se a relevância dela no documento.

O método para computar o TF-IDF é feito multiplicando duas métricas:

- Quantas vezes a palavra apareceu no documento (tf)
- Inverso da frequência da palavra entre os documentos (idf)

Em termos formais temos: Seja w uma palavra, d um documento, e D o conjunto dos documentos⁹, o tf-idf fica definido como:

$$tfidf(w, d, D) = tf(w, d)idf(w, D)$$

3.2 Ajustando Modelos

Vamos aqui ajustar uma regressão logística e um modelo de random forest para classificar o sentimento do texto. Desse modo vamos ajustar 4 modelos, pois vamos ajustar também um modelo para CV e um modelo para TF-IDF

3.2.1 Modelos Com CV

Pela Figura 13 somos capazes de constatar que apesar de CV ser um método ingênuo, ele é capaz de criar covariáveis que conseguem prever com uma acurácia extremamente adequada nos dados de teste o sentimento do autor do *review* em ambos os modelos.

⁸Documento no contexto de NLP significa observação

⁹Também conhecido como corpus

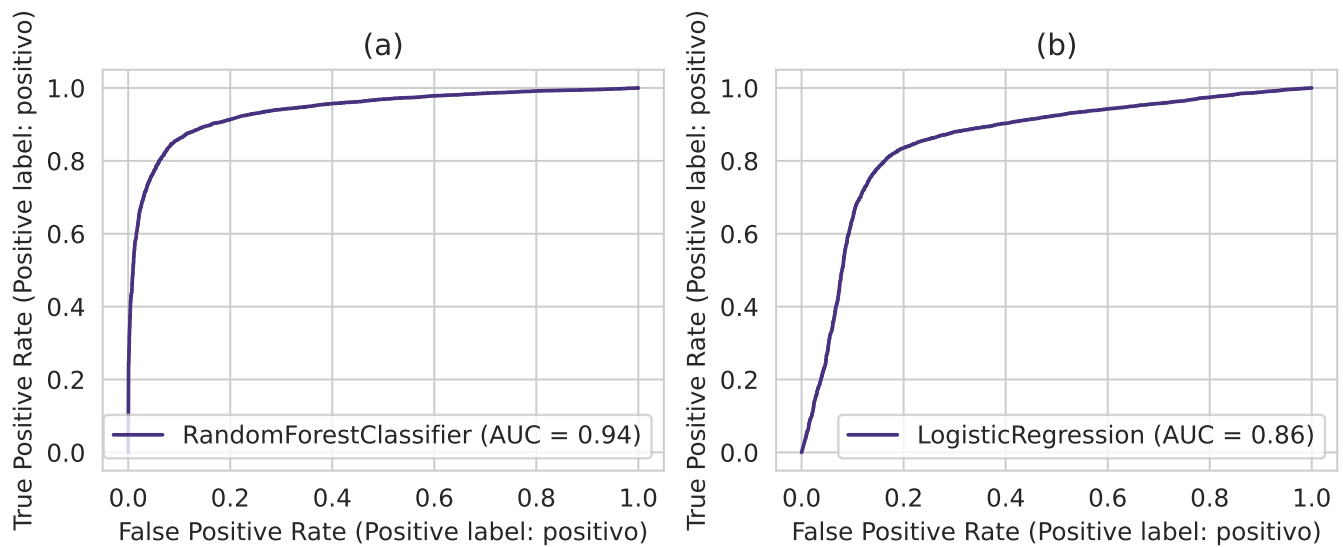


Figura 13: ?(caption)

3.2.2 Modelos Com TF-IDF

Apesar da técnica TF-IDF ser mais meticulosa que o CV, o seu esforço em criar covariáveis não condiz com a sua performance na detecção dos sentimentos, visto que em ambos os modelos, tem desempenho inferior aos modelos com CV.¹⁰

