Universidade Federal de Catalão
Instituto de Biotecnologia
Curso de Bacharelado em Ciência da Computação

Análise de Técnicas para Sumarização Automática de Opiniões na Área de Hotelaria

Paulo César de Morais Sousa

Catalão – GO
2023

Paulo César de Morais Sousa

Análise de Técnicas para Sumarização Automática de Opiniões na Área de Hotelaria

Monografia apresentada ao Curso de Bacharelado em Ciência da Computação da Universidade Federal de Catalão, como parte dos requisitos para obtenção do grau de Bacharel em Ciência da Computação.

Orientador: Prof. Dr. Sérgio Francisco da Silva
Co-orientador: Prof. Dr. Márcio de Souza Dias

Catalão – GO
2023

Ficha de identificação da obra elaborada pelo autor, através do
Programa de Geração Automática do Sistema de Bibliotecas da UFG.

Sousa, Paulo César de Morais
Análise de Técnicas para Sumarização Automática de Opiniões na Área de Hotelaria [manuscrito] / Paulo César de Morais Sousa. – 2023.
91 p.: il.

Orientador: Prof. Dr. Sérgio Francisco da Silva
Co-orientador: Márcio de Souza Dias
Monografia (Graduação) – Universidade Federal de Catalão, Instituto de Biotecnologia, Ciência da Computação, 2023.
Bibliografia.

1. Processamento de Língua Natural. 2. Mineração de Dados. 3. Comparação de Sumarizadores Automáticos. I. Silva, Sérgio Francisco da, orient. II. Dias, Márcio de Souza, coorient. III. Título.

CDU 004

RESUMO

SOUSA, P. C. M.. **Análise de Técnicas para Sumarização Automática de Opiniões na Área de Hotelaria**. 2023. 91 p. Monografia (Graduação) – Instituto de Biotecnologia, Universidade Federal de Catalão – , Catalão – GO.

Esta monografia apresenta uma comparação de diferentes técnicas voltadas à sumarização automática de textos, mais especificamente opiniões no ramo de hotelaria. São analisadas técnicas que geram um sumário baseado em aspectos, e também algumas que geram um sumário geral, com ambos os tipos de técnicas utilizando o modelo de sumarização extrativa. As opiniões em si foram extraídas de um corpus próprio, coletado da plataforma *TripAdvisor* e contendo opiniões sobre cinco hotéis de diferentes regiões do Brasil. Os sumários automáticos foram avaliados através do conjunto de métricas *Recall-Oriented Understudy for Gisting Evaluation* (ROUGE), com base em sumários criados por anotadores humanos. Os resultados indicaram algumas deficiências no conjunto ROUGE, e também apontam que algumas técnicas lidam melhor com partes específicas de sumarização de opiniões.

Palavras-chave: Processamento de Língua Natural, Mineração de Dados, Comparação de Sumarizadores Automáticos.

ABSTRACT

SOUSA, P. C. M.. **Análise de Técnicas para Sumarização Automática de Opiniões na Área de Hotelaria**. 2023. 91 p. Monografia (Graduação) – Instituto de Biotecnologia, Universidade Federal de Catalão – , Catalão – GO.

This monography presents a comparison of different techniques aimed at automatic summarization of textual content, more specifically, hotel reviews. Techniques that generate an aspect-based summary as well as techniques that generate a general summary are analysed, with both kinds of techniques following the extractive summarization model. The reviews were extracted from a novel corpus with data collected from the TripAdvisor platform and containing opinions on five hotels from different regions in Brazil. The automatic summaries were evaluated through the ROUGE set of metrics, based on summaries created by human annotators. The results indicate some deficiencies in the ROUGE set, and also show that some techniques handle certain portions of opinion summarization better than others.

Keywords: Natural Language Processing, Data Mining, Automatic Summarizer Comparison.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

BERT	*Bidirectional Encoder Representations from Transformers*
CNN	*Convolutional Neural Network*
CNNt	*Cable News Network* (não confundir com *Convolutional Neural Network*)
DM	*Daily Mail*
DUC	*Document Understanding Conference*
HTSS	*Highest Term Sentence Score*
JSD	*Jensen–Shannon Divergence*
LCS	*Longest Common Subsequence*
LDA	*Latent Dirichlet Allocation*
MALLET	*Machine Learning for Language Toolkit*
MMR	*Maximal Marginal Relevance*
NBC	*Naive Bayes Classifier*
NGD	*Normalized Google Distance*
NYT	*The New York Times*
PAM	*Partitioning Around Medoids*
PCA	*Principal Component Analysis*
PLMMR	*Probabilistic Latent Maximal Marginal Relevance*
PLN	Processamento de Língua Natural
PSVM	*Probabilistic Support Vector Machine*
ROUGE	*Recall-Oriented Understudy for Gisting Evaluation*
SDSS	*Standard Deviation Sentence Score*
STSS	*Summing Term Sentence Score*
SVM	*Support Vector Machines*
TAC	*Text Analysis Conference*
TF-IDF	*Term Frequency-Inverse Document Frequency*
VAE	*Variational AutoEncoder*
WLCS	*Weighted Longest Common Subsequence*

SUMÁRIO

CAPÍTULO

1

INTRODUÇÃO

A sumarização automática lida com a análise de uma ou mais fontes de informação, de onde extrai o conteúdo mais pertinente de acordo com o tema abordado, e o apresenta de uma forma condensada e voltada às necessidades do usuário (MANI, 2001). Esse conteúdo pode variar conforme o tópico abordado, e o interesse do usuário. Assim, é possível gerar mais que um único sumário para um mesmo conjunto de dados base.

Para que a sumarização automática seja de fato efetiva, um corpus bem representativo (que contém informações amplas e claras sobre um determinado tópico) e relacionado ao material a ser sumarizado se faz necessário. Com um corpus em mãos, dá-se início à tarefa de sumarização dos textos base, que segundo Nenkova e McKeown (2011), pode ser feita de duas formas: extrativa e abstrativa. A forma extrativa foca apenas em extrair as sentenças mais informativas dos textos originais, enquanto que a abordagem abstrativa faz uma reescrita do conteúdo extraído para gerar um sumário mais conciso. O sumário resultante pode ser classificado em uma de três categorias: indicativo, se possuir apenas os tópicos essenciais; informativo, caso tenha todas as informações principais e atue como um novo texto; e crítico, se este adicionar críticas em relação ao conteúdo gerado (CONDORI, 2014).

A importância da sumarização automática de opiniões vem aumentando bastante devido também ao volume de opiniões em diversos *websites* seguir um ritmo de crescimento acelerado nos últimos tempos. Trabalhos tais quais (SILVA; FILHO, 2014) e (CORTEZ; MONDO, 2018) demonstram que as análises e opiniões em *sites* de viagens e hotelaria são de suma importância para a decisão dos usuários destas plataformas, devido ao aumento do número de pessoas no mundo digital. Conforme a quantidade de pessoas *online* aumenta, o mesmo ocorre com o nível de uso destes serviços virtuais e consequentemente a quantidade de análises presentes tende a crescer com o tempo, o que implica em um aumento exponencial no volume de dados a serem analisados, visto que opiniões antigas ainda permanecem nestes *sites*.

Este crescimento do volume de dados torna a leitura manual de cada uma das opiniões

impraticável, o que resulta na incessante busca por bons sumarizadores automáticos para gerar resumos das análises presentes nestes sites. Portanto, esta pesquisa realizará um estudo de diferentes algoritmos e modelos de sumarização de *reviews* de hotéis a fim de comparar os resultados gerados por cada uma delas.

Essa sumarização automática de opiniões no ramo de hotelaria se faz importante por dois principais motivos: a possibilidade de uma verificação mais eficiente das opiniões sobre um hotel por parte dos gerentes para que consigam melhorá-lo, e a maior facilidade de se obter uma opinião embasada sobre este mesmo hotel por parte do consumidor. Isso ocorre porque ambos agora não terão que lidar com várias informações avulsas e redundantes, e sim, com um único resumo geral delas.

Entretanto, esse tipo de sumarização tem certas dificuldades a serem enfrentadas, como o fato de opiniões serem textos subjetivos e informais por natureza, utilizando bastante a linguagem coloquial (LIU, 2012), o que contrasta com textos jornalísticos que são inerentemente formais e usualmente trazem informações objetivas. Por isso, lidar com a informalidade dos textos e a subjetividade de algumas informações (ver Figura 1) são aspectos que precisam ser levados em consideração durante o desenvolvimento. Outra dificuldade é a quantidade de redundância normalmente encontrada ao se realizar uma sumarização multi-documento, ou que apenas aborda múltiplos textos relacionados a um mesmo tópico. Isso acontece porque diferentes opiniões ainda possuem informações bastante similares, sendo necessário utilizar um algoritmo ou método para identificar e reduzir a quantidade de palavras repetidas, tal como o de Patel, Shah e Chhinkaniwala (2019).

Figura 1 – Exemplo da informalidade e subjetividade de dois textos opinativos.

Atendimento bom e almoço saboroso
Muito bem atendita e recepcionada.
Quarto silencioso.
Água quentinha do chuveiro.
Tudo bem higienizado
Me chamou atenção ter de fazer o pagamento do restaurante a cada pedido na hora da entrega.
E também nao poder marcar na conta do quarto o consumo no bar do hall principal.
Mas isso é apenas um detalhe.

Para ser hotel 5 estrelas precisam melhorar muito desde atendimento ao cliente até as estruturas
Tivemos muita problemas em nossa estadia um casal e 3 crianças, primeiro não tem ar condicionado
quente e frio, toalhas de banho quase transparentes de tão velhas, pé da cama quebrou tivemos
que tirar inspires pés para que uma das crianças pudesse dormir, no restaurante do hotel não
conseguiram nos acomodar para um jantar por sermos 5 pessoas queriam nos colocar em uma mesa no
bar isso porque ficaram nos enrolando para nos atender e por último peguei 2 quartos e não nos
deram o conjugado pior foi ficar em um quarto conjugado que estavam outro casal e seu filho e
para nós eu pedi quarto conjugado e disseram que não seria possível lembrando que meus filhos
ficaram em outro quarto que tinha porta conjugada também.
Resumo tudo muito ruim experiência hoteleira péssima detalhe era aniversário do meu filho do
meio no dia da jantar no hotel.
Hotel em reforma também

Fonte: Corpus do autor

Na Figura 1, note que um mesmo elemento pode ser avaliado de forma positiva ou negativa (destacado em verde em vermelho) conforme a opinião, isto ocorre porque opiniões sobre hotéis variam largamente a depender de fatores como quarto e data. Alguns dos erros gramaticais cometidos pelos autores das opiniões também foram realçados em amarelo para exemplificar a informalidade destes textos.

Atualmente já existem vários sumarizadores para lidar com situações como esta, e que geram resumos com diferentes níveis de informatividade, clareza, redundância, dentre outros. Isso se dá devido à ampla gama de técnicas existentes: *text matching* (ZHONG *et al.*, 2020), grafos (ERKAN; RADEV, 2004), aprendizado de máquina (SCHILDER; KONDADADI, 2008), informação estatística (MCCARGAR, 2004), mapas de relacionamento (RIBALDO; PARDO; RINO, 2011), entre outras.

1.1 Problema de pesquisa

Este trabalho compara uma série de algoritmos de sumarização aplicados ao contexto de *reviews* de hotelaria. Por se tratar de uma análise comparativa, este trabalho aborda uma série de técnicas diferentes que pertencem à área de sumarização de textos, indo desde técnicas mais simples baseadas em dados estatísticos até algumas mais complexas que necessitam de anotações manuais para gerar sumários, todas atuando sobre o mesmo corpus detalhado no Capítulo 4.

Segundo Junior (2018), técnicas para o problema de sumarização geralmente recebem como entrada um conjunto de documentos e produzem um sumário com as sentenças mais representativas dessa coleção. É necessário tratar a redundância presente nas múltiplas opiniões sobre um mesmo tema. Nesses casos é possível utilizar requisitos considerados benéficos para a geração de um sumário de boa qualidade (LI *et al.*, 2009). Descrições mais detalhadas das entradas, saídas e dos processos envolvidos em cada uma das técnicas de sumarização abordadas neste trabalho são dadas no Capítulo 5.

1.2 Objetivos

O primeiro e principal objetivo desta pesquisa é aplicar várias técnicas voltadas à sumarização de opiniões em um corpus contendo opiniões retiradas de plataformas *online* de viagem e hotelaria, e analisar os sumários que elas são capazes de gerar com base nesse conteúdo. Após isso, gerar um relatório o mais completo possível contendo os resultados obtidos após os testes, e a capacidade de cada uma das técnicas em resumir as análises de hotéis, comparando os resultados com sumários gerados por anotadores humanos.

O segundo objetivo é desenvolver um novo corpus com opiniões retiradas de múltiplos hotéis da plataforma *TripAdvisor* para auxiliar na anotação, criação de sumários tanto de referência quanto automáticos, e a comparação das técnicas abordadas neste trabalho.

Já o terceiro e último objetivo é verificar se o conjunto de métricas ROUGE é capaz de avaliar a qualidade de sumários de opiniões com boa acurácia e baixa variação de resultados (evitando casos onde a pontuação dada a um sumário seja apenas por mera coincidência), visto que este conjunto foi pensado inicialmente para textos jornalísticos e científicos.

1.2.1 Objetivos específicos

1. Implementar algumas das técnicas de sumarização extrativa disponíveis na àrea de Processamento de Língua Natural (PLN).
2. Descobrir as características específicas e compartilhadas entre elas.
3. Analisar como elas lidam com os passos necessários para se gerar um sumário, tais como o tipo de entrada, o (pré) processamento feito, e o modelo usado para escolha das sentenças mais importantes.
4. Criar um novo corpus com opiniões retiradas da plataforma de viagens e hotelaria *TripAdvisor*.
5. Gerar sumários automáticos em cada uma das técnicas e uniformizá-los para realizar testes no conjunto ROUGE.
6. Criar sumários "ideais" com a ajuda de anotadores humanos para servirem como *gold standard* contra os quais os sumários automáticos são avaliados.
7. Analisar os resultados obtidos por cada método de sumarização em relação aos sumários gerados por anotadores humanos.
8. Apresentar tabelas contendo as técnicas e o desempenho das mesmas no conjunto ROUGE (especificamente ROUGE-1, ROUGE-2 e ROUGE-L), explicado com detalhes na subseção 2.3.1, para exibir o nível de qualidade alcançado pelos resumos automáticos em relação aos sumários "ideais".
9. Apresentar conclusões sobre a capacidade do conjunto ROUGE em avaliar a qualidade de sumários criados a partir de textos opinativos.

1.3 Organização do Texto

O restante deste trabalho é organizado da seguinte forma: No Capítulo 2 são descritos os conceitos e métodos utilizados como base para as técnicas; O Capítulo 3 contém os trabalhos correlatos; No Capítulo 4 é descrito o corpus criado neste trabalho e os passos de anotação e criação de sumários; No Capítulo 5 são apresentadas as técnicas de sumarização experimentadas; O Capítulo 6 descreve os resultados obtidos após a execução e validação das técnicas; E no

Capítulo 7 é apresentada uma conclusão geral sobre o que foi feito e as informações que puderam ser extraídas dos resultados.

CAPÍTULO

2

FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são abordados os diferentes conceitos e técnicas que serão vistas no decorrer deste trabalho e que são utilizadas na tarefa de sumarização automática de texto. A seção 2.1 trata dos diferentes tipos de sumarização, a seção 2.2 explica as etapas que constituem a sumarização de opiniões, e a seção 2.3 introduz os principais métodos disponíveis para avaliação dos sumários gerados pelos sumarizadores.

2.1 Tipos de sumarização

Esta seção é dedicada à explicação dos dois tipos de sumarização geralmente abordados pela sub-área de sumarização de textos, os quais são extrativos e abstrativos.

Sumarização Extrativa

A sumarização extrativa tem como único foco analisar os textos recebidos e gerar um sumário contendo as sentenças mais relevantes de acordo com a técnica e o resultado desejado, não realizando qualquer alteração gramatical no conteúdo. Essa característica torna a sumarização extrativa o principal escopo das pesquisas até então (NENKOVA; MCKEOWN, 2011), pois é a mais simples de se implementar visto que ela ignora completamente quaisquer interações linguísticas entre as sentenças selecionadas, e tampouco se preocupa em gerar ou alterar qualquer estrutura textual para melhorar a coesão do texto. Na Figura 2 é apresentado um exemplo do tipo de resultado normalmente gerado por um sumarizador extrativo com base no corpus utilizado por este trabalho. Nela é possível ver que o modelo extrativo apenas seleciona sentenças e as coloca no sumário com alguns separadores que indicam a ordem delas no sumário final, mas nada além disso.

Por ser o tipo mais estudado, existem várias técnicas que buscam gerar um sumário extrativo por meio de diferentes métodos de extração e classificação de sentenças.

Figura 2 – Exemplo de um Sumário Extrativo

[Reserva sem complicações no site do hotel.][1] [Café da manhã com boa variedade de itens.][2] [Localização excelente, vizinho ao Brasília Shopping.][3] [Setor de reservas muito atencioso.][4] [Cama razoável.][5] [Apesar de antigo é reformado e atende o que anuncia.][6]

Fonte: Autor

Técnicas baseadas em grafos por exemplo costumam representar as palavras do texto a ser sumarizado em vértices, e as relações entre as palavras em arestas. Essa característica de representar textos em grafos permitiu que Erkan e Radev (2004) desenvolvessem o *LexRank*, um sumarizador capaz de sumarizar múltiplos documentos.

Métodos estatísticos, como o de McCargar (2004), focam em analisar diretamente a frequência das palavras para verificar a probabilidade de serem centrais ao tema e boas candidatas a serem inseridas no sumário. Essa análise pode ser feita tanto pela frequência de cada palavra, quanto pelo método *Term Frequency-Inverse Document Frequency* (TF-IDF) explicado na subseção 2.2.4.

Abordagens com aprendizado de máquina buscam verificar a importância de uma dada sentença calculando a probabilidade dela aparecer no resumo gerado, e alguns até utilizam regressão, um exemplo sendo a *Support Vector Machines* (SVM) (DRUCKER *et al.*, 1996) utilizada pelo sumarizador FastSum de Schilder e Kondadadi (2008).

Há outras técnicas de sumarização, como as de (RIBALDO; PARDO; RINO, 2011) e (ZHONG *et al.*, 2020), que trabalham gerando múltiplos textos e buscando o relacionamento das palavras entre eles em um grafo para mensurar a importância delas, e comparando o documento original e o sumário gerado após a extração, respectivamente.

Mesmo assim, nenhuma destas técnicas visam alterar a estrutura do resumo gerado, deixando apenas as sentenças mais importantes extraídas do documento base. Por este motivo, sumarizadores extrativos costumam gerar documentos pouco coesos, com frases um tanto quanto desconexas e sentenças sem qualquer relação gramatical além de uma vírgula ou outro sinal de pontuação utilizado para separá-las. Essa é a principal fraqueza da abordagem extrativa, uma vez que resumos com estrutura gramatical mais próxima àquela gerada por um humano costumam ser mais fáceis de ler e compreender. Esse retrabalho na estrutura textual do sumário gerado é algo que os sumarizadores abstrativos buscam oferecer.

Sumarização Abstrativa

A sumarização abstrativa por outro lado consiste na geração de um sumário com sentenças extraídas de outros textos, mas com uma reescrita aplicada no conteúdo final de forma a se obter um documento com menos discrepâncias gramaticais entre sentenças. Então, se os sumarizadores

extrativos pecam em coesão textual e organização de sentenças, esse é justamente o foco das abordagens abstrativas, que atuam com base na saída gerada pelos métodos extrativos e aplicam um processamento nas sentenças presentes de forma a melhorar a qualidade do texto final, até mesmo gerando novas sentenças caso necessário.

De acordo com Nenkova e McKeown (2011), alguns dos métodos usados pelos sumarizadores abstrativos são:

- **Ordenação de sentenças:** Realiza uma verificação das sentenças atualmente presentes no texto e as ordenam de forma a tornar o documento o mais coeso possível. Essa ordenação é realizada de acordo com as sentenças julgadas mais importantes pelo algoritmo usado para extração de sentenças.

- **Revisão de sentenças:** Historicamente foi uma das primeiras formas de se gerar novas sentenças em uma sumarização, pois atua reutilizando trechos e expressões obtidos da entrada enviada ao sumarizador, inserindo-os conforme necessário para substituir algumas expressões julgadas pouco apropriadas no contexto onde estão. Métodos que focam na eliminação e combinação de sentenças também existem, podendo ser inclusive utilizadas em conjunto com o método de substituição de sentenças.

- **Fusão de sentenças:** Trabalha por meio da junção de duas sentenças que possuem informações similares, mas também algumas expressões diferentes, buscando gerar uma nova sentença que possua a informação contida nas duas anteriores, ou todas as informações destas sem qualquer redundância.

- **Compressão de sentenças:** Funciona com base no comportamento humano de citar ou mencionar trechos dos documentos originais em seus sumários, removendo alguns pedaços para torná-los mais concisos.

Mesmo sendo estudada há vários anos, a quantidade de trabalhos voltados à sumarização abstrativa é bem pequena se comparada com a abordagem extrativa. Isso se dá devido à complexidade de se alterar o sumário final e gerar ou alterar sentenças já escritas, o que envolve um amplo conhecimento gramatical, léxico, e do domínio ao qual o texto resumido pertence (CONDORI, 2014).

Ganesan (2013) define duas categorias de abordagens para sumarização abstrativa, uma sendo baseada em conhecimentos à priori enquanto a outra utiliza conceitos e sistemas de Geração de Linguagem Natural.

Como exemplo da primeira categoria, há o trabalho de Jung e Jo (2003), que visa sumarizar *E-mails* com base em modelos de documentos pré-criados, bastando inserir as sentenças nos espaços indicados para obter um sumário relativamente conciso. Cada documento pré-criado, ou

"modelo", possui etiquetas semânticas que indicam o tipo de conteúdo que deve ser inserido em cada espaço em branco, facilitando a tarefa de extração de informações.

A segunda categoria, por sua vez, é composta de técnicas que utilizam sistemas para geração de novas sentenças e alteração das já presentes, com um exemplo destas sendo visto em trabalhos como o de Radev (1997). A técnica em questão utiliza um sistema de sumarização com base em conhecimento do tópico (*knowledge-based*) e isso faz com que ela tenha de ser restringida a um domínio específico, que no caso foi o de notícias a respeito de ataques terroristas.

2.2 Sumarização de opiniões

Esta seção exemplifica o processo envolvido na sumarização de opiniões, pré-processamento das informações e as etapas até se obter um resumo com base nos textos enviados aos sumarizadores.

2.2.1 Definição

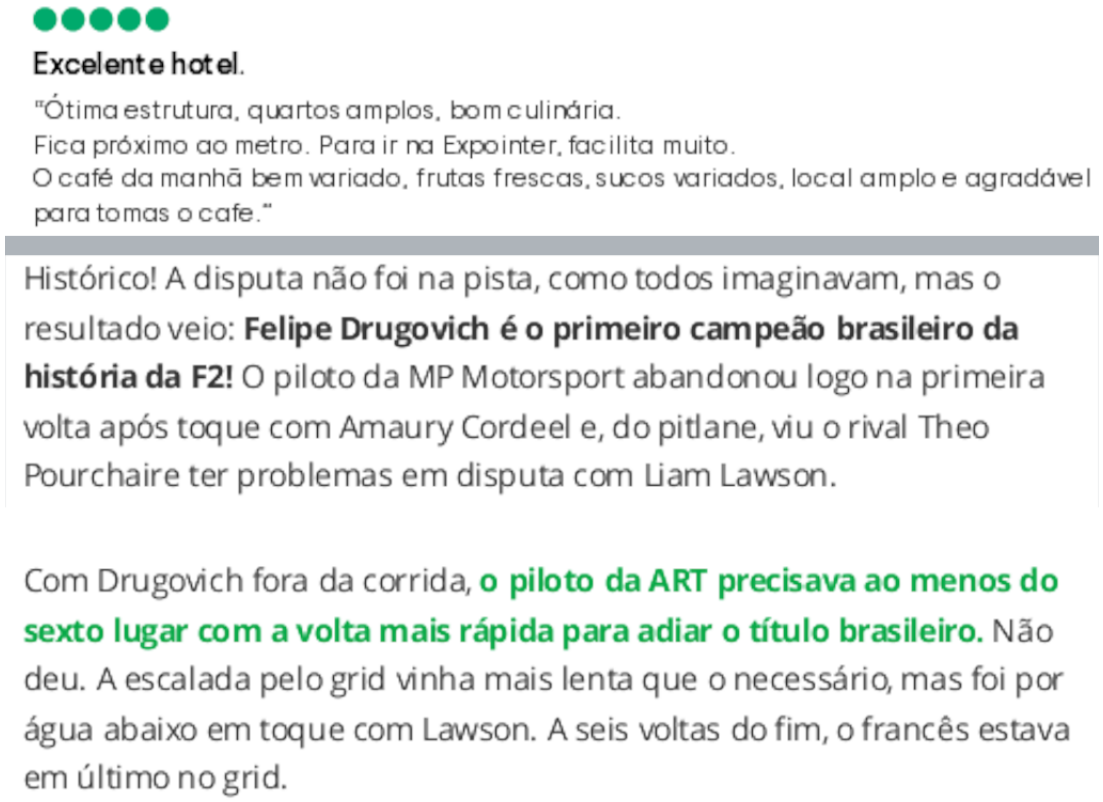
Conforme descrito por Mani (2001), a sumarização automática consiste na extração do conteúdo mais pertinente dentro do tema abordado, e a apresentação desse conteúdo de forma condensada e voltada às necessidades do usuário. Segundo Condori (2014), a Sumarização de Opiniões é uma mistura de Mineração de Opiniões e Sumarização Automática de Textos, buscando gerar um resumo de opiniões ao invés de textos, onde a Mineração de opiniões encontra a orientação semântica das opiniões e a Sumarização Automática de Textos encontra as opiniões mais relevantes, gerando um sumário com elas em seguida.

Em relação às diferenças quando comparada à sumarização de texto, as principais são que os resumos de opiniões focam nos aspectos avaliados e na comparação de sentimentos de diferentes opiniões, possuem um formato mais estruturado, e lidam com a escrita informal comumente encontrada nesses tipos de texto. A Figura 3 apresenta a diferença gramatical e estrutural entre uma opinião do *TripAdvisor*, e um texto jornalístico do portal de notícias G1.

Observe que na Figura 3, a opinião retirada do *TripAdvisor* possui alguns erros de concordância verbal e de grafia, além de conter informações subjetivas, isto é, de acordo com as experiências do autor da opinião em sua estadia no hotel. Já a notícia do G1 não possui erros gramaticais e apresenta informações objetivas, informando algo que factualmente aconteceu e pode ser comprovado por terceiros.

Outro fator a ser considerado na sumarização e opiniões é o de que a redundância de informações encontrada pelo sumarizador é um forte indicador da importância delas, ao contrário do sumarizador tradicional que descarta informações redundantes (PANG; LEE *et al.*, 2008). Isso ocorre pois a repetição de informações similares em opiniões diferentes denota uma maior confiabilidade do conteúdo destas informações.

Figura 3 – Diferença entre uma opinião do *TripAdvisor* e uma notícia do G1



Fonte: Autor

2.2.2 Classes de opiniões

Segundo Liu (2012) existem dois tipos de opiniões, sendo classificadas de acordo com a forma que elas descrevem as características do produto ou serviço avaliado. Por conta disso, as opiniões são classificadas em opiniões regulares ou comparativas com bases em alguns critérios de classificação. Tenha em mente que os exemplos citados nos próximos parágrafos foram retirados do corpus descrito no Capítulo 4.

As opiniões regulares são aquelas que descrevem as características de um produto ou serviço apenas de acordo com as expectativas do avaliador, não levando em conta quaisquer outros concorrentes. Portanto, uma opinião regular geralmente apenas descreve algo de forma positiva ou negativa, como “Hotel grande, com boa localização [...]” ou “muito caro para o que ofereceu. [...]” por exemplo, onde há um sentimento positivo e negativo na primeira e segunda sentença, respectivamente, mas nenhuma comparação.

Opiniões comparativas, por outro lado, sempre possuem alguma comparação em suas entrelinhas, fazendo com que a avaliação, seja de forma positiva ou negativa, esteja atrelada a algum outro produto ou serviço no mesmo setor daquele avaliado. Por conta disso, é comum que estas análises tenham uma estrutura similar a “[...] O cafe da manha deixou a desejar, um carinho na mesa com toalhas, mesa montada e flor natural e muito simples para um hotel deste padrão.”

e "[...] busquem outros hotéis com maior qualidade no atendimento [...]", onde as comparações são visíveis, mesmo que implicitamente.

2.2.3 Formas de se sumarizar opiniões

Ainda de acordo com Liu (2012), existem três formas distintas para se realizar a sumarização de opiniões, estas são: Sumarização Tradicional, Sumarização Contrastiva e Sumarização Baseada em Aspectos.

A Sumarização Tradicional é, em linhas gerais, uma das mais simples de ser realizada, pois não leva em conta características como sentimentos e aspectos, permitindo a seleção de sentenças mais gerais e que não focam em algo específico ou tenham alguma carga sentimental positiva ou negativa por trás. Ela também não verifica a razão entre opiniões positivas ou negativas, o que a impede de gerar um relatório contendo o sentimento geral das opiniões (LIU, 2012).

A Sumarização Contrastiva, por sua vez, busca opiniões contrastantes, isto é, que possuem pontos de vista contrários para informar melhor o leitor dos prós e contras em relação a algo, já que agora ele terá a possibilidade de comparar as opiniões, pesando os pontos negativos e positivos pessoalmente (LIU, 2012).

Por fim, a Sumarização baseada em Aspectos gera um resumo das opiniões mais relevantes para cada aspecto, com base em um conjunto de análises sobre um mesmo produto ou serviço. Ela atua em duas frentes, analisando a as opiniões e o sentimento por trás delas em cada aspecto, e a razão entre sentimentos positivos e negativos com base nas opiniões verificadas (CONDORI, 2014).

2.2.4 Pré-processamento do conteúdo textual

Os sumarizadores automáticos não recebem as opiniões como entrada diretamente, sendo necessário realizar um pré-processamento no conteúdo textual, e em alguns casos, a vetorização das palavras também é necessária.

2.2.4.1 Etapas de pré-processamento textual

O pré-processamento textual é dividido em várias etapas para delinear melhor o que é feito em cada uma delas até se obter o conteúdo preparado para ser enviado como entrada aos algoritmos de sumarização (JUNIOR, 2018). Por ser uma etapa que contém uma quantidade flexível de passos a depender da técnica de sumarização, diferentes autores utilizam conjuntos de passos distintos em seus trabalhos. Algumas destas etapas de pré-processamento são descritas a seguir, não sendo necessário seguir a ordem de aparição, nem todas as etapas apresentadas:

- **Segmentação de sentenças:** Responsável por receber o texto puro e devolver as sentenças que o compõem. Alguns trabalhos como o de Junior (2018) também realizam a resolução de ambiguidades e verificação de delimitadores de sentença através de algoritmos como Modelos de Entropia Máxima (RATNAPARKHI, 1998) nesta etapa.

- **Segmentação de *tokens*:** Recebem a saída da etapa anterior e geram termos não vazios, que por sua vez são caracteres comuns de palavras ou números, ou divididos em mais de um *token* quando iniciam ou terminam por uma pontuação. Essa divisão em vários *tokens* ocorre porque símbolos de pontuação tendem a ser usados como delimitadores de sentença em sumarização de opiniões.

- **Análise morfossintática:** Se encarrega de verificar as classes gramaticais das palavras presentes no texto, bem como gênero, número, tempo verbal e outros, através de algoritmos que aprendem a rotular sequências de *tokens*. Para facilitar essa tarefa, é possível utilizar algoritmos de *Part-Of-Speech tagging* como o de Porter (1980).

- **Remoção de *stopwords*:** Recebe a sequência de *tokens* e remove os termos que são muito comuns e irrelevantes para a classificação naquele idioma ou domínio, como “que”, “para”, etc. devido ao baixo valor semântico.

- **Redução ao radical (*Stemming*):** Transforma os termos em suas versões radicais, isto é, sem conjugação, gênero e outros, reduzindo ao máximo a palavra à sua raiz, resultando até mesmo em palavras inexistentes. Um exemplo de redução ao radical é a palavra “bebidas” que é transformada apenas em “beb”. Essa redução agressiva pode acarretar em problemas como reduzir demais uma palavra, e fazê-la perder seu sentido intrínseco, tornando o próximo ítem do conjunto de etapas mais preferível em casos onde a informação gramatical deve ser mantida.

- **Redução ao lema (*Lemmatization*):** É similar ao *stemming*, pois busca reduzir a palavra à sua raiz, retirando as inflexões da língua para obter o lema da palavra. Porém, a redução ao lema leva em conta a classe gramatical da palavra, o que lhe permite gerar um resultado gramaticalmente correto e que não perde o sentido após a redução. Como um exemplo de redução ao lema, a palavra “bebidas” é reduzida para “bebida”, que é uma palavra existente na língua portuguesa.

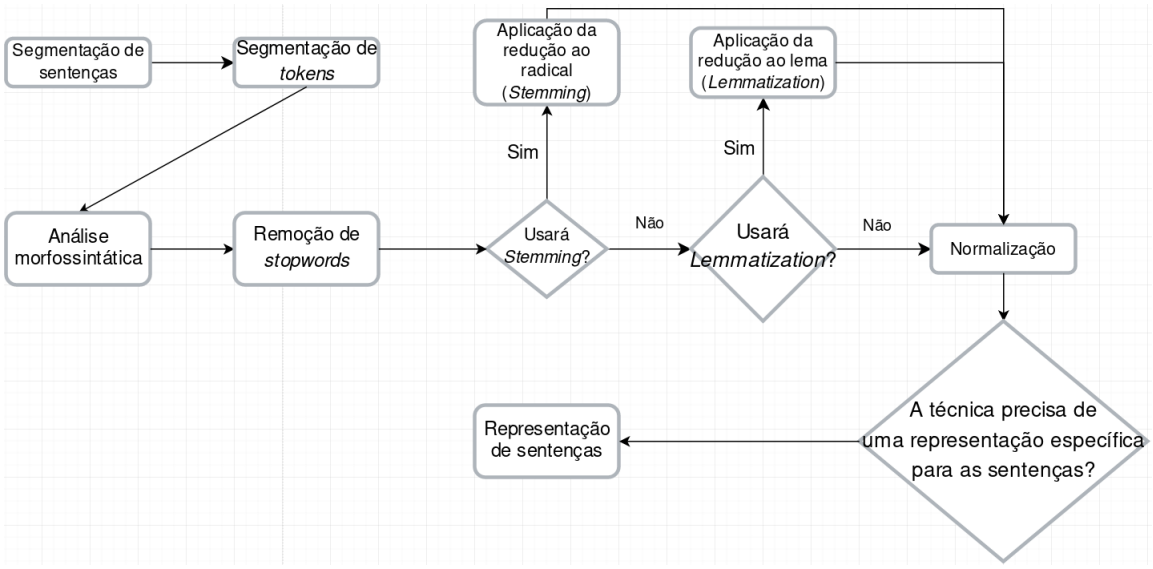
- **Normalização:** Remove caracteres especiais, números, além de converter os caracteres de um *token* para maiúsculo ou minúsculo a depender da representação desejada. Esta etapa ajuda a remover tokens iguais, mas com capitalização diferente ou com presença de símbolos de pontuação.

- **Representação de sentenças:** Embora raramente utilizada como pré-processamento, e sendo vista dessa forma no trabalho de Junior (2018), esta etapa formula a representação final das sentenças antes do envio dos dados a algoritmos com *machine learning*. Esta

representação final pode ser gerada através do uso de TF-IDF (a ser vista em seguida) para calcular a importância das palavras, ou composta por palavras cujas classes gramaticais são anotadas manualmente como substantivos, adjetivos ou advérbios. Isso se dá porque estes tipos de palavras com maior importância ou anotadas manualmente são os que condizem melhor com o processo de análise de sentimento e sentido das frases que compõem uma opinião.

Diferentes técnicas de sumarização utilizam diferentes etapas de pré-processamento, e a ordem delas também pode variar. A Figura 4 representa um diagrama contendo as etapas de pré-processamento mais comuns, vistas em trabalhos como (CONDORI, 2014) e (JUNIOR, 2018), além de um exemplo da sequências de aplicação utilizada nas técnicas contempladas por este trabalho.

Figura 4 – Diagrama exemplificando as etapas de pré-processamento, e uma das possíveis ordens



Fonte: Autor

2.2.5 Processamento e geração do sumário

Com as etapas de pré-processamento devidamente realizadas, dá se início à modelagem e descoberta de tópicos, cálculo de importância e similaridade das sentenças, e outros estágios que compõem a fase de processamento. Essa fase também é composta por várias etapas, vistas a seguir.

2.2.5.1 Cálculo do TF-IDF

Em todas as técnicas contempladas neste trabalho, é necessário obter as palavras mais importantes dos documentos para que seja possível ranquear as palavras e auxiliar as técnicas na extração das informações. Em situações como essas, é possível utilizar métodos como o

TF-IDF, que pontuam a importância das palavras com base na frequência das aparições delas entre diferentes documentos. Intuitivamente, uma pontuação alta é dada a uma palavra que aparece frequentemente em um documento, e uma pontuação baixa é dada a aquela que aparece em muitos documentos. Isso faz com que palavras tais como “por que” e “portanto” recebam uma baixa pontuação por exemplo.

O TF-IDF é uma união de duas estatísticas distintas, a *Term Frequency* de Luhn (1957) e a *Inverse Document Frequency* de Jones (1972), com a primeira sendo voltada à frequência dos termos em um mesmo documento, e a segunda à frequência deles em múltiplos documentos, mas de forma invertida (ou seja, quanto mais comum em vários documentos, menos importante). Por conta disso, os cálculos são realizados isoladamente, começando pelo TF que é dado pela Equação 2.1, com i sendo uma palavra (termo), e j um documento:

$$TF_{i,j} = \frac{n_{i,j}}{\sum_{i \in j} n_{i,j}} \tag{2.1}$$

A Equação 2.1 representa o cálculo da razão entre a quantidade de vezes n que uma certa palavra aparece em um documento em relação ao total de palavras presentes nele (calculado pela somatória da frequência de todas as k palavras únicas no documento), então quanto maior o valor $TF_{i,j}$ resultante, mais importante a palavra. O cálculo do IDF, por sua vez é realizado pela seguinte equação:

$$IDF_{i,N} = \log\left(\frac{|N|}{|\{j \in N : t \in j\}|}\right) \tag{2.2}$$

Na Equação 2.2, N é a quantidade total de documentos, então o valor IDF é resultado logarítmico da razão entre o total de documentos no numerador e a quantidade de documentos contendo o termo i no denominador. Com isso, a pontuação de uma palavra pelo TF-IDF é calculado através do produto das funções TF e IDF:

$$w_{i,j} = (TF_{i,j} \times IDF_{i,N}) \tag{2.3}$$

Conforme a Equação 2.3, a frequência da palavra em um certo documento é normalizada pela frequência dela em outros documentos no mesmo conjunto analisado. Essa característica evita que palavras e expressões comuns do dia a dia sejam incluídas nos sumários, ressaltando apenas aquelas mais importantes ao tópico central de um documento específico.

2.2.5.2 Modelagem e descoberta de tópicos

Esta etapa busca extrair os tópicos mais importantes abordados pelas opiniões presentes no corpus, visando calcular a importância das sentenças em relação aos tópicos encontrados anteriormente. Um bom modelo a ser utilizado para isso é o *Latent Dirichlet Allocation* (LDA)

descrito por Blei, Ng e Jordan (2003) pois ele é capaz de extrair os tópicos do corpus por meio de uma abordagem não-supervisionada e também permite uma série de customizações na execução e treinamento.

O LDA é um modelo gerativo probabilístico de um corpus, onde documentos são representados como misturas de tópicos latentes, com cada tópico sendo ligado a um conjunto de palavras. Suponha um conjunto de N de documentos contendo um vocabulário de P tipos de palavras (*types*) particionados em M tópicos. Nessa configuração cada documento é caracterizado por um *token* de um número X de palavras e um vetor de dimensão M para representar a distribuição de tópicos nesse documento. Essa distribuição é calculada com base na fórmula de probabilidade da distribuição de Poisson, enquanto que as P palavras pertencentes a um determinado tópico são calculadas por meio de uma distribuição multinomial.

As customizações do modelo podem ser feitas em uma série de parâmetros, sendo permitido ajustar variáveis como a quantidade de documentos processados durante o treinamento e tópicos buscados, e a quantidade de "épocas" e iterações. Por conta disso, é possível adaptá-lo a diferentes tamanhos de corpus, quantidades de tópicos e formas de aprendizado, o que lhe torna bastante flexível. Outro fator a se considerar é o de que não é necessário desenvolver a lógica do LDA a partir do zero, pois ele possui implementações em bibliotecas como Gensim [1] e Scikit-Learn [2], facilitando a aplicação do algoritmo.

A união dessa flexibilidade de configurações com a possibilidade de se encontrar tópicos e suas principais palavras em uma série de domínios distintos é o que torna o LDA um excelente algoritmo para modelagem e descoberta de tópicos.

Porém, antes de executar o LDA, é ideal que seja aplicada uma conversão das opiniões de texto para vetores de frequência por meio de técnicas como TF-IDF (RAMOS *et al.*, 2003). O modelo também necessita de um bom ajuste dos parâmetros para que possa gerar um resultado relevante, com os mais comuns sendo o número de tópicos a serem extraídos o número de iterações sobre os documentos, a decaída da taxa de aprendizagem e a quantidade de tópicos por palavra. Com a conversão feita e os parâmetros passados para a entrada do LDA, é feito o treinamento do modelo para encontrar a distribuição de tópicos abordados pelas opiniões, gerando como saída a quantidade de tópicos a ser levada em conta e a quantidade de palavras relevantes a cada um deles.

Após a aplicação do modelo, pode se aplicar uma normalização no tamanho das sentenças através da contagem de palavras, composta da intersecção de dois conjuntos de palavras diferentes, um deles representando uma sentença S e o outro um tópico T. Nessa contagem, quanto maior o coeficiente de intersecção entre eles, mais abrangente é o conteúdo do tópico T contido em S. Esse método de intersecção é mais adequado pois evita que sentenças maiores recebam pontuações mais altas apenas pela frequência de certas palavras, mesmo que estas não

[1] https://radimrehurek.com/gensim/models/ldamodel.html
[2] http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.LatentDirichletAllocation.html

cubram o tópico com tanta eficácia quanto outra menores (JUNIOR, 2018).

2.2.5.3 Processo de sumarização

A sumarização em si ocorre de diferentes maneiras a depender do algoritmo aplicado, ser supervisionada ou não supervisionada de acordo com a técnica utilizada. Abordagens supervisionadas passam por um estágio de treinamento para aprender a classificar as informações e então aplicar esse conhecimento no conjunto de informações de interesse, enquanto que as abordagens não supervisionadas atuam diretamente nas informações de interesse, não necessitando de um treinamento prévio. Por conta disso, as abordagens não supervisionadas podem classificar as informações de forma menos precisa do que as supervisionadas.

Abordagens supervisionadas exigem uma alimentação constante de dados para que consigam formular um modelo preciso capaz de sumarizar documentos e gerar bons resultados. Um exemplo deste tipo de abordagem é a técnica de Im *et al.* (2021), que utiliza modelos pré-treinados que atuam como codificadores (*encoders*) e decodificadores (*decoders*) para geração de textos. Um desses modelos é o *Bidirectional Encoder Representations from Transformers* (BERT) (LEWIS *et al.*, 2019), um modelo *Transformer* (VASWANI *et al.*, 2017) capaz de realizar tanto codificação quanto decodificação, além de ter um bom desempenho em sumarização quando ajustado para geração de texto. Modelos como esse atuam como uma "caixa preta", onde o algoritmo, após treinado, recebe uma entrada e gera uma saída, não sendo possível verificar com precisão como o resultado foi gerado.