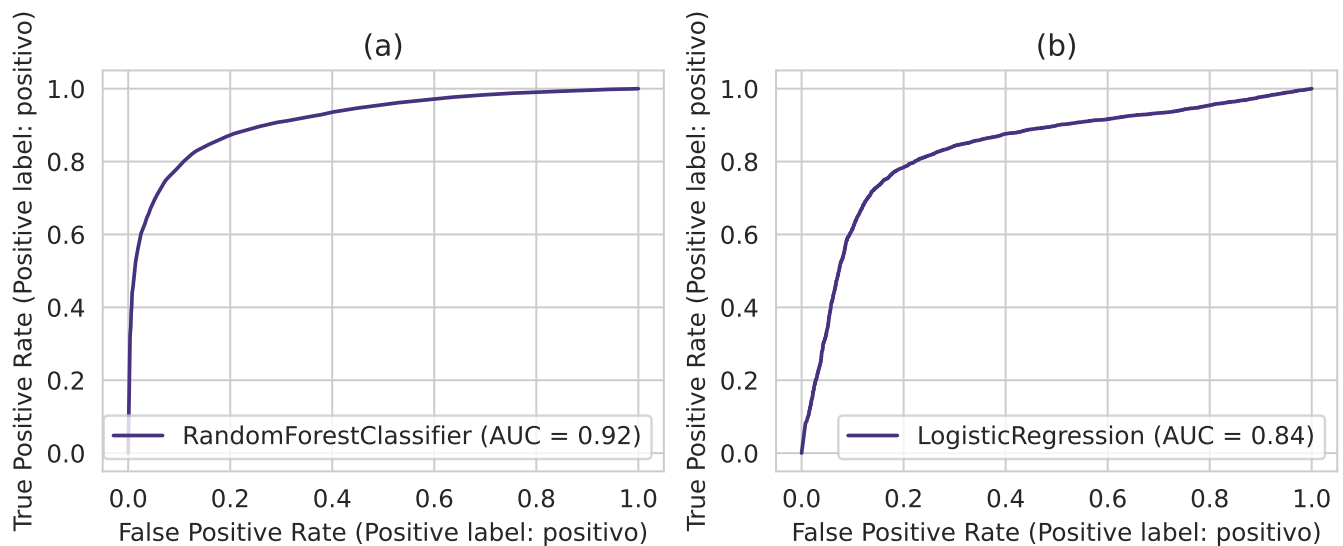


Figura 14: ?(caption)

¹⁰A realidade pode ser dolorosa, mas é algo que você deve aprender a aceitar.

3.2.3 Modelo Final

Finalizando, temos como modelo final uma random forest com pré processamento de CV, com 25% das observações sendo para teste, obteve aproximadamente uma acurácia de 88%.

4 O começo e o fim são um só

Um fato a ser considerado é que a existência do *review* escrito é dependente do review numérico, assim a análise anterior somente teria validade, se por algum motivo fosse possível deixar somente o review escrito, o que no momento não acontece.

Assim, é necessário extrair o sentimento do autor do texto **exclusivamente** pelo texto, pois dessa maneira podemos conflitar a predição do sentimento com relação a estrela deixada como *review*. Visto que existe um fenômeno tal qual o *review* numérico não condiz com o sentimento do autor, pois como já dizia Dostoiévski “A natureza das pessoas não são todas iguais; para muitas, uma conclusão lógica transforma-se às vezes num sentimento fortíssimo que domina todo o seu ser e que é muito difícil de expulsar ou transformar...”.

5 Epílogo

Vamos aqui introduzir um modelo mais robusto para analisar os sentimentos, vamos conflitar a classificação com o review numérico, e juntamente comparar com as predições do modelo anteriormente utilizado.

5.1 Modelo Vader

Valence Aware Dictionary for Sentiment Reasoning, ou Vader é um modelo de classificação de sentimentos semi-supervisionado onde é possível analisar ambas¹¹ polaridades de um texto, Vader é capaz de fornecer não somente o sentimento, mas também a intensidade do mesmo.

O modelo possui uma grande vantagem sobre os que foram utilizados aqui, pois não é necessário pré-processamento do texto e ele também consegue lidar bem com pontuações, emojis e palavras em caixa baixa/alta.

O Vader nos fornece a intensidade de sentimento positivo, negativo, neutro e também um escore denominado *compound* que normaliza¹² os três sentimentos anteriores em um só, sendo valores mais próximos de -1 indicando sentimento negativo e mais próximos de 1 sentimentos positivos.

Vejamos dessa maneira pela Tabela 3 a comparação da Random com CV e do Vader.

Tabela 3: Comparação dos Modelos

texts	Random Forest	Vader Resumido	Sentimento Real
0 Fix App!! It's messing again.	negativo	neutro	negativo
1 Fix dumb It 1 hour let skip	negativo	negativo	negativo
2 AWESOME great podcast. But got rid shows black screen.	positivo	positivo	positivo
3 I'm happy spotify. Music speaks soul & everything I need.	positivo	positivo	positivo

¹¹Positivo e Negativo

¹²A medida fica entre -1 e 1

	Random Forest	Vader Resumido	Sentimento Real
4 I recently joined let time full rating deserved	negativo	neutro	positivo

O que podemos notar é que em casos que o Vader classifica o *review* como neutro (*compound* = 0) a random forest classifica como negativo, e no ocorrências em que o Vader não classifica como neutro temos concordância entre os modelos.

Referências

- Albrecht, Jens, Sidharth Ramachandran, e Christian Winkler. 2020. *Blueprints for Text Analytics Using Python*. O'Reilly Media.
- Goyal, Chirag. 2022. «Roadmap to master NLP in 2022 NLP». *Analytics Vidhya*. <https://www.analyticsvidhya.com/blog/2022/01/roadmap-to-master-nlp-in-2022/>.
- Hapke, Hannes, Cole Howard, e Hobson Lane. 2019. *Natural Language Processing in Action: Understanding, analyzing, and generating text with Python*. Simon; Schuster.