Já abordagens não-supervisionadas são mais automáticas e não necessitam de iterações sucessivas até obterem um modelo preciso, podendo atuar quase que imediatamente na sumarização de textos. A técnica não-supervisionada *k-medoids* de Kaufmann e Rousseeuw (1987), por exemplo, cria um grafo com agrupamentos de sentenças e busca encontrar os k agrupamentos mais importantes. Um *medoid* é um objeto de um agrupamento cuja soma das dissimilaridades para todos os outros objetos no agrupamento é a menor encontrada. As sentenças extraídas dos agrupamentos obtidos serão adicionados ao sumário e, portanto devem atender aos requisitos impostos durante a fase inicial do processamento, não exigindo muita interação além disso até gerar o sumário.

Uma explicação com maiores detalhes sobre as várias técnicas de sumarização de opiniões utilizadas neste estudo e seus respectivos métodos é dada no Capítulo 5.

2.3 Métodos de avaliação dos resumos gerados

Aqui são descritas algumas das metodologias utilizadas na avaliação dos sumários gerados automaticamente, tais como as medidas ROUGE e a abordagem para avaliações humanas, mesmo que esta última não tenha sido aplicada devido a restrições de tempo.

2.3.1 Métricas ROUGE para avaliação automática dos sumários

O conjunto de métricas de avaliação ROUGE, proposto por Lin (2004), engloba uma série de métodos avaliativos para julgar a qualidade de um resumo. A avaliação é feita através da comparação dos sumários gerados automaticamente com sumários "ideais" feitos por um humano, que são usados como a meta a ser alcançada pelos sumarizadores, já que um resumo humano é (espera-se) perfeitamente compreensível e bastante informativo para outros humanos acerca de um determinado tópico.

O conjunto ROUGE é composto de quatro métricas diferentes e algumas extensões, sendo elas ROUGE-N, ROUGE-L, ROUGE-W, ROUGE-S e sua extensão ROUGE-SU, cada uma focada em mensurar e avaliar os documentos conforme um aspecto diferente da composição dos mesmos, e seus respectivos funcionamentos são exemplificados de uma forma geral nas sub-subseções a seguir:

2.3.1.1 ROUGE-N

Esta métrica lida com as estatísticas de co-ocorrência de n-gramas entre o sumário a ser analisado e uma série de sumários de referência, sendo calculado como a divisão do número de n-gramas do sumário analisado presentes nos sumários de referência pela quantidade total de n-gramas nos sumários de referência. Por ser uma abordagem voltada a n-gramas, é possível usá-lo para verificar uni-gramas comuns entre os sumários automáticos e de referência pelo ROUGE-1, bigramas pelo ROUGE-2, e assim em diante. O cálculo da ROUGE-N é exibido na Equação 2.4, onde o n é o comprimento do n-grama, $gram_n$ e $Count_{match}(gram_n)$ representam a quantidade máxima de co-ocorrências de n-gramas entre o resumo candidato e um conjunto de resumos usados como referência.

$$\text{Rouge-N} = \frac{\sum_{S \in \{SumariosDeReferencia\}} \sum_{gram_n \in S} Count_{match}(gram_n)}{\sum_{S \in \{SumariosDeReferencia\}} \sum_{gram_n \in S} Count(gram_n)} \tag{2.4}$$

Essa fórmula demonstra que a ROUGE-N é uma métrica *recall-based*, que calcula a quantidade de sobreposições de n-gramas encontradas no sumário automático e no de referência, e então divide o resultado pelo total de n-gramas do sumário "ideal". Logo, a quantidade de n-gramas presentes no denominador da equação aumenta conforme mais sumários de referência são adicionados, pois é possível que existam vários sumários ideais, e além disso, um resumo candidato que possua palavras e n-gramas presentes em mais referências é favorecido pelo ROUGE-N, já que é desejável que esse resumo seja o mais similar possível aos sumários de referência disponíveis.

2.3.1.2 ROUGE-L

Atua por meio da verificação de sequências e sub-sequências nos documentos, onde uma sequência A é sub-sequência de uma sequência B. Dadas as sequências A e B, se para todo

elemento k de A há um elemento k correspondente em um determinado índice i de B, quanto maior a cadeia de elementos em comum entre os resumos for, mais similares são A e B. Isso faz com que ela seja ideal para avaliar a similaridade dos sumários em casos onde a ROUGE-N não funciona como deveria, ou gera resultados aquém do esperado devido a palavras ou grupos de palavras muito recorrentes.

O ROUGE-L trabalha com o modelo *Longest Common Subsequence* (LCS) a nível de sentença, tratando a sentença como uma sequência de palavras para avaliar a similaridade de dois sumários. Na Equação 2.7, Lin (2004) propõe uma forma de estimar a similaridade entre dois sumários X e Y de tamanhos diferentes "m" e "n", com X sendo considerado uma sentença do sumário de referência e Y, uma sentença do sumário candidato. Considere que $LCS(X,Y)$ representa a maior sub-sequência comum presente tanto em X quanto em Y na Equação 2.5 e Equação 2.6.

$$R_{lcs} = \frac{LCS(X,Y)}{m} \tag{2.5}$$

O cálculo do R_{lcs} se refere à maior sub-sequência comum encontrada com base no tamanho da sentença do sumário de referência, com $LCS(X,Y)$ sendo dividido pelo tamanho m do sumário de referência.

$$P_{lcs} = \frac{LCS(X,Y)}{n} \tag{2.6}$$

Já o cálculo de P_{lcs} leva em conta a maior sub-sequência comum encontrada com base na sentença do sumário candidato, com $LCS(X,Y)$ sendo dividido pelo tamanho n do sumário candidato. Após obter estes resultados, pode-se inseri-los no cálculo da ROUGE-L, dado pela Equação 2.7.

$$F_{lcs} = \frac{(1+b^2)R_{lcs}P_{lcs}}{R_{lcs}+b^2P_{lcs}} \tag{2.7}$$

Aqui é calculada a medida-F (*F-measure*) da ROUGE-L, com b sendo uma variável que representa o cálculo de $\frac{P_{lcs}}{R_{lcs}}$ quando $\frac{F_{lcs}}{R_{lcs}} = \frac{F_{lcs}}{P_{lcs}}$. Mas é comum que em conferências como a *Document Understanding Conference* (DUC) a variável b seja configurada com um valor bem alto para que o algoritmo considere apenas o R_{lcs}. Esse cálculo de F é feito de forma a resultar em 1 caso X seja idêntico a Y, que é quando a maior sub-sequência comum entre ambos é todo o texto, e 0 caso não haja quaisquer sub-sequências em comum. Porém, essa fórmula LCS não leva em conta sequências mais curtas e outras aplicações LCS no resultado final, então caso hajam duas ou mais sub-sequências encontradas em uma mesma sentença, apenas uma delas é contabilizada. Também é possível aplicar o LCS a nível de sumário, mas essa modalidade de avaliação não foi aplicada neste trabalho pois seria necessário gerar uma grande quantidade de sumários de referência sobre um mesmo tópico para justificar o uso deste modelo do ROUGE-L.

2.3.1.3 ROUGE-W

É uma melhoria do ROUGE-L, adicionando pesos conforme a proximidade dos elementos em duas ou mais cadeias para remediar a limitação do ROUGE-L em detectar as relações entre as sequências analisadas. Para o ROUGE-L, uma sequência $A->B->C$, uma subsequência $A->*->B->*->C$ e outra $A->B->*->C$ possuem a mesma pontuação, sendo que a segunda deveria obter uma pontuação mais alta por conter os elementos em um posicionamento mais próximo da cadeia de referência, por exemplo. A ROUGE-W busca resolver isso adicionando pesos para certas características das correspondências, como correspondências consecutivas, em maior número, etc., de forma a não cair nas mesmas limitações da LCS sem pesos, com o cálculo sendo expresso pela Equação 2.8 e Equação 2.9. A fórmula para cálculo da ROUGE-W é dada pela Equação 2.10, exibindo várias similaridades com a ROUGE-L, mas ainda tendo alguns diferenciais importantes.

$$R_{wlcs} = f^{-1}\frac{WLCS(X,Y)}{f(m)} \tag{2.8}$$

Nessa equação é calculada a *Weighted Longest Common Subsequence* (WLCS), sendo basicamente o $LCS(X,Y)$ da ROUGE-L com pesos aplicados. Ela leva em conta alguns fatores extras para calcular a similaridade, como a propriedade $f(x+y) > f(x)+f(y)$ para a função f utilizada como o peso na fórmula. Isso garante que correspondências consecutivas serão mais valiosas que correspondências comuns. O resultado é então dividido pela função com o tamanho m do sumário de referência. No fim a função é multiplicada pela inversa dela mesma para normalizar o resultado final. O cálculo levando em conta o tamanho do sumário candidato na Equação 2.9 segue uma lógica similar à do cálculo do P_{lcs} visto na subseção da ROUGE-L, mas também aplicando os pesos.

$$P_{wlcs} = f^{-1}\frac{WLCS(X,Y)}{f(n)} \tag{2.9}$$

Por conta do diferencial ser apenas a aplicação de pesos, o cálculo de P_{wlcs} é similar ao de R_wlcs visto anteriormente, com a diferença sendo a função de tamanho que agora é baseada no tamanho n do sumário candidato. Com os resultados em mãos, basta aplicá-los no cálculo final da ROUGE-W, dado pela Equação 2.10.

$$F_{wlcs} = \frac{(1+b^2)R_{wlcs}P_{wlcs}}{R_{wlcs}+b^2P_{wlcs}} \tag{2.10}$$

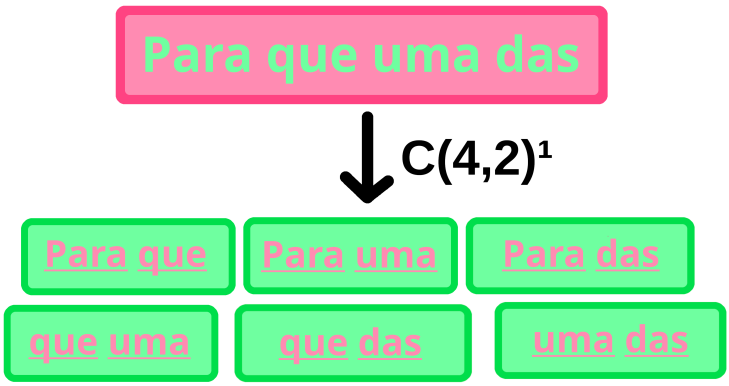
A pontuação final do ROUGE-W geralmente é normalizada, então atenção especial é dada para a utilização de funções que possuam uma função inversa similar no cálculo de R_{wlcs} e P_{wlcs}, com b e outras variáveis funcionando de forma análoga ao que foi visto na ROUGE-L. Um exemplo de uma função inversa é $f(x) = x^2$ que tem uma inversa $f^{-1}(x) = x^{1/2}$. Nas equações

anteriores ao F_{wlcs} é possível ver a normalização sendo aplicada nos dois sumários através da utilização de uma função inversa.

2.3.1.4 ROUGE-S

O foco desta métrica é avaliar as estatísticas de co-ocorrência de *Skip-Bigrams* nos documentos analisados. Um *Skip-Bigram* é um par de palavras que possuem uma ordem pre-definida, mas que podem conter quaisquer "espaços" entre seus elementos constituintes. Para ajudar a entender como os *skip-bigrams* funcionam, a Figura 5 ilustra um pequeno exemplo dos *skip-bigrams* que podem ser gerados a partir de um segmento de texto extraído do corpus desenvolvido neste trabalho.

Figura 5 – Exemplo de um pequeno trecho de uma sentença convertido em Skip-Bigrams



Fonte: Autor

Com base nisso, o ROUGE-S se encarrega de verificar a quantidade de sobreposições de *Skip-Bigrams* entre o sumário automático e os ideais. A Equação 2.11 ilustra os passos utilizados para se calcular a similaridade entre dois textos X e Y por meio da ROUGE-S. Na equação Equação 2.12, Equação 2.13 e Equação 2.11, $SKIP2(X,Y)$ se refere à quantidade de *Skip-Bigrams* correspondentes entre X e Y, e C é uma função combinatória entre o comprimento dos textos e o número de palavras que compõem um *Skip-Bigram*, neste caso, duas. A Equação 2.12 demonstra o cálculo de correspondências de *Skip-Bigrams* entre X e Y com base no tamanho do sumário de referência, com uma lógica análoga ao cálculo do R_{lcs}.

$$F_{skip2} = \frac{(1+b^2)R_{skip2}P_{skip2}}{R_{skip2}+b^2P_{skip2}} \tag{2.11}$$

As variáveis R_{skip2} e P_{skip2} são calculadas através da Equação 2.12 e pela Equação 2.13, respectivamente.

$$R_{skip2} = \frac{SKIP2(X,Y)}{C(m,2)} \tag{2.12}$$

A principal diferença aqui é que além da contagem de *Skip-Bigrams*, o tamanho é a combinação C explicada acima, que recebe o tamanho m do documento de referência como uma de suas variáveis. Para calcular as sobreposições de *Skip-Bigrams* com base no tamanho do sumário candidato, é preciso aplicar a Equação 2.13.

$$P_{skip2} = \frac{SKIP2(X,Y)}{C(n,2)} \tag{2.13}$$

Nessa equação, o P_{skip2} segue um método similar ao utilizado pelo R_{skip2}, com a principal diferença sendo o fato dele considerar o tamanho n do sumário candidato na combinação. Feitos esses cálculos, é possível jogar os resultados obtidos diretamente no cálculo final da ROUGE-S, expresso pela Equação 2.11.

Embora a utilização de uma métrica com base em *Skip-Bigrams* em vez de LCS traga algumas vantagens como maior flexibilidade na categorização e contagem de correspondências entre dois textos, ela também possui suas desvantagens. Um exemplo disso é o fato dela contar até as correspondências mais banais entre os dois textos caso nenhum tratamento em relação à distância entre as palavras ou características destas seja aplicado. Isso faz com que ela acabe permitindo a entrada de correspondências tais como “por isso” ou “já que” em certos casos.

2.3.1.5 ROUGE-SU

É uma extensão do ROUGE-S que utiliza uni-gramas como medida de contagem, o que lhe permite diferenciar casos onde o ROUGE-S classificaria uma sentença com um único elemento similar em ambos os tipos de sumário e outra sentença completamente diferente nos dois como idênticas, sendo que uma delas claramente possui um coeficiente de similaridade maior.

2.3.2 Abordagem para avaliação humana dos sumários

Embora o conjunto ROUGE seja bem robusto no que tange à avaliação dos sumários em termos objetivos, este é incapaz de verificar um sumário por si só, ele sempre necessita de um ou mais sumários *gold standard* para poder comparar a qualidade do resumo candidato (LIN, 2004). Em casos onde um sumário precisa ser avaliado em sua qualidade linguística, e sem ser comparado com quaisquer outros documentos, uma avaliação realizada por pessoas da área de PLN se faz necessária.

Nesse quesito avaliativo, conferências como a DUC e *Text Analysis Conference* (TAC) utilizam cinco critérios linguísticos para avaliar os sumários extrativos e abstrativos, que segundo Dang (2005), são os seguintes:

- **Gramaticalidade:** Não devem haver erros gramaticais, problemas de capitalização, formatações e textos utilizados apenas pelos algoritmos, ou quaisquer outros elementos que

dificultem a leitura do texto.

- **Não redundância:** O sumário não deve conter repetições desnecessárias, sejam sentenças, palavras ou até nomes em ocasiões onde "ele/ela" descrevem bem o suficiente.
- **Clareza referencial:** A identificação do sujeito ou pronome ao qual se refere deve ocorrer de forma simples e direta, com menções claras às entidades e sua importância sendo essenciais.
- **Foco:** As sentenças no sumário devem conter apenas informações pertinentes ao restante do texto, ou seja, o foco do texto deve ser bem claro.
- **Estrutura e Coerência:** O documento precisa estar bem estruturado e organizado, não sendo apenas um amontoado de informações, devendo conter uma estrutura coerente que liga cada sentença para formar uma informação compreensiva sobre um determinado tópico.

Nas conferências, geralmente cada um dos itens linguísticos acima é avaliado em uma escala de pontos de 1 a 5, onde cada número representa, em ordem crescente, "Muito ruim", "Ruim", "Minimamente aceitável", "Bom", e "Muito bom". Também é possível criar novas categorias como "utilidade", "novidade" e outras para melhor avaliar o resumo automático, embora não seja realmente necessário para se obter uma ideia de como ele se sai em geral com base nos cinco itens anteriores.

Também é preciso salientar que alguns dos itens acima não são aplicados a sumários extrativos, pois não faz sentido avaliar coerência e clareza referencial em uma modalidade de sumarização que somente extrai sentenças. Além disso, gramaticalidade não pode ser usada em sumários de opiniões visto que a própria fonte pode possuir problemas nessa métrica.

CAPÍTULO

3

TRABALHOS CORRELATOS

Neste capítulo são descritos os trabalhos relacionados ao conteúdo desta pesquisa, contendo informações como objetivos buscados, metodologias usadas, experimentos realizados, resultados obtidos, e a conclusão chegada pelos autores em seus respectivos trabalhos.

3.1 Opinion mining from online hotel reviews - A text summarization approach

No trabalho de Hu, Chen e Chou (2017), estes propuseram uma técnica de sumarização de texto multi-documento que consegue gerar sumários a partir de um conjunto de opiniões sobre hotéis encontradas *online*.

O corpus utilizado possui opiniões coletadas do *website TripAdvisor* considerando quatro aspectos: credibilidade do autor; recência da opinião; utilidade da opinião, mensurada por meio da classificação da opinião frente a outras similares; e opiniões conflitantes, que incorporam análise de sentimentos para extrair mais informações das opiniões. As opiniões foram coletadas com base em dois hotéis, o *Red Roof Inn* e o *Gansevoort Meatpacking Hotel*, a data das opiniões coletadas varia entre 01/01/2012 e 31/03/2013. Devido à amplitude do *TripAdvisor* em relação a quantidade de linguagens e suporte entre plataformas, os autores optaram por considerar apenas opiniões em inglês, e de usuários do próprio *TripAvisor*.

As opiniões coletadas então passaram por uma sequência de passos de forma a prepará-las para serem recebidas pelo sumarizador. A fase de pré-processamento incluiu todas as etapas exemplificadas na subseção 2.2.4, com os autores utilizando o *tagger* de Porter (1980) e a lista de *stopwords* fornecida pelo MySQL[1]. Durante a filtragem de palavras, foram mantidas apenas aquelas que são substantivos, adjetivos e advérbios negativos nas sentenças. Após o

[1] http://dev.mysql.com/doc/refman/5.5/en/fulltext-stopwords.html

pré-processamento, cada sentença em uma opinião continha ao menos um substantivo e um adjetivo.

Feito o pré-processamento, em seguida foi realizado o cálculo da importância, similaridade e a seleção das k melhores sentenças para o sumário. A importância das sentenças foi calculada levando em conta a credibilidade do autor, a utilidade da opinião, a data de postagem dela e o conteúdo das sentenças. A similaridade das sentenças, por sua vez, é calculada entre pares de substantivos em cada duas sentenças com base no algoritmo *Normalized Google Distance* (NGD) de Cilibrasi e Vitanyi (2007), que fornece a quantidade de *hits* (resultados de busca) de palavras em plataformas de busca como *Google Search*. A similaridade também é calculada entre o conteúdo e o sentimento entre cada par de sentenças, que resultam na similaridade final entre as duas sentenças. A seleção das k melhores sentenças é feita através de uma adaptação do algoritmo *k-medoids* de Kaufmann e Rousseeuw (1987), que particiona as sentenças em p agrupamentos, com $p > k$. O algoritmo então seleciona p sentenças como um conjunto inicial de *medoids*, calculando a semelhança entre cada sentença e *medoid*, colocando as sentenças nos *medoids* mais próximos e criando novos agrupamentos a cada iteração até que eles fiquem estáveis.

Para verificar os resultados, o método proposto (abordagem C) foi comparado com duas outras formas de se determinar as k melhores sentenças. A primeira abordagem (abordagem A) trata todas as opiniões como um único documento e levando em conta apenas a importância das sentenças, enquanto que a segunda abordagem (abordagem B) também trata as opiniões como um único documento mas dessa vez as sentenças eram agrupadas em k agrupamentos e a sentença com a menor distância para cada *centroid*[2] era usada como a sentença representativa do agrupamento. A avaliação foi feita com base na geração de uma lista com as melhores sentenças de cada abordagem, na qual os avaliadores deveriam selecionar cegamente as sentenças mais úteis entre os três conjuntos, com a melhor abordagem recebendo 3 pontos, e a pior, 1. Em relação aos testes do sumário gerado, a abordagem C gerou pontuações consideravelmente mais altas que as outras, tendo uma média de 2.5 a 2.75 pontos comparado com as duas outras que raramente passavam de 2 pontos. Já os resultados de significância demonstraram que a pontuação de utilidade da abordagem C foi bastante superior ao das outras duas.

Concluindo, a abordagem de Hu, Chen e Chou (2017) consegue gerar resultados superiores às outras usadas como comparação por levar em conta a importância das sentenças, dos autores, a recência das opiniões coletadas, os sentimentos de opiniões contrastantes e a classificação delas. Isso a coloca em vantagem sobre as duas outras por estas realizarem apenas o pré-processamento do conteúdo textual, não levando em conta os outros vários fatores que caracterizam uma opinião.

[2] Um *centroid* em agrupamento de sentenças é o resultado da média aritmética da posição de todas as sentenças em um agrupamento.

3.2 Sumopinions: Sumarização Automática de Opiniões Sobre Pontos Turísticos

O objetivo de Junior (2018) foi desenvolver um método para sumarizar opiniões sobre vários pontos turísticos através de sites de hotelaria como o *TripAdvisor*, utilizando modelagem de tópicos e modelos probabilísticos não supervisionados. A ideia então é gerar um sumário destas opiniões através de uma sumarização extrativa baseada em tópicos, facilitando a obtenção dessas informações de forma mais condensada e informativa.

Foram coletadas várias análises do *TripAdvisor*, cujas informações são separadas em três categorias para os autores das opiniões: Nível do autor, Quantidade de opiniões postadas, e Quantidade de agradecimentos (*likes*) recebidos. Enquanto que as opiniões são agrupadas em cinco categorias: Título, Data, Nota (que varia de uma a cinco estrelas), Opinião e Quantidade de agradecimentos recebidos.

As informações acima são então pré-processados pelas várias técnicas de PLN explicadas na subseção 2.2.4 para gerar um conteúdo que facilite a sumarização. Em seguida é feita a modelagem e descoberta dos tópicos por meio do algoritmo LDA, combinado com TF-IDF para gerar um vetor de frequência de palavras.

A importância das sentenças é calculada de acordo com a representatividade do usuário, importância da opinião (calculada com base na quantidade de recomendações recebidas em relação às outras opiniões), o quão recente é a opinião, e o conteúdo das sentenças em uma opinião. Essa importância afeta diretamente a seleção das sentenças para o sumário pois leva em conta a credibilidade do autor, o seu *Score* de recomendação, sua representatividade, a utilidade de sua opinião e outros aspectos. A sumarização, por fim, atua sobre essas sentenças com o algoritmo não supervisionado *k-medoids* (KAUFMANN; ROUSSEEUW, 1987) visto anteriormente no trabalho de Hu, Chen e Chou (2017).

Os experimentos se baseiam em dois pontos turísticos distintos, a Torre Eiffel de Paris e o Parque Central de Nova Iorque, sendo verificada a similaridade do conteúdo das sentenças para julgar o que deve ser colocado nos sumários. Essa similaridade é calculada por meio da NGD (CILIBRASI; VITANYI, 2007), que, como visto no trabalho correlato anterior é algo que demanda bastante tempo, o que acabou limitando a coleta de dados e incorrendo em um maior tempo de preparação.

Os resultados obtidos pelo SumOpinions, em geral, são melhores que os das abordagens utilizadas por (HU; CHEN; CHOU, 2017) no quesito de cobertura de tópicos, o que a princípio indica um melhor desempenho na seleção de sentenças. Por outro lado, o SumOpinions possui quase o dobro da redundância em relação aos sumários do algoritmo de referência, mas mesmo assim o valor de redundância de ambos os modelos é bastante baixo. Após a análise dos resultados, foi concluído pelo autor que o SumOpinions pode ser considerado mais preciso que

os métodos usados por Hu, Chen e Chou (2017).

3.3 Uma abordagem de sumarização automática de textos aplicadas a debates online

Simonassi *et al.* (2016) propôs um modelo de sumarização de debates *online* em fóruns utilizando sumarização automática de textos. Segundo o autor, essa escolha foi feita devido à sumarização de debates permitir a visualização do conteúdo de forma condensada e simplificada, facilitando a entrada de novos participantes e ampliando a democracia participativa durante os debates.

As amostras dos debates foram coletadas com a ferramenta Dialoguea apresentada por Seilles (2012), uma evolução da Argumentea, também de Seilles, e que permite selecionar um tema a ser debatido e receber informações sobre ele e os argumentos apresentados pelos usuários *online*. A ferramenta apresenta uma descrição do tema no lado esquerdo da interface para que os participantes possam entendê-lo e argumentar sobre. O lado direito da ferramenta é onde os argumentos dos participantes são visualizados, sendo possível ver se eles são apoiados por argumentos de outros participantes. Essas informações são então extraídas pelo autor para formar a base de dados, que foi criada a partir do tema "Fórum de debates, anotação de debates digitais e sua ética".

Os experimentos começam com o autor aplicando as etapas de pré-processamento e processamento dos textos com algumas adaptações, com o pré-processamento sendo bastante similar ao conjunto de passos utilizados por Junior (2018). O processamento em si, por outro lado, é baseado inteiramente na técnica TF-IDF para gerar o sumário final com os argumentos mais importantes presentes. Feito o processamento, a pontuação das sentenças no sumário é calculada por meio de três técnicas, a *Highest Term Sentence Score* (HTSS), a *Summing Term Sentence Score* (STSS) e a *Standard Deviation Sentence Score* (SDSS). A HTSS atribui a uma sentença o maior valor TF-IDF do conjunto onde seus termos se encontram, a STSS atribui a uma sentença a somatória dos valores TF-IDF de todos os seus termos no conjunto, e a SDSS atribui a somatória do TF-IDF dos termos da sentença que possuem um desvio padrão de 1 em relação aos outros termos.

A primeira análise de resultados realizada foi a respeito das sentenças em comum às técnicas de seleção presentes no trabalho, onde a HTSS e STSS tiveram o maior percentual de similaridade. Em seguida foi avaliada a dispersão dos argumentos por participantes e a precisão das técnicas em selecionar esses argumentos, com a STSS se saindo melhor nessa categoria. O último quesito avaliado foi a proporção de sentenças em comum selecionadas pelos sumarizadores automáticos e pelos avaliadores humanos em seus sumários manuais, onde o destaque foi para as técnicas STSS e HTSS em geral.

No fim, Simonassi *et al.* (2016) concluiu que a Dialoguea é a ferramenta que melhor atende às características importantes em uma ferramenta de análise de debates quando comparada a outras similares como a Argumentea. Além disso, foi concluído as técnicas HTSS, STSS e SDSS possuem bastante potencial para serem utilizadas em debates *online* devido aos resultados obtidos.

3.4 Sumarização Automática de opiniões baseada em aspectos

Em seu trabalho, Condori (2014) teve o objetivo de desenvolver métodos de sumarização de opiniões para o português brasileiro, tanto por sumarização extrativa quanto abstrativa, e também comparar os sumários gerados. Essa comparação foi feita em diferentes métricas como informatividade, utilidade, qualidade linguística e facilidade de leitura com base em um corpus de opiniões em Português do Brasil, além das métricas ROUGE.

Devido à cobertura de tópicos do trabalho, o corpus utilizado pelo autor possui dados provenientes de duas fontes distintas: O corpus ReLi de Freitas *et al.* (2012), e coletas manuais de comentários sobre quatro produtos no *site* Buscapé. Foram utilizados dois métodos para cada tipo de sumarização, além de um método customizado por Condori em cada tipo e um sumarizador extra chamado Rsumm Ribaldo, Akabane e Pardo (2012). Para o modelo extrativo, foram considerados os modelos de Hu e Liu (2004) e Tadano, Shimada e Endo (2010). Em relação ao modelo abstrativo, são contemplados os métodos de Ganesan, Zhai e Han (2010) e o de Gerani *et al.* (2014). Condori também desenvolveu duas técnicas extras adaptadas dos conhecimentos recentes da área, sendo nomeados Opizer-E e Opizer-A, em referência aos termos extrativo e abstrativo, respectivamente.

Os sumários foram avaliados com base em três tipos de métodos, relacionados aos aspectos de: informatividade do sumário, a qualidade linguística e a utilidade. A informatividade é analisada pelo conjunto ROUGE (LIN, 2004), que compara a similaridade entre um sumário automático e um ou mais sumários humanos, que são a referência. A qualidade linguística por sua vez é mensurada por assessores humanos que seguem cinco critérios linguísticos: gramaticalidade, não redundância, clareza referencial, foco e estrutura, e coerência (DANG, 2005). A avaliação de utilidade foi feita perguntando aos leitores do artigo sobre o quão útil ele foi em auxiliá-los em uma situação de decisão.

Os resultados obtidos na ROUGE demonstram que o melhor valor das medidas-F (equações para tipos ROUGE vistas na subseção 2.3.1) foram obtidas pelo Opizer-E na ROUGE-1, ROUGE-2 e ROUGE-L, indicando que a estratégia de seleção de aspectos do Opizer-E foi a mais apropriada. Já para os métodos abstrativos o pior método avaliado foi o de Gerani *et al.* (2014) por causa da rigidez dos *templates* utilizados que não possuem muitas palavras comuns com as opiniões fonte, e aqui o Opizer-A obteve resultados levemente superiores aos outros.

Nestes testes, o Rsumm teve resultados aquém dos demonstrados pelo Opizer-E em todas as categorias. As avaliações de qualidade linguística e utilidade foram realizadas por 26 pessoas com conhecimento de PLN. Elas tiveram de ler os sumários produzidos e avaliá-los conforme as metodologias explicadas anteriormente. Nesse quesito, tanto Opizer-E e Opizer-A tiveram resultados razoáveis frente aos outros métodos.

Ao fim de tudo, foi concluído que os algoritmos propostos por Condori podem ser considerados como evoluções de outros modelos tais como os de Hu e Liu (2004), por exemplo. Embora o desempenho tenha regredido em certos aspectos, o desempenho geral deles ainda é consistentemente superior aos das outras técnicas contempladas pelo trabalho.

3.5 Unsupervised Opinion Summarization as Copycat-Review Generation

O objetivo do trabalho de Bražinskas, Lapata e Titov (2019) é realizar a tarefa de sumarização de opiniões através de um modelo gerador capaz de coletar opiniões. Esse modelo se aproveita da intuição de que, ao gerar um resumo de opiniões de um produto, a quantidade de informações novas e de desvio do tema devem ser controláveis.

O modelo de sumarização latente dos autores, denominado *Copycat*, captura a organização hierárquica dos agrupamentos de opiniões do corpus, com cada agrupamento contendo opiniões sobre um mesmo produto. O *Copycat* utiliza dois conjuntos de variáveis latentes, onde uma variável constante c é atrelada a cada agrupamento de opiniões para capturar as "semânticas latentes" dele. Outros aspectos relacionados à modelagem hierárquica como probabilidade logarítmica e estimativa de pseudo-probabilidade também são considerados no modelo. Após realizar todos os ajustes no modelo geral, são desenvolvidos os componentes do mesmo, que auxiliam o modelo na tarefa do processamento de textos.

Após todas as preparações e treinamento dos componentes, é feita a geração do sumário, que é realizada com base em um *dataset* de *reviews* e opiniões do Amazon, e outro do *Yelp*. Esse segundo conjunto de dados exigiu mais ajustes para remover informações pessoais e irrelevantes. Para avaliar o modelo, eles utilizaram 100 resumos criados por humanos, disponibilizados em (CHU; LIU, 2019).

Os resultados obtidos na avaliação automática apontaram que o algoritmo *Copycat* dos autores é superior a outros como o LexRank (ERKAN; RADEV, 2004), Opinosis (GANESAN; ZHAI; HAN, 2010), MeanSum (CHU; LIU, 2019) e *Variational AutoEncoder* (VAE) (BOWMAN *et al.*, 2015) em ambos os *datasets* nas três métricas ROUGE testadas (ROUGE-1, ROUGE-2 e ROUGE-L). Já a avaliação humana pelo método *Best-Worst Scaling* (LOUVIERE; FLYNN; MARLEY, 2015) mostrou que embora o *Copycat* seja melhor em geral, ele ainda é inferior ao LexRank em alguns quesitos como "Consenso de Opiniões", por exemplo.

Concluindo, os resultados demonstraram que o modelo apresentado por Bražinskas, Lapata e Titov (2019) alcança bons resultados quando comparados com outros sumarizadores, especialmente em relação ao único outro modelo capaz de sumarizar múltiplos documentos usado na comparação. O modelo dos autores se saiu melhor por conta da capacidade dele de gerar um resultado que reflete com mais precisão o conteúdo dos textos recebidos na entrada.

3.6 A novel concept-level approach for ultra-concise opinion summarization

Lloret *et al.* (2015) buscaram, no estudo, apresentar uma abordagem de sumarização de opiniões que é capaz de gerar sumários ultra-concisos. Para tal, ela se utiliza de um método abstrativo e a nível de conceitos.

O corpus desenvolvido para o sumarizador utilizou o *crawler* criado por Fernández, Gómez e Barco (2010) para coletar opiniões presentes em páginas como *Amazon*, *WhatCar* e *Twitter*. O foco da coleta incidiu sobre opiniões em inglês, sendo selecionados os telefones e carros localizados no topo dos resultados retornados pelo *crawler*, resultando em dez documentos de cada *site* e domínio.

Para desenvolver a abordagem de sumarização, os autores se basearam nos procedimentos de sumarização dados por Jones (1999), que especifica três fases gerais: Interpretação, Transformação, e Geração do Sumário. A primeira fase trata da interpretação e entendimento do texto de entrada, a segunda transforma esse texto de entrada em uma representação interna para extrair melhor as ideias principais, e a terceira determina qual o conteúdo mais relevante a ser colocado no sumário final.

Estas três fases foram adaptadas pelo autor em segmentos menores e mais claros: Simplificação sintática de textos, que reduz sentenças longas em duas ou mais sentenças menores e mais simples, simplificando as árvores de dependência entre elas; Análise de textos, que se encarrega de analisar as sentenças geradas no estágio anterior, extraindo tríplices sujeito-verbo-objeto, entidades nomeadas e frases com substantivos claramente especificados; Representação de conceitos, que retrabalha o conteúdo extraído anteriormente para refletir melhor as características deles para gerar uma sentença com informações válidas; Análise e sumarização de Conceitos, responsável por analisar os conceitos gerados no estágio anterior, levando em conta o conteúdo compartilhado por eles e classificando-os para então sumarizá-los; Representação de superfícies, onde os conceitos selecionados na etapa anterior são convertidos para texto puro; e Seleção de sentenças, que escolhe as sentenças com a melhor colocação após a classificação e tradução dos conceitos em representações textuais.

Os experimentos foram realizados com o sumarizador em quatro configurações distintas: $S+SVO+TFNE$, que gera uma sentença simplificada durante a fase de representação de super-

fícies e faz a pontuação dela conforme a frequência das palavras durante a fase de sumarização; $S+SVO$, que também gera uma sentença simplificada, mas a pontua apenas de acordo com o conteúdo compartilhado entre os conceitos obtidos; $R+SVO+TFNE$, que é similar à primeira configuração, mas efetua a re-geração da sentença durante a representação de superfícies usando conhecimentos de PLN; $R+SVO$, que re-gera a sentença durante a representação de superfícies, e a pontua conforme o conteúdo compartilhado entre os conceitos.

A avaliação realizada foi qualitativa, e consistia em três *experts* sendo encarregados de avaliar independentemente cada uma das configurações do sumarizador. Os critérios usados na avaliação foram o conteúdo do sumário, a legibilidade do mesmo, e a responsividade dele (o quão confiável e adequado ele é para ser aplicado em situações reais).

Os resultados obtidos mostraram que as configurações que utilizavam geração de língua natural se saíram ligeiramente melhor segundo os avaliadores. Já sobre os domínios de telefones e carros, as configurações exibiram todas a mesma tendência, com as informações específicas do domínio não afetando a performance do sistema. Já o domínio de telefones continha muitos anúncios e era composto de opiniões mais curtas e menos detalhadas que as do domínio de carros. Em relação à sumarização de *tweets*, o sistema dos autores foi comparado com outras abordagens como a de Ganesan, Zhai e Viegas (2012), e gerou resultados superiores em ambos os domínios.

Concluindo, os resultados obtidos pelo sumarizador de Lloret *et al.* (2015) foram superiores em relação a outras abordagens voltadas à sumarização de opiniões mais compactas e concisas, como é o caso das opiniões postadas no *Twitter*. A possibilidade de se configurar o sumarizador deles de quatro formas diferentes também aumenta a flexibilidade do mesmo e o permite lidar melhor com opiniões maiores e mais detalhadas.

3.7 Query-Focused Opinion Summarization for User Generated Content

O trabalho de Wang *et al.* (2016) teve como objetivo propor um sumarizador baseado em consultas que é capaz de sumarizar diferentes tipos de conteúdos gerados por usuários, como *blogs* ou perguntas e respostas no *Yahoo!*. Para alcançar essa meta, eles desenvolveram um *framework* de sumarização de opiniões sub-modular baseado em funções. Para tal, eles definem uma série de funções que compõem a função objetivo: Função de Relevância, Função de Dispersão, e Funções de Cobertura.

A Função de Relevância utiliza ranqueamento estatístico para gerar uma ordenação preferencial das sentenças, embora eles expliquem que usar TF-IDF ou sobreposição de palavras chave resulta em ordens similares. A aplicação da Função de Dispersão visa minimizar a redundância das informações, onde os autores testaram várias funções diferentes como Dissimilaridade

Léxica, Dissimilaridade Semântica e Dissimilaridade Tópica. Já as funções de Cobertura são compostas por Cobertura de Tópicos, Cobertura de Autoria, Cobertura de Polaridade e Cobertura de Conteúdo, buscando maximizar os aspectos abordados pelas sentenças. O sumário é então criado por um algoritmo guloso que visa gerar uma aproximação da solução ótima, já que criar um sumário que maximiza a função objetivo dos autores é um problema NP-Difícil (CHANDRA; HALLDÓRSSON, 1996).

Para a experimentação foram utilizados dois *datasets* distintos, um contendo perguntas e respostas comunitárias do *Yahoo!*, e o outro contendo conteúdos similares, mas em blogs, apresentado na TAC 2008 por Dang e Owczarzak (2008). Foi construído um classificador para detectar perguntas que são orientadas a gerar respostas opinativas no primeiro *dataset* (*Yahoo!*), algo que não foi necessário para o outro, pois o conteúdo deste é garantidamente opinativo segundo os autores. Eles comparam o sumarizador desenvolvido por eles com as abordagens de Dasgupta, Kumar e Ravi (2013) e os sistemas criados por Lin e Bilmes (2011), usando a resposta mais votada pelos usuários como parâmetro no *dataset* comunitário do *Yahoo!*. Já o *dataset* da TAC 2008 leva em conta também as principais sentenças selecionadas pelo ranqueador dos autores, os melhores sistemas de sumarização apresentados na TAC 2008 (Kim *et al.* (2008); Li, Liu e Agichtein (2008)), e uma linha de base gerada por TF-IDF e um *lexicon*.

Os resultados foram avaliados em uma série de quesitos, levando em conta o ranqueamento, a sumarização feita no conjunto de dados comunitário com base em avaliadores automáticos e humanos, e a sumarização realizada no conteúdo dos *blogs* também utilizando os mesmos avaliadores. O método de avaliação automática *Jensen–Shannon Divergence* (JSD) (DAGAN; LEE; PEREIRA, 1997) e um método de avaliação apoiada em avaliadores humanos foram utilizados. Os resultados obtidos pelo ranqueador dos autores foi superior a todos os três outros ranqueadores usados como base. Já a avaliação feita nas opiniões comunitárias retornou que os sumarizadores de Dasgupta, Kumar e Ravi (2013) e o dos autores foi superior ao de Lin e Bilmes (2011) na avaliação automática, enquanto que os avaliadores humanos preferiram a abordagem dos autores por gerar sumários com sugestões mais úteis e com mais opções. A avaliação automática das opiniões em *blogs* mostrou que o sumarizador de Wang *et al.* (2016) foi um dos melhores em praticamente todas as métricas, enquanto que a humana reportou que o mesmo sumarizador era o que continha menos redundância nas informações.

Ao fim de tudo, foi concluído que o *framework* baseado em funções de Wang *et al.* (2016) obteve resultados melhores que todos os outros sumarizadores utilizados para comparação. Além disso, o *framework* deles também consegue utilizar informações estatísticas para encontrar a relevância das sentenças e encorajar o sumário a cobrir uma gama de tópicos bastante diversa.

3.8 Opinion Mining and Summarization of Hotel Reviews

O objetivo de Raut e Londhe (2014) foi minerar e sumarizar opiniões de hotéis utilizando abordagens supervisionadas com aprendizado de máquina, e uma técnica baseada no método *SentiWordNet* (ESULI; SEBASTIANI, 2006) para classificar as opiniões como positivas ou negativas. A sumarização em si é realizada por meio de um *framework* que pode ser utilizado para mineração, recuperação e sumarização de opiniões.

O corpus dos autores é constituído de opiniões retiradas do *website TripAdvisor*, tomados os devidos cuidados para permitir o treinamento dos modelos supervisionados. As opiniões foram classificadas como positivas e negativas por classificadores que também utilizam aprendizado de máquina e um algoritmo baseado no *SentiWordNet*. As opiniões classificadas foram então pré-processadas, com a pontuação das sentenças ocorrendo em seguida. Foram usadas 1000 opiniões para treinamento e teste dos modelos com aprendizado de máquina, com metade delas sendo composta de opiniões negativas, e a outra metade de positivas.

As abordagens com aprendizado de máquina usados pelos autores foram *Naive Bayes*, *Support Vector Machine* e *Decision Tree*. Já o *SentiWordNet* é um conjunto de recursos léxicos para análise de sentimentos, que calcula a polaridade das palavras com base nas palavras já classificadas que ele possui internamente. Para a sumarização das opiniões, o *framework* utilizado foi o de Lloret *et al.* (2012), que trabalha pontuando as sentenças através da frequência dos termos e da relevância delas, similar ao trabalho de Wang *et al.* (2016).

Os resultados obtidos pelos autores levaram em conta apenas os classificadores, pois o sumarizador foi utilizado sem quaisquer alterações sobre o documento base. Em relação à acurácia das classificações, o *Naive Bayes* atingiu um valor de 88%, o *Support Vector Machine* conseguiu 83.5%, a *Decision Tree* atingiu 78.4%, e o *SentiWordNet* alcançou 87.6%. Já a precisão delas foi 0.895, 0.844, 0.768 e 0.90, respectivamente. Os resultados de *Recall* foram 0.890, 0.835, 0.784 e 0.876. Por fim, a medida-F de cada uma delas foi 0.879, 0.834, 0.784 e 0.888.

Em conclusão, os sistemas de Raut e Londhe (2014) obtiveram até mais que 87% de acurácia na classificação da polaridade das opiniões com o modelo *Naive Bayes*. Esse resultado mostra que as técnicas de classificação de opiniões por aprendizado de máquina estão se tornando bastante precisas.

3.9 A Comparative Study of Opinion Summarization Techniques

O trabalho de Bhatia (2021) teve o objetivo de comparar uma abordagem extrativa e outra abstrativa de forma a encontrar qual delas era capaz de gerar sumários mais coerentes e completos. A abordagem abstrativa trabalha através da geração de grafos enquanto que a extrativa utiliza uma metodologia nova desenvolvida para o trabalho em questão.

A abordagem extrativa utiliza o princípio de *Principal Component Analysis* (PCA) para aplicar uma redução do número de dimensões dos aspectos e ranquear as opiniões de acordo com a relevância delas aos principais aspectos encontrados. O PCA (JOLLIFFE, 2011) é um método estatístico que visa reduzir a dimensão do *dataset* e a correlação entre as variáveis presentes. Esse método é capaz de identificar padrões presentes nas informações e representá-las de forma a realçar as similaridades e dissimilaridades encontradas. Como o PCA pode ser aplicado a uma variedade de tarefas, a inclusão dele em uma abordagem de sumarização extrativa se dá devido à possibilidade de se selecionar as sentenças e opiniões mais importantes diretamente por ela.

Já a abordagem abstrativa é baseada na metodologia de Ganesan (2013), mas com o diferencial de inserir análise de sentimentos por meio de uma *Convolutional Neural Network* (CNN). Essa CNN é utilizada para conferir pontos às sentenças com base na fusão dos sentimentos atrelados a elas.

As abordagens foram aplicadas em três *datasets* e tiveram seus resultados comparados através do conjunto ROUGE, mais especificamente a ROUGE-1 e ROUGE-2 (implementações da ROUGE-N para uni-gramas e bigramas respectivamente). Após as avaliações, foi constatado que a abordagem abstrativa, embora superior à extrativa nas categorias "Precisão" e "F-Measure", acaba sendo inferior na categoria "recall". Isso se dá ao costume do método extrativo de apenas selecionar as sentenças sem realizar alterações no corpo delas, o que aumenta a quantidade de texto em comum entre o sumário de referência e o candidato.

Por fim, foi concluído que a abordagem abstrativa é capaz de gerar sumários mais concisos e relevantes se comparada à abordagem extrativa, algo corroborado pelos resultados da ROUGE-1 e ROUGE-2.

3.10 Aspect based Sentiment Oriented Summarization of Hotel Reviews

Akhtar *et al.* (2017) tiveram como objetivo estudar as informações presentes nas opiniões e classificá-las. Também é utilizado o LDA para identificar informações ocultas e análise de sentimentos para adquirir a polaridade das sentenças.

As opiniões coletadas foram classificadas inicialmente com base nos aspectos mais recorrentes, obtidos através de uma inspeção manual dos dados. A modelagem de tópicos foi realizada por meio da ferramenta *Machine Learning for Language Toolkit* (MALLET) [3], com o número de tópicos sendo inicialmente assumido como 15. Após a modelagem de tópicos, foi realizada a análise de sentimentos, implementada com ajuda do corpus *SentiWordNet*. O foco da análise de sentimentos foi em analisar as sentenças dentro dos tópicos já classificados anteriormente para calcular a polarização delas, o que é facilitado pela organização das sentenças

[3] https://mimno.github.io/Mallet/topics.html

em um arquivo para cada aspecto.

A sumarização, por sua vez, é baseada no resultado obtido pelos processos de classificação, selecionando as sentenças polarizadas e neutras mais importantes encontradas para cada aspecto encontrado. Apenas as três sentenças com maior pontuação de cada categoria (positivas, negativas e neutras) foram escolhidas, com essa pontuação sendo o agregado das pontuações que elas obtiveram em cada etapa de classificação.

Não foi realizado nenhum teste de acurácia ou similaridade através de métricas como a ROUGE pois os autores não buscaram comparar os resultados com o de outras técnicas, focando em mostrar apenas a razão entre pontuações negativas e positivas para cada aspecto, além de um exemplo das sentenças mais importantes para um dos aspectos.

3.11 Extractive Summarization as Text Matching

Zhong *et al.* (2020) tiveram, em seu trabalho, o intuito de gerar sumários com base em uma técnica que seleciona sentenças através da busca por correspondências entre um sumário de referência e diferentes sumários candidatos. Para isso, o trabalho de sumarização foi formulado como um problema de correspondência, seguindo a intuição de que um bom sumário tem que ser semelhante semanticamente ao documento fonte, e não a sumários de baixa qualidade deste.

Antes de dar início ao desenvolvimento do sumarizador, foi realizado um estudo para descobrir se era mais eficaz atuar a nível de sentença ou a nível de *dataset* (também referido como "nível de sumário"). Foi concluído que sumarização a nível de *dataset* se sai melhor que a sumarização a nível de sentença, com o tamanho dos sumários gerados tendo uma forte correlação com a vantagem encontrada, onde sumários de tamanho médio (aproximadamente 60 palavras) se beneficiavam da sumarização a nível de *dataset*.

Com esta informação em mãos, foi projetado um algoritmo capaz de extrair sumários e pontuá-los diretamente. Isso é feito por meio de um problema de correspondência onde o documento fonte e os sumários candidatos são comparados em um espaço semântico. O algoritmo então utiliza uma arquitetura composta de dois sistemas BERT (DEVLIN *et al.*, 2018) para verificar a correspondência entre os documentos e os sumários candidatos. Essa configuração é chamada de BERT-siamês, por ser baseada na estrutura de redes siamesas de Bromley *et al.* (1993). A ideia é que o sumário candidato tenha o maior valor de correspondência em relação aos outros sumários concorrentes gerados.

Foram realizados experimentos em seis *datasets* distintos, o primeiro composto por textos jornalísticos retirados das páginas *web* da *Cable News Network* (não confundir com *Convolutional Neural Network*) (CNNt) e *Daily Mail* (DM), o segundo por textos curtos retirados do *Reddit*, o terceiro por textos do *XSum* (NARAYAN; COHEN; LAPATA, 2018), o quarto por conteúdo da *WikiHow*, o quinto por publicações da *PubMed* e por fim, o sexto por notícias

Multi-News.

Os resultados obtidos demonstraram que o *MatchSum* superou outras técnicas do estado da arte em todos estes seis *datasets* distintos nas métricas ROUGE, o que mostra a eficácia do método desenvolvido por Zhong *et al.* (2020).

3.12 Text Summarization with Pretrained Encoders

O objetivo de Liu e Lapata (2019) em seu trabalho foi desenvolver um sumarizador automático que se utiliza de *encoders* pré-treinados com base no modelo BERT de Devlin *et al.* (2018), sendo capaz de realizar sumarizações extrativas e abstrativas de vários tipos de conteúdos textuais. Para isso, foi desenvolvido um *framework* geral e um *encoder* baseado no BERT que atua a nível de documento, capaz de codificar documentos e obter as representações de suas sentenças.

Foi aplicado um *fine-tuning* no modelo para sumarização de textos através do uso de *embeddings* posicionais para indicar a posição de cada sentença, do otimizador *Adaptive Moment Estimation* (Adam), além da configuração de um número específico de camadas do *transformer*, e de uma taxa de aprendizado idêntica à encontrada no trabalho de Vaswani *et al.* (2017). A abordagem de *fine-tuning* foi dividida em duas etapas: A primeira voltada a tarefa sumarização extrativa, e a segunda a sumarização abstrativa, pois trabalhos como os de Dong *et al.* (2019) e Gehrmann, Deng e Rush (2018) sugerem que tarefas de sumarização abstrativa se beneficiam da utilização de objetivos extrativos.

O modelo foi avaliado em três *datasets* distintos, o primeiro sendo um *dataset* contendo textos jornalísticos dos *sites* CNNt e *Daily Mail*, o segundo contendo textos do *The New York Times* (NYT) e o terceiro sendo o corpus XSum de Narayan, Cohen e Lapata (2018). As repartições para treinamento, validação e testes foram de (90.266/1.220/1.093 e 196.961/12.148/10.397) para o corpus CNNt/DM, (100.834/4.000/3.452) para o corpus NYT após filtragens e (204.045/11.322/11.334) para o corpus XSum. Para comparar os resultados obtidos, foram utilizadas as métricas Rouge-1, Rouge-2 e Rouge-L.

Os resultados obtidos nas três métricas, Rouge-1, Rouge-2 e Rouge-L, respectivamente, para o modelo combinado foram de 42.13, 19.60 e 39.18 no corpus CNNt/DM, 49.02, 31.02 e 45.55 no corpus NYT e por fim 38.81, 16.50 e 31.27 no corpus XSum. Estes resultados demonstram que a técnica de Liu e Lapata (2019) é bastante flexível e capaz de atingir resultados similares ao estado da arte em diferentes tipos de sumário, e documentos fonte.

3.13 Extractive Summarization Using Supervised and Semi-supervised Learning

O objetivo do trabalho de Wong, Wu e Li (2008) foi apresentar um sumarizador extrativo que trata a tarefa de se realizar uma sumarização extrativa como um problema de classificação. Este gera sumários através de classificadores como SVM e *Naive Bayes* co-treinados sob um mesmo corpus para extrair características de documentos e sentenças.

Abordagens supervisionadas, segundo os autores, costumam se sair melhor que abordagens semi-supervisionadas em tarefas de classificação, entretanto, necessitam de uma quantidade muito maior de informações rotuladas para tal. Por conta disso, os autores propuseram a utilização de dois classificadores no mesmo *dataset* de forma a preencher a parte não rotuladas das informações com base nos resultados de classificação dos mesmos, utilizando a parte já rotulada para treinamento.

Foram definidas quatro características para as sentenças presentes no corpus: *surface* (com base na estrutura delas), *content* (palavras centrais), *event* (associação entre termos e elementos citados por eles) e *relevance* (relacionamento intra-sentencial). Devido a esse número limitado de características, o co-treinamento no corpus utilizou dois classificadores distintos, *Probabilistic Support Vector Machine* (PSVM), e *Naive Bayes Classifier* (NBC) em uma estratégia similar a de Blum e Mitchell (1998) para rotular sentenças adicionais.

Para a sumarização em si, foi utilizada a implementação de SVMs dada pela biblioteca LibSVM[4], pois segundo os autores os classificadores SVM costumam se sair melhor nessas tarefas quando comparados com os NBCs. Os experimentos foram realizados nos *datasets* da DUC 2001 e 2004, com o *dataset* de 2001 tendo 25 de seus 30 agrupamentos de documentos relevantes sendo usados para treinamento, e apenas 5 para testes.

O melhor desempenho de classificação foi obtido usando uma combinação *surface* + *content* + *relevance* de características, e o mesmo conjunto de características foi o que se saiu melhor nos resultados das métricas ROUGE. Por conta disto, os autores constataram que as características do tipo *event* não possuíram um impacto positivo, com a justificativa sendo a de que isso foi causado pela ausência de características deste tipo em alguns agrupamentos.

Em relação ao co-treinamento, foi constatado que seu desempenho é superior aos dois classificadores atuando de forma separada quando a quantidade e sentenças rotuladas é pequeno (foram utilizadas 2000). A conclusão obtida demonstra que abordagens de classificação semi-supervisionadas possuem o potencial de serem tão robustas quanto as supervisionadas, mas com a vantagem de não precisarem de uma quantidade tão grande de sentenças e documentos rotulados.

[4] http://www.csie.ntu.edu.tw/~cjlin/libsvm/

3.14 Sentence Centrality Revisited for Unsupervised Summarization

Zheng e Lapata (2019) apresentaram um sistema de sumarização não-supervisionado que necessita de menos dados que outros similares e que também pode lidar com diferentes domínios e línguas, além trazer melhorias para o conceito de centralidade de sentenças através de uma melhor captura de significados sentenciais e similaridade de sentenças através de um modelo BERT, e a utilização de arestas direcionadas pois a contribuição de dois vértices para suas respectivas centralidades pode não ser a mesma.

Para calcular a centralidade das sentenças, foi utilizado o BERT para computar a similaridade entre os pares de sentenças por meio do mapeamento destas em representações contínuas, além de aplicar uma hipótese de distribuição de sentenças similar à de Polajnar, Rimell e Clark (2015) para o treinamento do modelo, com a matriz de similaridade sendo gerada através do produto escalar entre as sentenças ao invés da distância cosseno entre elas.

Foram utilizados dois corpora nos experimentos, com estes sendo o corpus CNNt/DM e NYT citados anteriormente. A repartição de documentos para treinamento, validação e testes no corpus CNNt/DM foi de (90.266/1.220/1.093) documentos da CNNt e (196.961/12.148/10.397) documentos do *Daily Mail*. Já para o corpus NYT, a repartição foi de (589.284/32.736/32.739) documentos.

A análise dos resultados foi feita através das métricas ROUGE, mais especificamente as métricas ROUGE-1, ROUGE-2 e ROUGE-L. Os resultados obtidos no corpus CNNt/DM foram 40.7, 17.8 e 36.9 com o PACSUM-BERT, e no corpus NYT os resultados foram 41.4, 21.7 e 37.5.

Estes resultados apontam que o sistema dos autores, mesmo sendo mais geral e necessitando de menos dados, possui performance superior a uma série de outros sistemas análogos que entretanto precisam ser preparados para domínios e línguas específicas para operarem bem, além de exigirem mais dados para treinamento e validação.

Tabela 1 – Tabela de trabalhos correlatos

Autor(es)	**Técnica(s) base**	**Sumarização**	**Tipo de dados**	**Tamanho do Dataset**	**Distribuição do Dataset (Treinamento / Teste)**	**Medida de Desempenho**
(HU; CHEN; CHOU, 2017)	K-Medoids	Extrativa	Opinativos	Não informado	Não foi necessária	Avaliação humana
(JUNIOR, 2018)	K-Medoids	Extrativa	Opinativos	Não informado	Não foi necessária	ROUGE & Avaliação humana
(SIMONASSI *et al.*, 2016)	HTSS & STSS & SDSS	Extrativa	Opinativos	Não informado	Não foi necessária	Avaliação humana
(CONDORI, 2014)	K-Means & K-Medoids & Opizer-A & Opizer-E	Extrativa & Abstrativa	Opinativos	40 / 130	Não foi necessária	ROUGE & Avaliação humana
(BRAŽINSKAS; LAPATA; TITOV, 2019)	Opinosis & LexRank & MeanSum & VAE	Extrativa & Abstrativa	Opinativos	1,125,653 & 4,807,338	(1,012,280 / 113,373) & (4,566,519 / 240,819)	ROUGE & Avaliação humana
(LLORET *et al.*, 2015)	(R + SVO) & (R + SVO + TFNE) & (S + SVO) & (S + SVO + TFNE)	Abstrativa	Opinativos	(220 & 200)	Não foi necessária	Original (baseada em critérios da DUC/TAC)
(WANG *et al.*, 2016)	Original (*Query-based*)	Extrativa & Abstrativa	Opinativos	130,609	(92,109 / 38,500)	JSD & Avaliação humana
(RAUT; LONDHE, 2014)	Original (apoiado em *SentiWordNet*)	Extrativa	Opinativos	2000	(1000 / 1000)	Não informado (similar à ROUGE)

Tabela continua na próxima página...

Autor(es)	**Técnica(s) base**	**Sumarização**	**Tipo de dados**	**Tamanho do Dataset**	**Distribuição do Dataset (Treinamento / Teste)**	**Medida de Desempenho**
(BHATIA, 2021)	Original (Uma baseada em PCA & e outra baseada em Ganesan (2013))	Extrativa & Abstrativa	Opinativos & Factuais/Científicos	5100 & 50 & Não informado	Não foi necessária	ROUGE
(AKHTAR *et al.*, 2017)	Original (baseada no resultado de classificação das sentenças)	Extrativa	Opinativos	1200	Não foi necessária	Avaliação humana
Zhong *et al.* (2020)	Original (com adaptações do modelo de Devlin *et al.* (2018)	Extrativa	Opinativos & Factuais/Científicos	(92.579 + 219.506), 122.933, 133.000, 230.843, 226.711, 56.216,	Não informada	ROUGE
Liu e Lapata (2019)	Original (com adaptações do modelo de Devlin *et al.* (2018) e modelos pré-treinados	Extrativa & Abstrativa	Factuais/Científicos	(92.579 + 219.506), 654.759, 226.711	(90.266 / 1.220 / 1.093 e 196.961 / 12.148 / 10.397), (100.834 / 4.000 / 3.452), (204.045 / 11.322 / 11.334)	ROUGE
Wong, Wu e Li (2008)	Original (baseada em SVM e Naive Bayes)	Extrativa	Factuais/Científicos	Em agrupamentos de documentos: 30, 50	Em agrupamentos: (25/5), não informado	ROUGE
Zheng e Lapata (2019)	Original (baseada no modelo BERT de Devlin *et al.* (2018)	Extrativa	Factuais/Científicos	(92.579 + 219.506), 654.759	(90.266 / 1.220 / 1.093 e 196.961 / 12.148 / 10.397), (589.284 / 32.736 / 32.739)	ROUGE

CAPÍTULO

4

CORPUS, ANOTAÇÃO E CRIAÇÃO DE SUMÁRIOS

4.1 Corpus de opiniões de hotelaria

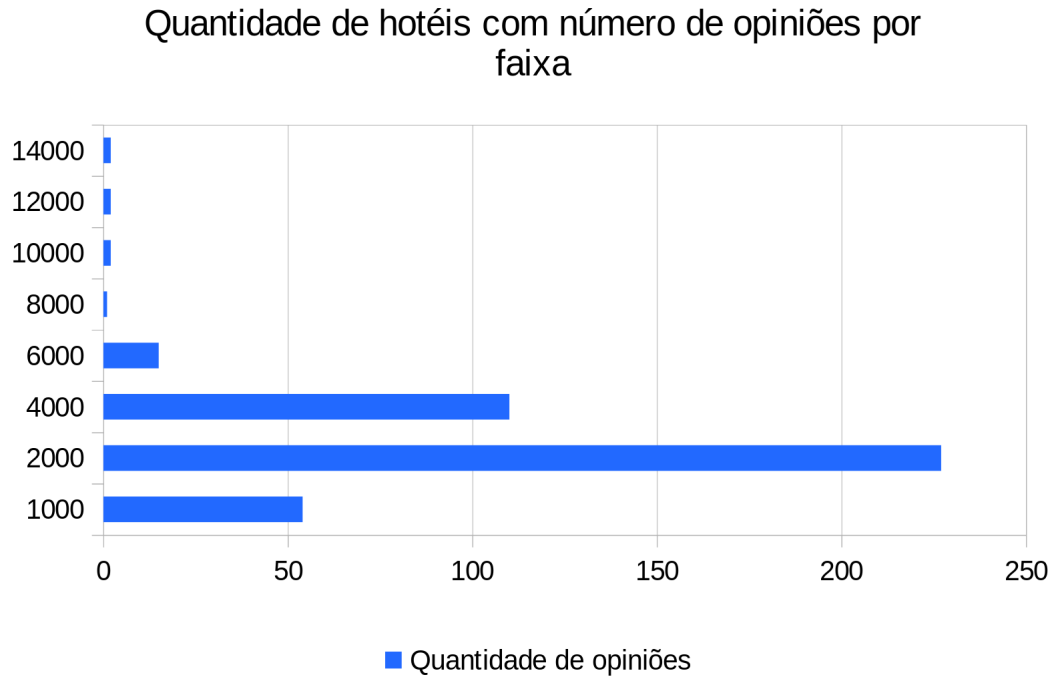
O corpus desenvolvido para este trabalho é constituído de dados coletados da plataforma de viagens e hotelaria *TripAdvisor*, composto por 413 hotéis e 826.436 opiniões no total, resultando em uma média de aproximadamente 2.001 opiniões para cada, mas com alguns hotéis contendo mais que 10.000 opiniões, e outros, menos que 1.000 na realidade, com um histograma da distribuição do número de opiniões sendo dado pela Figura 6. O hotel com a maior quantidade de opiniões possui 13.284 opiniões, e o com a menor quantidade possui 410. O desvio padrão encontrado foi de aproximadamente 1.470,15. Após tokenizar os conteúdos e títulos das opiniões com a biblioteca NLTK[1] antes de aplicar qualquer pré-processamento como remoção de *stopwords* e normalização, foi verificado que o corpus possui 65.715.668 *tokens* nos conteúdos das opiniões, e 2.999.528 *tokens* nos títulos. Ao somar a quantidade de *tokens* obtida nos conteúdos e títulos das opiniões, é possível chegar a um total 68.715.196 *tokens*.

Os hotéis foram selecionados de todas as regiões do Brasil (com a busca restrita apenas ao Brasil), visto que as necessidades e interesses dos clientes podem variar de região para região e uma maior cobertura é sempre bem vinda. As opiniões estão distribuídas em diferentes polaridades, sendo positivas, negativas ou neutras, com essa polaridade sendo baseada na nota de 1 a 5 pontos dada pelo hóspede do hotel. Opiniões contendo menos que 3 estrelas são consideradas negativas, com exatamente 3 estrelas são neutras, e ao conterem mais que 3 estrelas são positivas. Informações adicionais como a data da opinião e o título também estão presentes.

Neste trabalho, cada uma das opiniões presentes é considerada um documento próprio, o que torna esta tarefa uma de sumarização multi-documento. A redundância do conteúdo também

[1] https://www.nltk.org

Figura 6 – Gráfico contendo a distribuição de hotéis por faixa de opiniões



Fonte: Autor

é um fator a se considerar, e o foco avaliativo das opiniões é a nível de entidade, ou seja, as análises geralmente falam sobre o hotel em si e não sobre características específicas do mesmo, embora seja possível extrair características e aspectos das opiniões.

Algumas técnicas tais como a abordagem *A* de (HU; CHEN; CHOU, 2017) consideram informações alheias ao conteúdo da opinião para ranquear as sentenças e opiniões mais importantes. Essas informações podem ser sobre a representatividade do autor, sua credibilidade e a pontuação de recomendação deste no *website* por exemplo, mas como o corpus usado não contempla estes tipos de informação sobre uma opinião e seu autor, então apenas técnicas que utilizam informações como “Data da opinião”, “Conteúdo da opinião”, “Título” e “Nota” são abordadas neste trabalho.

4.2 Criação dos sumários de referência

4.2.1 Metodologia de criação dos sumários gold standard

O processo de criação dos sumários gerais se baseou em partes na metodologia utilizada nas iterações da DUC em 2001, 2002 e 2003. Nelas, cada participante teve de criar um sumário relacionado a cada tópico presente na conferência no modelo de sumarização multi-documento, com a quantidade de palavras e tópicos variando ao longo dos anos.

Já o processo de criação de sumários de referência baseados em aspectos se apoiou

também na metodologia de (CONDORI, 2014) para geração de sumários extrativos, pois este também trabalhou com sumários baseados em aspectos. Esta metodologia foi escolhida porque não só duas das técnicas contempladas são adaptações dele, como seu manual de anotação é um excelente referencial para essa tarefa de sumarização e permite uniformizar melhor a estrutura dos sumários gerados com a de suas adaptações.

Cada hotel foi tratado como um tópico, e como foram selecionados 5 hotéis do corpus, os participantes tiveram de sumarizar 5 "tópicos". Foram definidos 4 anotadores para se encarregarem da tarefa de criação de sumários de referência, com o intuito de realizar testes em todos os sumários gerados.

Por algumas das técnicas utilizadas serem baseadas em aspectos, como a de (TADANO; SHIMADA; ENDO, 2010), certas características estruturais precisaram ser levadas em conta nos sumários resultantes: (i) Os aspectos mais importantes deveriam ser categorizados e ranqueados; (ii) Cada aspecto deveria idealmente conter uma sentença positiva e negativa sobre o mesmo, embora não tenha sido obrigatório incluir uma de cada polaridade pois existiam casos onde não haviam sentenças positivas ou negativas sobre um dado aspecto recorrente nas opiniões sobre um hotel, portanto a informatividade também foi o fator decisivo na seleção de sentenças para os sumários baseados em aspectos; (iii) A categorização a ser seguida deve fazer com que o aspecto mais importante e suas sentenças venham no topo do sumário, o segundo mais importante logo abaixo, e assim por diante.

A criação de sumários abstrativos não foi realizada, visto que apenas técnicas extrativas foram utilizadas, logo, as próximas seções descrevem apenas as técnicas extrativas analisadas e seus detalhes de implementação.

4.2.2 Detalhes do processo de criação

O processo de criação de sumários levou um período de 10 dias do início ao fim, com os principais passos seguidos pelos anotadores para a geração dos sumários *gold standard* sendo dados a seguir:

- As opiniões fornecidas eram mistas, ou seja, havia um apanhado de opiniões positivas, negativas e neutras. Por conta disso, os anotadores não precisaram se atentar a uma única polaridade e puderam selecionar apenas as sentenças mais informativas independente da polaridade destas.

- Cada anotador deveria gerar um sumário de cada tipo por hotel (um geral e um baseado em aspectos). Ou seja, para cada hotel, o anotador deveria gerar um sumário apenas selecionando sentenças mais relevantes, e outro onde as sentenças mais relevantes estão atreladas a algum aspecto.

- No sumário geral, cada entidade deveria possuir sumários com 50 sentenças, não importando a polaridade delas. Essa quantidade de sentenças é a utilizada por padrão em todas as técnicas utilizadas, incluindo as adaptações das implementações de Condori (2014), promovendo uniformização nas técnicas contempladas.

- Já para o sumário baseado em aspectos, deveriam ser encontrados os 5 aspectos mais importantes (podendo considerar os que mais se repetiam entre as opiniões, por exemplo). Cada aspecto deveria conter um total de 10 sentenças relacionadas a ele, podendo ter polaridade positiva, negativa ou neutra, desde que as sentenças em questão sejam julgadas as mais informativas assim como feito no sumário geral. As opiniões selecionadas garantiram que há ao menos 5 aspectos na somatória de todas as opiniões para cada hotel selecionado.

- As sentenças deveriam ser selecionadas de um agrupamento de 50 opiniões sobre cada entidade, retiradas diretamente do corpus. O título de uma opinião também contou como sentença, então caso fosse relevante, ele poderia ser incluído no sumário. Essas opiniões foram escolhidas aleatoriamente após uma filtragem para remover opiniões curtas e pouco relevantes.

- Por se tratar de textos informais, expressões coloquiais e erros de escrita poderiam estar presentes. Caso o anotador percebesse alguma, não deveria realizar quaisquer correções, selecionando a sentença da forma que ela está ao incluí-la no sumário.

- Como o foco do trabalho é sumarização extrativa, os sumários deveriam conter apenas sentenças extraídas diretamente das opiniões, não podendo ser feita qualquer alteração na estrutura gramatical. As sentenças extraídas deveriam ser separadas por uma quebra de linha, mas nunca receber uma reescrita para acomodar sentenças subsequentes.

- Não havia qualquer limite de caracteres ou palavras nos sumários extrativos. Nenhuma restrição do tipo foi imposta pois o tamanho das sentenças é variável, e a informatividade não pode ser sacrificada em prol do tamanho do sumário resultante.

CAPÍTULO

5

METODOLOGIA

Este capítulo aborda a metodologia aplicada no desenvolvimento deste trabalho, contendo detalhes sobre o corpus, a preparação das informações do corpus para utilização posterior, a criação dos sumários de referência, as técnicas utilizadas, e a metodologia de avaliação das mesmas.

5.1 Técnicas utilizadas

Esta seção detalha as técnicas contempladas por este trabalho, explicando os cálculos realizados, a lógica utilizada na seleção de sentenças relevantes, além de outros fatores específicos para cada uma delas.

5.1.1 Técnica 1 - Agrupamento de sentenças pelo método K-Medoids

Esta técnica é baseada na abordagem B vista no trabalho de Hu, Chen e Chou (2017) e utiliza o método *k-medoids* de Kaufmann e Rousseeuw (1987) para realizar a seleção das *k* melhores sentenças a serem colocadas no sumário final. O *k-medoids* trabalha através da obtenção de *k* agrupamentos de sentenças buscando encontrar os agrupamentos mais relevantes ao final da execução.

O processo então segue de forma similar à vista no correlato de Hu, Chen e Chou (2017) no Capítulo 3, onde várias iterações de seleção de sentenças, cálculo da similaridade entre as sentenças e os *medoids* e criação de novos agrupamentos com as sentenças mais similares e próximas dos *medoids* acontecem. Essas iterações ocorrem até que os agrupamentos fiquem estáveis e não seja possível encontrar novos e melhores *medoids*, ou o limite de iterações predefinido seja alcançado[1].

[1] https://scikit-learn-extra.readthedocs.io/en/stable/generated/sklearn_extra.cluster.KMedoids.html

Por ser uma técnica que agrupa objetos e busca aqueles que possuem os maiores níveis de similaridade por meio da minimização da dissimilaridade média deles com base no *medoid*, é necessário utilizar um modelo que possua dois tipos de decisões: A seleção dos objetos como representativos em cada agrupamento terá uma variável Y_i igual a 1 apenas se um objeto i for selecionado como representativo; e a atribuição de um objeto j a um agrupamento onde i é o objeto representativo e também o *medoid* fará com que a variável Z_{ij} seja igual a 1, com este sendo o funcionamento do *k-medoids* original (KAUFMANN; ROUSSEEUW, 1987).

Neste modelo, a restrição $r1$ indica que cada objeto j deve ser atribuído a um único objeto representativo, e junta com a $r4$, para um dado objeto j, haverá alguma variável Z_{ij} que é igual ao valor 1, e todas as outras serão iguais a 0. A restrição $r3$ faz com que um objeto j só seja designado a um objeto i se este segundo for selecionado como um objeto representativo (que representa o agrupamento como um todo), e caso isso não ocorra, a variável Y_i será 0 e por conta disso, a $r4$ garante que todas as Z_{ij} também serão 0. Se i for um objeto representativo, então todas as variáveis Z_{ij} relacionadas a ele poderão assumir valor 0 ou 1. A $r4$ faz com que exatamente k objetos sejam escolhidos como objetos representativos, e conforme os agrupamentos são formados atribuindo os objetos aos mais representativos, haverão k agrupamentos não vazios. A $r2$ implica que a dissimilaridade entre um objeto j e seu objeto representativo é dada por $\sum_{i=1}^{n} d(i,j)Z_{ij}$, e como todos os objetos precisam ser designados, a dissimilaridade total é resultado de $\sum_{j=1}^{n}\sum_{i=1}^{n} d(i,j)Z_{ij}$, que é a função a ser minimizada.

O método utilizado para encontrar os *k-medoids* no grafo gerado com base no documento de entrada é o *Partitioning Around Medoids* (PAM), que particiona os dados das opiniões em k agrupamentos localizados na região dos *medoids* encontrados pelo modelo de agrupamento de Kaufmann e Rousseeuw (1987). O Algoritmo 1 descreve o funcionamento geral do PAM, assim como suas entradas e saídas. Nenhuma alteração foi realizada na execução do PAM, embora soluções mais interessantes já existam, como as melhorias de agrupamento feitas por Schubert e Rousseeuw (2019) por exemplo.

Devido a essa implementação genérica, é possível adaptar o *k-medoids* tanto para textos factuais quanto opinativos, de forma a calcular a similaridade das sentenças tratando estas como os objetos dos agrupamentos. Com isso, basta passar um documento de texto contendo as informações a serem sumarizadas, e outro com as sentenças a serem usadas como "centro" do agrupamento, que serão os *medoids* iniciais contra os quais as sentenças terão suas distâncias e similaridade calculadas.

Algoritmo 1 – Algoritmo de particionamento PAM

Entrada:
D = { t_1,t_2,t_3, ...,t_n } /* Conjunto de sentenças do documento */
A /* Matriz de adjacências que mostra a distância entre as sentenças */
k /* Quantidade de agrupamentos a serem extraídos */
Saída:
K /* Quantidade de agrupamentos obtidos */
Algorítmo do PAM:

Selecione arbitrariamente k medoids de D;

Para cada t_h que não for um medoid **faça**

  Para cada medoid t_i **faça até que:** $TC_{ih} \geq 0$:

    Calcule o erro quadrático médio TC_{ih};

    Encontre i, h onde TC_{ih} assume o menor valor

    se $TC_{ih} < 0$ **então**

      Troque o medoid t_i pelo t_h;

    fim se

  fim para

fim para

Para cada $t_i \in D$ **faça**

  Atribua t_i a K_j onde $dist(t_i, t_j)$ seja a menor sobre todos os medoids;

fim para

Para calcular a distância e dissimilaridade entre as sentenças, foi utilizada a medida de distância euclideana entre as matrizes das opiniões, mas também é possível passar uma matriz de distâncias pré-computadas para o algoritmo. Essas matrizes, por sua vez, foram obtidas através de uma implementação manual do TF-IDF, desenvolvida para lidar com a representação das opiniões como listas de listas. Isso se dá porque já que a representação das opiniões é bi-dimensional, faz sentido manter os vetores TF-IDF no mesmo formato e separar as sentenças por opinião, o que acaba tornando as representações internas em um objeto bi-dimensional que no fim apenas precisa de uma conversão para vetor de forma a ser processado pelo algoritmo de agrupamento.

5.1.2 Técnica 2 - Opizer-E

Desenvolvida por Condori (2014) com base em outras abordagens extrativas, a *Opizer-E* busca gerar sumários extraindo algumas sentenças de opiniões sobre os principais aspectos avaliados, com estes aspectos sendo providos manualmente por meio de anotações. Isso é feito para preservar ao máximo a cobertura de tópicos e aspectos, além da distribuição da polaridade das opiniões.

O funcionamento da técnica ocorre por meio de duas etapas: classificação de sentenças e ranqueamento de sentenças, com o ranqueamento tendo um estágio onde são ranqueados os aspectos e qualificadores por proximidade.

A classificação de sentenças é baseada nos aspectos dos quais elas tratam, além da polaridade delas sobre este mesmo aspecto. Isso faz com que sentenças que tratam do mesmo aspecto, mas possuem polaridades diferentes sejam colocadas em grupos distintos, pois uma das duas características consideradas é diferente em ambas. Após a classificação por aspectos, calcula-se a importância deles utilizando a frequência da ocorrência dos aspectos. Essa importância é calculada de forma similar à vista em trabalhos como o de Hu e Liu (2004), em que quanto mais um aspecto ocorre, mais importante ele é.

Devido a esse enfoque nos aspectos e polaridades, são estas as características que ditam a classificação das sentenças, ou seja, modelos de agrupamento como *k-medoids* ou *k-means* não são utilizados. Por conta disso, a *Opizer-E* utiliza um modelo próprio que separa e classifica as sentenças que abordam um mesmo aspecto com a mesma polaridade em um mesmo grupo, fazendo com que a granularidade seja a nível de aspecto e polaridade. Para que essa classificação funcione bem, é necessário que o corpus possua anotações indicando quais sentenças tratam sobre qual aspecto, e qual a polaridade (também chamada de qualificação) delas nesse contexto, para que o algoritmo possa ler a entrada e descobrir como classificar as sentenças.

A função de distância para ranquear as sentenças mais importantes, assim como na abordagem de Tadano, Shimada e Endo (2010), é realizada por meio da distância euclideana, com o diferencial de que a distância é calculada entre os aspectos e os qualificadores de uma sentença, com ambos sendo explicados a seguir. O cálculo de distância busca encontrar as sentenças onde os pares de aspecto e qualificador possuem a menor distância entre si, pois a noção intuitiva do algoritmo é a de que uma sentença que possui um aspecto e qualificador próximos é mais informativa. Isso faz com que a *Opizer-E* consiga ponderar melhor as sentenças com base no nível de positividade ou negatividade que cerca um dado aspecto, deixando o sumário final mais completo em informações.

Já o ranqueamento de sentenças em si favorece sentenças que tenham aspectos e seus respectivos qualificadores (trechos que denotam algum sentimento sobre o aspecto) próximos uns dos outros, e caso o qualificador de uma determinada sentença acabe estando em outra, o ranqueamento avalia a distância entre o qualificador e o início da sentença. Estas etapas fazem com que o sumário gerado possua sentenças cujas opiniões expressam mais claramente e com mais detalhes os aspectos presentes no texto, facilitando a obtenção do sentimento geral das opiniões sobre um produto ou serviço. A Equação 5.1 apresenta o cálculo do peso para cada sentença.

$$PesoPosicao(sentenca) = \frac{TamanhoOpiniao - Posicao(sentenca)}{TamanhoOpiniao} \tag{5.1}$$

Nessa equação, o $PesoPosicao(sentenca)$ é o peso que varia entre 0 e 1, $TamanhoOpiniao$ é o número de sentenças em uma opinião, e $Posicao(sentenca)$ expressa a posição da sentença na opinião, com a primeira sentença sendo aplicada na posição 0.

Os qualificadores na *Opizer-E* são segmentos textuais ou n-gramas que expressam um sentimento sobre um aspecto, embora essa análise de sentimentos não seja considerada neste trabalho. O ranqueamento de proximidade, por sua vez, é feito através do cálculo da distância entre os aspectos e os qualificadores das sentenças, selecionando aquelas que tiverem esses dois objetos mais próximos, sendo resultado de nada mais que a distância em palavras entre ambos.

A importância final das sentenças em cada aspecto é calculada por uma equação que leva em conta os ranqueamentos de sentença e proximidade explicados anteriormente. Esse cálculo pode ser visto na Equação 5.2.

$$Importancia(sentenca) = \alpha \times PesoPosicao + (1 - \alpha) \times PesoProximidade \tag{5.2}$$

Nesta, $Importancia(sentenca)$ representa a importância da sentença em relação a um determinado aspecto, enquanto que a variável α é utilizada para equilibrar o peso dado pelos dois tipos de ranqueamento. Segundo Condori (2014), o melhor valor de α encontrado foi de 0.65, onde *PesoPosicao* e *PesoProximidade* indicam, respectivamente o peso dado pelos ranqueamentos por posição e proximidade.

A extração das sentenças é separada por aspectos e polaridade, com a sentença mais importante em cada aspecto e polaridade sendo adicionada ao sumário final. Isso resulta em um texto que, para cada aspecto, possui uma sentença positiva e negativa julgadas as mais importantes segundo as equações vistas anteriormente.

5.1.3 Técnica 3 - Abordagem de Tadano, Shimada e Endo (2010)

Essa técnica se baseia na extração de características das opiniões sobre um dado tópico ou aspecto. São levados em conta o teor da opinião, verificando se esta utiliza palavras mais negativas ou positivas; o valor TF-IDF das sentenças, obtido conforme explicado na subseção 2.2.4, dividindo as sentenças em morfemas e avaliando apenas palavras que possuam conteúdo real, isto é, palavras com pronomes e sufixos são ignoradas; e a quantidade de menções a elementos em um mesmo agrupamento relacionado a um certo tópico, pois opiniões mencionadas por outras em um mesmo agrupamento costumam ser mais importantes. A abordagem utilizada aqui é uma adaptação feita por (CONDORI, 2014), que gera um único sumário geral contendo as sentenças positiva e negativa mais importantes para os cinco aspectos mais recorrentes encontrados. O sistema de anotação usado nesta técnica também é o mesmo utilizado pela técnica *Opizer-E*, sendo necessário anotar os aspectos e qualificadores em cada sentença.

O modelo de agrupamento utilizado nessa abordagem é o *K-Means* (STEINHAUS *et al.*, 1956), considerando apenas palavras que eram adjetivos e substantivos nas opiniões. O objetivo dessa técnica é agrupar as sentenças mais semelhantes em relação ao conteúdo e polaridade, com esta segunda sendo descoberta por meio dos adjetivos presentes nas sentenças.

A função de distância utilizada, por sua vez, é a Distância Euclideana, que ajuda a calcular os centroides conforme o *Lloyd's algorithm* (LLOYD, 1982), buscando minimizar a distância entre os agrupamentos como uma medida de maior similaridade. Para tornar os cálculos mais reproduzíveis, o algoritmo foi configurado para evitar agrupamentos vazios.

A implementação do *K-Means* utilizada nessa abordagem começa com k centroides escolhidos arbitrariamente pelo algoritmo, com um vetor sendo alocado para o agrupamento com o centroide cuja menor distância euclideana for encontrada. O algoritmo utiliza refinamentos interativos, recalculando os centroides de cada agrupamento e repetindo este processo até os membros dos agrupamentos se estabilizarem, ou seja, até o algoritmo convergir. [2]

O sumarizador computa a importância de cada agrupamento pela integração do valor TF-IDF de cada sentença e o número de menções a ela. A importância de um agrupamento é dado pela Equação 5.3.

$$Imp(C) = \frac{\sum_{S \in C} TF - IDF_S}{|C|} \times log(|C| + 1) \tag{5.3}$$

Nessa equação, C é um agrupamento de sentenças S, e $|C|$ representa a quantidade total de sentenças no agrupamento. O processo de extração de sentenças é composto de três passos:

1. Identificação da sentença representativa de cada agrupamento, que é aquela com o maior valor TF-IDF em C,

2. Classificação das sentenças representativas de acordo com o esquema de pontuação do corpus (pense no sistema de cinco estrelas do TripAdvisor, por exemplo),

3. Extração das sentenças com um valor alto de $Imp(C)$ com base na distribuição do total de sentenças representativas alocadas em cada pontuação do sistema.

Também é possível utilizar opiniões com qualificações de certos aspectos como um fator na importância delas, para permitir análises a nível de aspecto. Por ser uma adaptação que leva em conta os aspectos das opiniões, essas qualificações de aspectos foram utilizadas na implementação da técnica de Tadano, Shimada e Endo (2010) presente neste trabalho, pois faz com que essa técnica gere resultados organizados de forma similar aos da abordagem *Opizer-E*.

[2] https://tedboy.github.io/nlps/generated/generated/nltk.cluster.KMeansClusterer.html

5.1.4 Técnica 4 - LexRank

Introduzida no trabalho de Erkan e Radev (2004), a *LexRank* é apresentada como uma técnica de sumarização baseada em grafos que utiliza a centralidade lexical de uma sentença como sua saliência no documento de onde foi tirada. Aqui, as sentenças das opiniões representam os vértices do grafo, e a relação de similaridade entre pares de sentenças constituem as arestas. Essa forma de representar sentenças e relações de similaridade é o que permite ao algoritmo utilizar várias heurísticas de centralidade para escolher sentenças.

O algoritmo atua com base no conceito de centralidade de sentenças, isto é, se uma sentença é similar a várias outras em um mesmo grupo, esta é mais central, e portanto mais importante, para o tópico. A *LexRank* aborda esse conceito de duas formas: computando a similaridade entre cada par de sentenças, e computando a centralidade geral de uma dada sentença em relação à sua similaridade às outras. O grau de centralidade é computado através do uso de limiares para remover sentenças que não tenham um coeficiente de similaridade maior que o imposto pelo limiar. O algoritmo padrão não possui pesos nas arestas, mas é possível adicioná-los com base no valor de similaridade adquirido entre cada par de sentenças, constituindo a variação chamada de *Continuous LexRank*.

Por ser uma técnica baseada em grafos, um agrupamento de documentos pode ser descrito como uma rede de sentenças relacionadas entre si, onde existem sentenças bastante similares a outras ao mesmo tempo que algumas compartilham quase nenhuma informação com o restante delas. Para encontrar a similaridade entre duas sentenças, o algoritmo utiliza o modelo *Bag-of-Words* para representar cada sentença como um vetor de n dimensões, com n sendo a quantidade de todas as palavras presentes no vocabulário da língua usada na sumarização. A similaridade entre as duas sentenças é calculada através de uma equação IDF modificada, expressa pela Equação 5.4.

$$\text{IDF-modified-cosine}(x,y) = \frac{\sum_{w \in x,y} TF_{w,x} TF_{w,y} (IDF_w)^2}{\sqrt{\sum_{xi \in x} (TF_{xi,x} IDF_{xi})^2} \times \sqrt{\sum_{yi \in y} (TF_{yi,y} IDF_{yi})^2}} \tag{5.4}$$

Essa equação expressa que a similaridade entre as duas sentenças é o resultado do cosseno entre dois vetores correspondentes, onde $TF_{w,s}$ é a quantidade de ocorrências de uma palavra w em uma sentença s. Um agrupamento de documentos pode ser representado como uma matriz de semelhança por cosseno onde cada valor é a similaridade entre o par de sentenças correspondente, onde uma sentença $ID = dXsY$ indica que se trata da Y-ésima sentença no X-ésimo documento. Também é possível representar a matriz como um grafo ponderado para exibir a similaridade entre um par de sentenças nas arestas.

Por se tratar de um agrupamento de documentos (nesse caso, opiniões) sobre um mesmo tópico, é esperado que várias sentenças sejam similares, o que pode ser um problema dependendo do nível de similaridade entre elas. Para melhorar a correspondência de sentenças no grafo, é

definido um limiar para fazer com que todas as sentenças menos similares que esse limiar sejam ignoradas, mantendo apenas as sentenças que possuírem uma similaridade significativa com as outras. Isso facilita a obtenção dos graus de centralidade das sentenças, que são os graus dos vértices presentes no grafo e obtidos através da contagem das arestas ligadas ao vértice.

Esse grau de centralidade, porém, sofre de alguns problemas, como a possibilidade de ser influenciado negativamente por sentenças indesejadas que estão ligadas entre si, aumentando a centralidade das mesmas. Isso pode acontecer em conjuntos de documentos onde a maioria trata de um mesmo tópico, mas alguns na verdade abordam outros temas porém possuem sentenças importantes sobre o tema tratado pela maioria. Nesses casos, as sentenças relacionadas ao tópico de interesse terão seus graus de centralidade inflados artificialmente por causa dos graus desse conjunto específico de sentenças no documento. Para evitar ocorrências deste tipo, é considerada a localização dos vértices e sua centralidade para pesar cada aresta, além de ser feita uma distribuição do valor de centralidade de um vértice para seus vizinhos.

5.1.5 Técnica 5 - Maximal Marginal Relevance

Apresentado no trabalho de Carbonell e Goldstein (1998), o *Maximal Marginal Relevance* (MMR) é um algoritmo que busca equilibrar a relevância e diversidade das informações retornadas de um conjunto de dados onde este for aplicado. Esta técnica foi pensada inicialmente para ser aplicada em motores de pesquisa (tais como Google, Yahoo, etc.) para retornar links e páginas web, mas também pode ser aplicada a nível de documento e de sentença para a geração de sumários.

O MMR trabalha por meio da seleção de subgrupos de um conjunto de dados aplicando uma abordagem gulosa. O funcionamento desta técnica pode explicado da seguinte forma: Dado um conjunto de dados D, encontre uma quantidade k de subconjuntos $S_k \subset D$ (com $|S_k| = k$ e $k < |D|$) que sejam relevantes a uma dada consulta q através da seleção de subconjuntos S_j^* dado um subconjunto $S_{j-1} = s_1^*, ..., s_{j-1}{}^*$ em que $S_j = S_{j-1} \cup S_j^*$. O critério de seleção é dado pela Equação 5.5.

$$S_j^* = \underset{s_j \in D/S_{j-1}}{\operatorname{argmax}} [\lambda(\mathrm{Sim}_1(s_j, q)) - (1-\lambda) \max_{s_i \in S_{j-1}} \mathrm{Sim}_2(s_j, s_i)] \tag{5.5}$$

Nesta equação, Sim_1 calcula a relevância entre um determinado item (ou neste caso, sentença) e a consulta enquanto que Sim_2 mede a similaridade entre dois itens ou sentenças nos subconjuntos. λ é usado na equação para permitir o *trade-off* entre relevância e similaridade, com os valores que este pode assumir sendo restringidos ao domínio $\lambda \in [0, 1]$. Portanto, o MMR pode ser ajustado para maximizar a diversidade de tópicos através de um λ menor, ou buscar informações semelhantes e reforçar documentos relevantes através de um valor maior para λ.

Por ser uma técnica voltada a ferramentas de busca, o objetivo inicialmente é retornar os documentos mais relevantes possíveis. Entretanto, ao realizar um ajuste na técnica para deixar

de atuar a nível de documento e passar a atuar a nível de sentença, o objetivo da técnica passa a ser a obtenção das sentenças mais relevantes.

Uma das adaptações realizadas na técnica para a sumarização de textos foi a adição de um método para geração automática das consultas q. Nesta versão do MMR as consultas são geradas automaticamente através da seleção das palavras com o maior valor TF-IDF presentes nos dados recebidos como entrada.

Também é possível melhorar o algoritmo através do uso de inferências probabilísticas e de um modelo gráfico de variáveis latentes, gerando o que pode ser chamado de *Probabilistic Latent Maximal Marginal Relevance* (PLMMR) Guo e Sanner (2010). Entretanto, a implementação utilizada aqui é a baseada somente no MMR original, visto que este é mais flexível e pode ser aplicado em uma vasta gama de tarefas sem necessitar de muitos ajustes.

5.2 Detalhes da implementação dos algoritmos

A implementação dos algoritmos foi realizada na linguagem de programação *Python*, já que esta linguagem não só é popular, o que facilita o compreendimento para aqueles que se interessarem na implementação, como também possui pacotes e sintaxe que facilitam a aplicação de ferramentas que se baseiam em aprendizado de máquina e processamento de textos.

A existência de pacotes como *Gensim*, *Scikit-Learn*, NLTK, *SpaCy* e vários outros facilitam bastante a tarefa de preparação das informações e sua posterior utilização nos sumarizadores, além de ter uma sintaxe de fácil compreendimento.

Todos os códigos foram desenvolvidos e ajustados para o *Python* em sua versão 3.7, além de estarem organizados como *Notebooks* do *Google Colaboratory*, que foi o ambiente de execução utilizado em todas elas.

Todas as técnicas implementadas e utilizadas neste trabalho estão disponíveis em um repositório do GitHub[3] para eventuais consultas e replicação de testes. Porém, leve em consideração que todos os testes foram realizados utilizando o Python 3.7 como base, portanto é possível que algumas técnicas demandem ajustes no código para funcionarem em versões mais recentes.

5.2.1 Detalhes da técnica k-medoids

Como entrada, a técnica *k-medoids* espera um vetor de notas das opiniões e uma lista de listas, onde a lista mais interna representa as sentenças de uma opinião, e a lista mais externa representa as opiniões em si. Embora essa lista de listas torne a execução do código mais lenta em relação a uma lista contendo todas as sentenças das opiniões, a representação visual dos documentos se torna muito mais clara, sendo fácil de distinguir os delimitadores de opinião.

[3] https://github.com/AShiningRay/ExtractiveSum-Comparison

Também é possível converter facilmente documentos de texto, ou opiniões delimitadas por caracteres de nova linha, para essa representação interna em listas de listas.

O valor de k foi definido como 50, fazendo com que a técnica resulte em cinquenta agrupamentos e a implementação selecione o *medoid* de cada agrupamento como sentença a ser escolhida, o que naturalmente resulta na inclusão de cinquenta sentenças no sumário final.

A saída, por sua vez, não leva em conta aspectos ou polaridades, apenas imprimindo as n sentenças no documento sem qualquer separação além de uma quebra de linha, com o valor n estando atrelado diretamente ao valor de k definido para os agrupamentos.

5.2.2 Detalhes de implementação da Opizer-E

Funciona de forma similar à técnica acima, pois também foi inicialmente desenvolvida por Condori (2014). A entrada é praticamente idêntica, consistindo das opiniões separadas por arquivo, e um único arquivo de anotações contendo os aspectos avaliados de forma positiva e negativa.

Após verificar com o autor da técnica, foi constatado que não há uma forma de se anotar automaticamente as opiniões para a *Opizer-E*, então estas tiveram de ser feitas manualmente. Para cada uma das 50 opiniões selecionadas de cada hotel, anotações contendo os aspectos e qualificadores tiveram de ser inseridas para que a *Opizer-E* pudesse classificar as sentenças e gerar resultados. Estas anotações foram realizadas pelo autor deste trabalho.

No quesito de quantidade de aspectos avaliados, esta técnica também foi restringida a apenas 5, totalizando um máximo de 50 sentenças extraídas no total devido à inclusão de dez sentenças por aspecto.

A saída, porém, se difere um pouco, já que a *Opizer-E* contabiliza a quantidade de qualificações para cada aspecto (contada com base na quantidade de anotações contendo esses qualificadores), além de escolher o par de sentenças mais importantes para cada um deles assim como na adaptação da técnica de Tadano, Shimada e Endo (2010).

5.2.3 Detalhes da técnica de Tadano, Shimada e Endo (2010)

Como entrada, esta técnica precisa que as opiniões sejam separadas em arquivos de texto distintos, e todas elas estejam anotadas em um outro arquivo específico para anotações. Essa anotação deve conter, para cada opinião, a quantidade e nome dos aspectos, e os trechos onde uma sentença qualifica o aspecto de forma positiva ou negativa. Isso é feito para auxiliar a técnica na extração de aspectos e na ponderação destes.

Por compartilhar o mesmo sistema de anotação para ponderar aspectos e qualificadores utilizado pela *Opizer-E*, os mesmos detalhes sobre o processo de anotação explicados na subseção anterior também se aplicam aqui.

Embora a implementação inicial desta técnica também seja baseada em aspectos, o sumário gerado não indicava essa característica. Mas é possível definir a quantidade de sentenças extraídas com base no número de aspectos considerados. O número de aspectos foi definido como 5, resultando na extração das 10 sentenças mais relevantes para cada aspecto.

Alguns ajustes foram feitos na saída gerada pelo documento, que agora consiste em um documento de texto que ainda contém as sentenças mais importantes encontradas, mas agora separadas pelos aspectos analisados, ao invés de serem imprimidos apenas os *n* pares de sentenças mais importantes sem qualquer separação.

5.2.4 Detalhes de implementação do LexRank

Embora similar ao *K-Medoids* na entrada, aqui as opiniões não podem ser organizadas como uma lista de listas, pois o *LexRank* espera que as sentenças de todas elas estejam numa mesma lista. Por conta disso, a leitura dos documentos ou delimitadores tem de ser realizada de uma forma diferente e que apenas organiza as sentenças sequencialmente com base na ordem das opiniões lidas.

O limiar utilizado para excluir sentenças pouco similares foi de 0.25, assim, apenas sentenças com similaridade calculada entre 0.26 e 1.0 são contabilizadas. Esse limiar foi escolhido pois foi o valor que melhor evitou a seleção de sentenças pouco similares e exclusão de sentenças relevantes.

A saída gerada é similar à do *K-Medoids*, contendo apenas as sentenças mais relevantes e sem qualquer separação por aspecto ou polaridade, e também selecionando as cinquenta sentenças mais relevantes adquiridas pelas heurísticas de centralidade.

5.2.5 Detalhes de implementação do MMR

Apesar de ser uma das menos convencionais por inicialmente tratar de uma técnica voltada a ferramentas de busca, a implementação a nível de sentença desta acaba tornando-a mais próxima das técnicas anteriores.

Para gerar uma consulta, são extraídas as *n* palavras com maior valor TF-IDF presente no conjunto de opiniões recebido da entrada, como explicado anteriormente na descrição do MMR. A partir disso o algoritmo passa a computar os valores TF-IDF das palavras nos agrupamentos de opiniões (pois cada opinião é tratada como um documento) e retira as *k* sentenças com os melhores valores TF-IDF e *Marginal Relevance* para integrar ao sumário final.

O valor dado para λ foi de 0.5, buscando equilibrar a diversidade e cobertura de tópicos com a obtenção das sentenças mais relevantes e que possuem informações semelhantes em relação ao conteúdo de outras sentenças.

5.3 Método de avaliação e validação

Conforme visto anteriormente na subseção 2.3.1, o conjunto ROUGE é composto de quatro métricas diferentes e algumas extensões, todas voltadas a avaliar os sumários automáticos em diferentes quesitos. Para poupar tempo na bateria de testes iniciais, apenas duas das métricas foram utilizadas, estas sendo a ROUGE-N em sua versão para uni-gramas e bigramas, e a ROUGE-L. São utilizadas apenas estas pois elas já cobrem os principais testes de interesse neste trabalho, que são a analise de semelhança entre os documentos por meio da contagem de uni-gramas e bigramas em comum, além da medida da maior sub-cadeia em comum entre eles. Além disso, técnicas como a ROUGE-W e ROUGE-S avaliam os sumários de forma similar à ROUGE-L e ROUGE-2 respectivamente, então o foco foi em utilizar a menor quantidade de métricas que são capazes de gerar a gama mais ampla de resultados e informações sobre os sumários.

Outro fator considerado na utilização destas três métricas é que elas também estiveram presentes em vários outros trabalhos utilizados como referência por este. Trabalhos como os de Tadano, Shimada e Endo (2010), Condori (2014), Bražinskas, Lapata e Titov (2019) e Bhatia (2021) utilizam alguma combinação destas três técnicas, o que ajuda a entender qual a faixa de pontuações que técnicas extrativas costumam ter na ROUGE e verificar se as técnicas abordadas estão funcionando corretamente.

Existem várias implementações distintas da ROUGE, e que trabalham com diferentes iterações da mesma. A implementação utilizada neste trabalho é uma biblioteca independente da implementação de Lin (2004) e portanto pode gerar resultados levemente diferentes. Essa biblioteca provê uma boa integração com python e facilita bastante a obtenção dos resultados de similaridade entre os sumários[4].

[4] https://github.com/pltrdy/rouge

CAPÍTULO

6

RESULTADOS

Como não foi possível selecionar os melhores sumários de referência devido ao número reduzido de anotadores, os testes na ROUGE ocorreram para todos os sumários, resultando em 100 testes no total (5 hotéis $*$ 4 anotadores $*$ 5 técnicas). No âmbito geral, foram utilizadas as técnicas MMR, *LexRank* e *K-Medoids*, já no ramo de técnicas baseadas em aspectos, Opizer-E e Tadano, Shimada e Endo (2010) foram as técnicas empregadas. Os resultados obtidos foram separados por anotador, portanto a Tabela 2 apresenta as pontuações adquiridas após os testes na ROUGE com os sumários do anotador 1, a Tabela 3 para o anotador 2, a Tabela 4 para o anotador 3 e Tabela 5 para o anotador 4. As melhores pontuações obtidas para cada hotel estão marcadas em negrito, para realçar qual foi a técnica que as atingiu.

Nestas tabelas, *Recall* representa a quantidade de informações do sumário de referência presentes no sumário candidato, *Precision* representa a quantidade de informações relevantes presentes no sumário candidato, e *F-measure* é a média harmônica obtida com base nos resultados destas duas medidas anteriores (LIN, 2004). Estas medidas permitem entender com boa profundidade se as técnicas estão retendo informações consideradas importantes com base nos sumários de referência.

Os resultados estão divididos em técnicas que geram sumários gerais, e técnicas baseadas em aspectos. Esta divisão teve de ser feita devido à impossibilidade de se gerar comparações entre os dois tipos de sumários, visto que a metodologia utilizada por um é bastante diferente da metodologia utilizada pelo outro. Portanto, os sumarizadores gerais e baseados em aspectos são analisados separadamente, de forma a obter conclusões para cada tipo e não omitir quaisquer informações apresentadas pelos resultados obtidos.

Os testes foram realizados com as métricas ROUGE-1, ROUGE-2 e ROUGE-L para todos os hotéis e técnicas. Foram selecionadas apenas estas três pois são as mais utilizadas em testes envolvendo o conjunto ROUGE, facilitando a comparação entre diferentes técnicas e metodologias de anotação. Estas três também constituem as principais avaliações realizadas pelo

conjunto, sendo a avaliação por correspondência de uni-gramas e bigramas, e também da maior sub-cadeia em comum entre os sumários automáticos e gerados por humanos.

Para fins de uniformização dos resultados, foi definido que cada técnica extrativa deveria gerar um sumário contendo 50 sentenças no total, de forma a manter o tamanho dos documentos gerados similar entre todas as técnicas e facilitar as comparações com os sumários gerados pelos anotadores humanos. No caso de técnicas extrativas baseadas em aspectos, estas deveriam gerar um sumário contendo os 5 aspectos mais importantes, e as 10 sentenças mais informativas encontradas para cada aspecto. Como uma das técnicas (Opizer-E) extrai sentenças buscando balancear a quantidade de sentenças positivas e negativas, esta trabalhou através da seleção de até 5 sentenças de cada polaridade para cada aspecto relevante, com o objetivo de totalizar em até 10 sentenças por aspecto.

Anotador 1										
		ROUGE-1			ROUGE-2			ROUGE-L		
		Recall	Precision	F-measure	Recall	Precision	F-measure	Recall	Precision	F-measure
Sumários Gerais										
K-Medoids	Hotel 1	**0.2011**	**0.2107**	**0.1902**	**0.0901**	**0.0966**	**0.0910**	**0.1864**	**0.1943**	**0.1780**
	Hotel 2	0.0889	0.0988	0.0848	0.0137	0.0179	0.0149	0.0738	0.0821	0.0695
	Hotel 3	**0.1176**	**0.1200**	**0.1080**	**0.0221**	**0.0235**	**0.0228**	**0.1030**	**0.1047**	**0.0940**
	Hotel 4	**0.0751**	0.0829	**0.0710**	**0.0150**	**0.0158**	**0.0154**	**0.0691**	0.0738	**0.0644**
	Hotel 5	**0.1700**	**0.1711**	**0.1630**	**0.0665**	**0.0694**	**0.0677**	**0.1521**	**0.1535**	**0.1465**
LexRank	Hotel 1	0.0674	0.1184	0.0798	0.0056	0.0092	0.0065	0.0615	0.1114	0.0735
	Hotel 2	**0.1070**	**0.1444**	**0.1142**	**0.0383**	**0.0434**	**0.0388**	**0.0948**	**0.1260**	**0.1011**
	Hotel 3	0.0612	0.0941	0.0683	0.0074	0.0123	0.0088	0.0566	0.0874	0.0632
	Hotel 4	0.0562	0.1067	0.0646	0.0118	0.0218	0.0127	0.0533	**0.1043**	0.0620
	Hotel 5	0.0932	0.1605	0.0997	0.0286	0.0378	0.0309	0.0866	0.1528	0.0933
MMR	Hotel 1	0.0685	0.0762	0.0646	0.0014	0.0006	0.0009	0.0593	0.0680	0.0564
	Hotel 2	0.0558	0.1184	0.0606	0	0	0	0.0503	0.1020	0.0537
	Hotel 3	0.0484	0.0521	0.0413	0.004	0.0011	0.0017	0.0445	0.0497	0.0383
	Hotel 4	0.0665	**0.0884**	0.0676	0.0033	0.0033	0.0031	0.0607	0.0821	0.0621
	Hotel 5	0.0630	0.0775	0.0612	0.0035	0.0036	0.0035	0.0482	0.0581	0.0463
Sumários Baseados em Aspectos										
Opizer-E	Hotel 1	0.0447	0.0594	0.0487	0	0	0	0.0438	0.0581	0.0477
	Hotel 2	0.0560	0.0644	0.0531	**0.0188**	**0.0215**	**0.0195**	0.0521	0.0582	0.0490
	Hotel 3	0.0721	0.0626	0.0609	**0.0115**	**0.0064**	**0.0074**	0.0677	0.0579	0.0563
	Hotel 4	0.0599	0.0775	0.0610	**0.0206**	**0.0224**	**0.0211**	0.0585	0.0769	0.0601
	Hotel 5	0.0436	0.0602	0.0425	0	0	0	0.0403	0.0574	0.0397
Tadano	Hotel 1	**0.1152**	**0.1296**	**0.1145**	**0.0224**	**0.0265**	**0.0220**	**0.1123**	**0.1247**	**0.1110**
	Hotel 2	**0.0865**	**0.0960**	**0.0838**	0.0004	0.0030	0.0007	**0.0846**	**0.0940**	**0.0818**
	Hotel 3	**0.0868**	**0.1121**	**0.0923**	0.0008	0.0008	0.0008	**0.0813**	**0.1060**	**0.0866**
	Hotel 4	**0.0846**	**0.0990**	**0.0857**	0.0181	0.0181	0.0181	**0.0820**	**0.0976**	**0.0839**
	Hotel 5	**0.1109**	**0.1413**	**0.1165**	**0.0222**	**0.0318**	**0.0252**	**0.1070**	**0.1344**	**0.1121**

Tabela 2 – Resultados retornados pela ROUGE para o Anotador 1.

Anotador 2										
		ROUGE-1			ROUGE-2			ROUGE-L		
		Recall	Precision	F-measure	Recall	Precision	F-measure	Recall	Precision	F-measure
Sumários Gerais										
K-Medoids	Hotel 1	**0.0961**	0.0852	0.0811	0.0165	0.0147	0.0142	**0.0783**	0.0740	0.0687
	Hotel 2	0.1034	0.1020	0.0895	0.0084	0.0075	0.0076	0.0891	0.0871	0.0761
	Hotel 3	**0.0820**	**0.0869**	**0.0752**	0.0026	0.0058	0.0031	**0.0701**	0.0734	**0.0638**
	Hotel 4	**0.0893**	0.0807	0.0676	0.0015	0.0021	0.0015	0.0768	0.0659	0.0562
	Hotel 5	**0.1236**	**0.1083**	**0.1071**	**0.0229**	**0.0174**	**0.0193**	**0.1035**	0.0862	**0.0874**
LexRank	Hotel 1	0.0906	**0.1457**	**0.1030**	**0.0171**	**0.0187**	**0.0173**	0.0781	**0.1287**	**0.0889**
	Hotel 2	**0.1116**	**0.1669**	**0.1186**	**0.0189**	**0.0346**	**0.0217**	**0.0997**	**0.1485**	**0.1049**
	Hotel 3	0.0683	0.0799	0.0650	0.0053	0.0027	0.0036	0.0616	0.0721	0.0582
	Hotel 4	0.0830	**0.0855**	**0.0731**	**0.0051**	**0.0035**	**0.0042**	**0.0769**	**0.0767**	**0.0665**
	Hotel 5	0.0940	0.1086	0.0877	0.0128	0.0124	0.0113	0.0840	**0.0995**	0.0790
MMR	Hotel 1	0.0718	0.0813	0.0612	0.0055	0.0035	0.0042	0.0597	0.0697	0.0519
	Hotel 2	0.0554	0.0991	0.0625	0.0040	0.0172	0.0063	0.0448	0.0824	0.0513
	Hotel 3	0.0708	0.0845	0.0652	**0.02**	**0.02**	**0.0199**	0.0637	**0.0745**	0.0594
	Hotel 4	0.0683	0.0762	0.0590	0.0004	0.0005	0.0004	0.0606	0.0701	0.0532
	Hotel 5	0.0654	0.0594	0.0543	0.0050	0.0043	0.0042	0.0582	0.0531	0.0479
Sumários Baseados em Aspectos										
Opizer-E	Hotel 1	0.0763	0.0793	0.0710	**0.0157**	**0.0060**	**0.0086**	0.0706	0.0756	0.0667
	Hotel 2	0.0508	0.0500	0.0443	0.0038	0.0026	0.0031	0.0482	0.0457	0.0416
	Hotel 3	0.0759	0.0602	0.0626	0.0150	0.0068	0.0086	0.0691	0.0546	0.0566
	Hotel 4	0.0824	0.0696	0.0660	**0.0296**	**0.0122**	**0.0154**	0.0798	0.0663	0.0632
	Hotel 5	0.0918	0.0720	0.0652	**0.0125**	0.0013	0.0024	0.0918	0.0720	0.0652
Tadano	Hotel 1	**0.0976**	**0.1055**	**0.0943**	0.0051	0.0012	0.0020	**0.0923**	**0.1022**	**0.0902**
	Hotel 2	**0.0978**	**0.1275**	**0.0986**	**0.0181**	**0.0181**	**0.0181**	**0.0907**	**0.1220**	**0.0931**
	Hotel 3	**0.1165**	**0.1403**	**0.1160**	**0.0246**	**0.0228**	**0.0231**	**0.1140**	**0.1343**	**0.1128**
	Hotel 4	**0.0936**	**0.1134**	**0.0902**	0.0018	0.0022	0.0020	**0.0921**	**0.1127**	**0.0893**
	Hotel 5	**0.1063**	**0.1234**	**0.0995**	0.0029	**0.0042**	**0.0034**	**0.1034**	**0.1176**	**0.0959**

Tabela 3 – Resultados retornados pela ROUGE para o Anotador 2.

Anotador 3										
		ROUGE-1			ROUGE-2			ROUGE-L		
		Recall	Precision	F-measure	Recall	Precision	F-measure	Recall	Precision	F-measure
Sumários Gerais										
K-Medoids	Hotel 1	0.1097	0.1086	0.0942	0.0033	0.0043	0.0034	0.0912	0.0866	0.0767
	Hotel 2	0.1211	0.0916	0.0897	0.0254	0.0213	0.0212	0.1115	0.0804	0.0800
	Hotel 3	**0.0847**	0.0790	0.0681	0.0005	0.0029	0.0009	**0.0700**	0.0697	0.0581
	Hotel 4	**0.0942**	0.0720	**0.0740**	**0.0180**	**0.0161**	**0.0168**	**0.0822**	0.0625	0.0649
	Hotel 5	**0.1251**	0.1062	**0.1076**	**0.0247**	**0.0222**	**0.0225**	**0.1103**	0.0931	**0.0949**
LexRank	Hotel 1	**0.1253**	**0.1601**	**0.1229**	**0.0540**	**0.0518**	**0.0504**	**0.1192**	**0.1561**	**0.1185**
	Hotel 2	**0.1357**	**0.1681**	**0.1345**	**0.0389**	**0.0373**	**0.0372**	**0.1295**	**0.1585**	**0.1276**
	Hotel 3	0.0699	**0.0917**	**0.0692**	**0.0067**	**0.0095**	**0.0074**	0.0647	**0.0864**	**0.0644**
	Hotel 4	0.0729	**0.0811**	0.0680	0.0075	0.0063	0.0064	0.0707	**0.0763**	**0.0650**
	Hotel 5	0.0987	**0.1410**	0.0993	0.0137	0.0180	0.0137	0.0886	**0.1315**	0.0907
MMR	Hotel 1	0.0870	0.0812	0.0755	0.0199	0.0214	0.0205	0.0811	0.0783	0.0719
	Hotel 2	0.0583	0.1165	0.0616	0.0029	0.0126	0.0043	0.0540	0.1057	0.0560
	Hotel 3	0.0707	0.0665	0.0589	0.0033	0.0005	0.0009	0.0586	0.0528	0.0476
	Hotel 4	0.0591	0.0621	0.0533	0.0022	0.001	0.0013	0.0520	0.0546	0.0470
	Hotel 5	0.0568	0.0520	0.0454	0.0064	0.0042	0.0048	0.0477	0.0458	0.0388
Sumários Baseados em Aspectos										
Opizer-E	Hotel 1	**0.1046**	0.1176	**0.1016**	**0.0597**	**0.0518**	**0.0533**	**0.1020**	0.1072	**0.0977**
	Hotel 2	0.0691	0.0876	0.0691	**0.0219**	**0.0221**	**0.0202**	0.0636	0.0832	0.0643
	Hotel 3	**0.0518**	0.0551	0.0459	**0.0049**	0.0024	**0.0031**	**0.0489**	0.0521	0.0431
	Hotel 4	0.0375	0.0387	0.0336	0.0020	0.0012	0.0015	0.0349	0.0377	0.0322
	Hotel 5	0.0433	0.0629	0.0462	0.0026	0.0053	0.0034	0.0401	0.0596	0.0431
Tadano	Hotel 1	0.0963	**0.1213**	0.1007	0.0029	0.0051	0.0036	0.0927	**0.1166**	0.0973
	Hotel 2	**0.0909**	**0.0955**	**0.0826**	0.0036	0.0022	0.0027	**0.0883**	**0.0910**	**0.0795**
	Hotel 3	0.0500	**0.0808**	**0.0560**	0.0006	0.0024	0.0010	0.0476	**0.0738**	**0.0527**
	Hotel 4	**0.0742**	**0.0961**	**0.0797**	**0.0193**	**0.0194**	**0.0193**	**0.0734**	**0.0938**	**0.0785**
	Hotel 5	**0.1048**	**0.1297**	**0.1102**	**0.0375**	**0.0391**	**0.0379**	**0.1022**	**0.1260**	**0.1071**

Tabela 4 – Resultados retornados pela ROUGE para o Anotador 3.

Anotador 4										
		ROUGE-1			ROUGE-2			ROUGE-L		
		Recall	Precision	F-measure	Recall	Precision	F-measure	Recall	Precision	F-measure
Sumários Gerais										
K-Medoids	Hotel 1	**0.1581**	**0.1946**	**0.1646**	**0.0721**	**0.0741**	**0.0727**	**0.1421**	**0.1750**	**0.1485**
	Hotel 2	**0.1686**	**0.2106**	**0.1785**	**0.0869**	**0.0921**	**0.0888**	**0.1537**	**0.1948**	**0.1642**
	Hotel 3	**0.1030**	0.1446	**0.1072**	**0.0189**	**0.0258**	**0.0208**	**0.0880**	0.1245	**0.0926**
	Hotel 4	**0.1279**	**0.1494**	**0.1314**	**0.0564**	**0.0575**	**0.0569**	**0.1152**	**0.1351**	**0.1187**
	Hotel 5	**0.1410**	0.1799	**0.1483**	**0.0608**	**0.0638**	**0.0622**	**0.1281**	0.1652	**0.1358**
LexRank	Hotel 1	0.0643	0.1909	0.0910	0.0035	0.0103	0.0052	0.0573	0.1722	0.0815
	Hotel 2	0.0831	0.1818	0.1049	0.0123	0.0288	0.0153	0.0739	0.1600	0.0927
	Hotel 3	0.0695	**0.1460**	0.0876	0.0079	0.0165	0.0103	0.0623	**0.1322**	0.0785
	Hotel 4	0.0550	0.1127	0.0705	0.0065	0.0095	0.0076	0.0516	0.1042	0.0658
	Hotel 5	0.0892	**0.1925**	0.1076	0.0231	0.0257	0.0235	0.0784	**0.1736**	0.0954
MMR	Hotel 1	0.0620	0.0939	0.0693	0.0013	0.0015	0.0014	0.0536	0.0826	0.0598
	Hotel 2	0.0610	0.1501	0.0752	0.0143	0.0433	0.0183	0.0557	0.1438	0.0698
	Hotel 3	0.0712	0.1102	0.0753	0.0063	0.0123	0.0072	0.0554	0.0907	0.0598
	Hotel 4	0.0774	0.1216	0.0819	0.0211	0.0247	0.0218	0.0673	0.1108	0.0726
	Hotel 5	0.0679	0.1141	0.0738	0.0032	0.0018	0.0023	0.0514	0.0932	0.0568
Sumários Baseados em Aspectos										
Opizer-E	Hotel 1	**0.0755**	**0.1121**	**0.0703**	**0.0240**	**0.0382**	**0.0259**	**0.0726**	**0.1052**	**0.0670**
	Hotel 2	**0.0706**	0.0724	0.0428	0.0021	0.0076	0.0032	**0.0669**	0.0624	0.0377
	Hotel 3	**0.0773**	0.0751	**0.0570**	0.0004	0.0025	0.0008	**0.0701**	0.0672	**0.0497**
	Hotel 4	0.0237	0.0380	0.0274	0	0	0	0.0218	0.0357	0.0253
	Hotel 5	0.0556	0.0623	0.0435	**0.0026**	0.0008	0.0013	0.0556	0.0623	0.0435
Tadano	Hotel 1	0.0487	0.0783	0.0491	0.0019	0.0083	0.0028	0.0480	0.0738	0.0479
	Hotel 2	0.0591	**0.1252**	**0.0701**	**0.0058**	**0.0131**	**0.0076**	0.0506	**0.1136**	**0.0608**
	Hotel 3	0.0472	**0.0903**	0.0529	**0.0046**	**0.0074**	**0.0055**	0.0416	**0.0795**	0.0461
	Hotel 4	**0.0446**	**0.0718**	**0.0501**	**0.0194**	**0.0208**	**0.0198**	**0.0438**	**0.0708**	**0.0492**
	Hotel 5	**0.0650**	**0.1130**	**0.0700**	0.0014	**0.0032**	**0.0019**	**0.0610**	**0.1058**	**0.0650**

Tabela 5 – Resultados retornados pela ROUGE para o Anotador 4.

Com estes resultados em mãos, podemos observar alguns padrões. Veja que independentemente do anotador e da técnica comparada, os valores obtidos são, no geral, muito baixos. Isto se dá devido à natureza subjetiva de se sumarizar opiniões e a subsequente seleção de sentenças bem distinta entre os anotadores, o que não costuma ser um problema em textos factuais e científicos por exemplo. O tamanho do sumário também conta nestes casos, já que sumários de textos científicos e jornalísticos costumam ser bem maiores que os de textos opinativos, e como a ROUGE avalia o número de palavras presentes em cada par de sumários, ela acaba lidando melhor com documentos maiores e com palavras similares.

Uma outra característica interessante é a grande taxa de variação entre os resultados obtidos por uma mesma técnica em diferentes hotéis, sinalizando que a ROUGE não parece lidar bem com textos pequenos, o que faz sentido já que ela foi desenvolvida inicialmente para lidar com textos jornalísticos, em que mesmo os sumários são muito maiores que aqueles gerados com base em opiniões. O conteúdo subjetivo das opiniões também contribui para essa alta taxa de variação entre os resultados.

Note também o quão melhor a técnica de Tadano, Shimada e Endo (2010) se saiu nas avaliações dos sumários baseados em aspectos em relação ao Opizer-E para todos os quatro anotadores, com esta alcançando a maioria dos melhores resultados retornados pela ROUGE nos sumários baseados em aspectos. Isto pode ser explicado pela diferença estrutural entre os sumários gerados por ambas as técnicas. Enquanto a técnica de Tadano, Shimada e Endo (2010)

foca em extrair as n sentenças mais informativas para cada aspecto, o Opizer-E foca em extrair até $n/2$ sentenças positivas, e negativas para cada aspecto, mesmo que algumas das sentenças não sejam as mais informativas. A metodologia de anotação para os sumários baseados em aspectos seguiu a estruturação da técnica de Tadano, Shimada e Endo (2010), influenciando nos resultados da Opizer-E que utiliza uma metodologia diferente para estruturação e seleção de sentenças.

Com os resultados dos anotadores em sumários gerais, podemos perceber que as técnicas *K-Medoids* e *LexRank* são competitivas entre si e se sobressaem em relação ao MMR, o que pode ser visto pela quantidade de melhores resultados obtidos por ambas em todos os testes envolvendo os quatro anotadores. Mesmo que as pontuações obtidas por todos os sumarizadores gerais não sejam altos, ainda é possível ver uma clara vantagem da *K-Medoids* e *LexRank* na seleção de sentenças informativas para um sumário. Os resultados fazem sentido ao considerar que o MMR não foi pensado inicialmente para ser utilizado em sumarização de textos, e sim para classificação de páginas *web* buscando uma maior diversidade de tópicos em relação à consulta do usuário. Por conta disso, é possível ver que a utilização de redundância como indicador de confiabilidade e informatividade das sentenças é algo benéfico para a sumarização automática de opiniões.

Também é possível ver que, a depender do anotador, há casos onde a *K-Medoids* se sai melhor, e outros onde o *LexRank* é o que se sobressai. Como os valores são muito baixos, isso pode ser atribuído à seleção de sentenças das técnicas ter sido mais próxima das sentenças selecionadas por determinados anotadores, portanto não pode-se afirmar a superioridade de quaisquer das duas técnicas no quesito de seleção de sentenças.

Por fim, a ROUGE não aparenta ser uma boa forma de avaliar os sumários de conteúdos opinativos devido aos valores baixos e pouco consistentes entre os testes. De início a conclusão para tal era a de que a estruturação e coesão (ou falta delas) das técnicas extrativas gerava problemas na avaliação da ROUGE, mas de acordo com o trabalho de Tay *et al.* (2019), estes mesmos problemas vistos aqui ocorrem também para sumários abstrativos, indicando que o problema não advém de como estes sumários de opiniões são gerados, e sim da dificuldade do próprio conjunto ROUGE em avaliá-los de forma precisa. Isto pode ser explicado pelo conteúdo informal dos textos opinativos que gera palavras com o mesmo significado, mas escritas de formas diferentes, problemas gramaticais que atrapalham na quebra de sentenças, e a natureza mais curta de textos opinativos que não funciona bem em um conjunto de métricas que avalia sumários através da co-ocorrência de palavras.

Estas dificuldades demostradas pelo conjunto ROUGE podem ser diretamente relacionadas a questões como “Qual a forma ideal de se gerar um sumário de opiniões?”, “Ao criar sumários baseados em aspectos, como se deve estruturar o conteúdo do sumário?”, “Um sumário de opiniões se beneficia mais ao conter informações gerais, ou separadas por aspecto?”, “Qual seria a forma ideal de se avaliar sumários de textos opinativos?”, “É possível desenvolver um método para avaliação automática de sumários de opiniões?”, pois estes são os principais problemas

enfrentados pela sumarização automática de opiniões até então.

CAPÍTULO

7

CONCLUSÃO

Este trabalho apresentou uma comparação entre diferentes tipos de sumarizadores automáticos em um mesmo corpus criado especificamente para esta tarefa de sumarização de opiniões, voltado ao ramo hoteleiro. Informações sobre o corpus, processo de anotação e criação de sumários *gold standard*, especificidades das técnicas e metodologia de testes também foram descritas.

Os testes também foram realizados de forma a verificar como o conjunto de métricas ROUGE se sai ao avaliar sumários de opiniões ao invés de sumários de textos jornalísticos e científicos, visto que não há muitos trabalhos, em especial na língua portuguesa, que realizaram tal estudo. Como a ROUGE é uma das principais métricas usadas na avaliação automática de sumários, é necessário verificar como ela lida com conteúdos opinativos e apontar eventuais problemas em sua metodologia nesse caso.

Os testes realizados apontaram que o conjunto ROUGE aparenta ser dependente da estrutura textual dos sumários comparados e também possui dificuldades em lidar com textos opinativos, que são mais curtos e informais por natureza. Esta característica pôde ser vista nas pontuações obtidas pela técnica de Tadano, que é a que mais se assemelha estruturalmente aos sumários *gold standard*. Também foi possível perceber que em se tratando de sumários gerais, as técnicas *K-Medoids* e *Lexrank* se saem melhor que o MMR na vasta maioria das vezes, já nos sumários baseados em aspectos, a Opizer-E fica consistentemente abaixo da técnica de Tadano na maioria dos resultados devido às observações estruturais levantadas anteriormente.

7.1 Contribuições geradas

Este trabalho gerou uma série de contribuições no ramo de sumarização de opiniões para a língua portuguesa, sendo o primeiro trabalho nesta língua que aborda a sumarização automática de opiniões no ramo de viagens e hotelaria. As principais contribuições geradas por este trabalho

são:

- A análise e implementação de diferentes técnicas para sumarização extrativa de textos voltada a opiniões de hotéis, contendo técnicas que geram sumários gerais, e técnicas que geram sumários baseados em aspectos.
- O desenvolvimento de um novo corpus composto exclusivamente por opiniões do ramo de hotelaria, e considerado o maior até então na língua portuguesa, contendo 825.068 opiniões no total, distribuídas entre 412 hotéis.
- A comparação das diferentes técnicas, bem como sumários gerais e baseados em aspectos no conjunto ROUGE, a fim de analisar como o conjunto se comportava ao avaliar sumários com conteúdos similares, mas estruturados de formas distintas.
- A conclusão de que não é recomendável utilizar o conjunto ROUGE para avaliar sumários de textos de cunho opinativo, e como citado anteriormente, o trabalho de Tay *et al.* (2019) também chegou a conclusões similares, mas com foco em sumários abstrativos. A conclusão que se pode tirar de ambos os trabalhos é a de que a dificuldade da ROUGE com textos opinativos é generalizada, e não fruto das especificidades dos sumários automáticos da modalidade extrativa ou abstrativa.
- Disponibilização de todas as técnicas adaptadas, bem como os resultados, em um repositório no GitHub para fácil execução das técnicas, e a visualização e replicação dos testes realizados no conjunto ROUGE.

7.2 Limitações encontradas

Durante o desenvolvimento deste trabalho, foram detectadas várias limitações que o impediram de alcançar os objetivos estipulados inicialmente. A principal limitação encontrada foi a quantidade de tempo disponibilizada, que impediu a adição de técnicas mais avançadas e também dificultou a participação de mais anotadores, o que acabou por inviabilizar a utilização do coeficiente *Cohen's Kappa*, visto em trabalhos como o de Landis e Koch (1977) e McHugh (2012). Este coeficiente ajuda a selecionar os melhores sumários gerados pelos anotadores e avaliar a concordância dos participantes na seleção, o que certamente influenciaria nos resultados obtidos.

A restrição de tempo também afetou a criação do corpus, que não atingiu seu tamanho máximo esperado devido à existência de opiniões de alguns hotéis que não puderam ser coletadas a tempo; e também o desenvolvimento da metodologia de anotação que não pôde ser testada de forma rígida para encontrar trechos que poderiam sofrer melhorias, beneficiando os sumários resultantes do processo.

Outra limitação encontrada diz respeito à falta de recursos computacionais que, acrescida das limitações de tempo, impediu a anotação do parte do corpus para atender modelos supervisionados, e por conta disso, inviabilizou a implementação e o treinamento de técnicas de sumarização que trabalham com aprendizado de máquina, mesmo as menos exigentes como SVM e *Naive Bayes*.

Já a última limitação encontrada está relacionada à dificuldade de se desenvolver uma metodologia para a avaliação de sumários de opiniões que leve em conta a informatividade e relevância destes, visto que o conjunto ROUGE por si só não é capaz de avaliar o quão informativos e/ou relevantes os sumários gerados durante o trabalho realmente são.

7.3 Trabalhos Futuros

Ao fim deste trabalho também foram encontrados vários cenários que permitem extensões e pesquisas futuras baseando-se no que já foi realizado. Alguns dos pontos de maior destaque são os seguintes:

- A inclusão de técnicas abstrativas na bateria de testes, pois estas não só trabalham com um conceito de sumarização bastante diferente das técnicas extrativas, como também buscam gerar sumários mais coesos.
- Expandir o corpus de opiniões coletadas do *TripAdvisor* através da coleta de opiniões adicionais, uma vez que este tem o potencial de se tornar um corpus grande o suficiente para ser utilizado por técnicas com aprendizado de máquina menos gulosas e que exijam um volume de dados reduzido.
- Anotar uma boa parcela das opiniões coletadas para permitir que o corpus seja utilizado por técnicas de sumarização supervisionadas, que dependem destas anotações na fase de treinamento.
- O desenvolvimento de uma técnica de sumarização extrativa apoiada em BERT para a língua portuguesa, ou uma adaptação do próprio BERT para sumarização automática, pois já existe um modelo pré-treinado para certas tarefas de PLN em português como Reconhecimento de Entidades Nomeadas através do BERTimbau[1] (SOUZA; NOGUEIRA; LOTUFO, 2020).
- Desenvolver uma nova forma de avaliar automaticamente sumários de textos opinativos ou apenas desenvolver um método manual que seja consistente e pouco subjetivo, já que o conjunto ROUGE tem dificuldade em avaliar sumários deste tipo e não pode ser usado.

[1] https://huggingface.co/neuralmind/bert-base-portuguese-cased

REFERÊNCIAS

AKHTAR, N.; ZUBAIR, N.; KUMAR, A.; AHMAD, T. Aspect based sentiment oriented summarization of hotel reviews. **Procedia computer science**, Elsevier, v. 115, p. 563–571, 2017. Citado nas páginas 51 e 57.

BHATIA, S. A comparative study of opinion summarization techniques. **IEEE Transactions on Computational Social Systems**, v. 8, n. 1, p. 110–117, 2021. Citado nas páginas 50, 57 e 74.

BLEI, D. M.; NG, A. Y.; JORDAN, M. I. Latent dirichlet allocation. **Journal of machine Learning research**, v. 3, n. Jan, p. 993–1022, 2003. Citado na página 32.

BLUM, A.; MITCHELL, T. Combining labeled and unlabeled data with co-training. In: **Proceedings of the eleventh annual conference on Computational learning theory**. [S.l.: s.n.], 1998. p. 92–100. Citado na página 54.

BOWMAN, S. R.; VILNIS, L.; VINYALS, O.; DAI, A. M.; JOZEFOWICZ, R.; BENGIO, S. Generating sentences from a continuous space. **arXiv preprint arXiv:1511.06349**, 2015. Citado na página 46.

BRAŽINSKAS, A.; LAPATA, M.; TITOV, I. Unsupervised opinion summarization as copycat-review generation. **arXiv preprint arXiv:1911.02247**, 2019. Citado nas páginas 46, 47, 56 e 74.

BROMLEY, J.; GUYON, I.; LECUN, Y.; SÄCKINGER, E.; SHAH, R. Signature verification using a"siamese"time delay neural network. **Advances in neural information processing systems**, v. 6, 1993. Citado na página 52.

CARBONELL, J.; GOLDSTEIN, J. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In: **Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval**. [S.l.: s.n.], 1998. p. 335–336. Citado na página 70.

CHANDRA, B.; HALLDÓRSSON, M. M. Facility dispersion and remote subgraphs. In: SPRINGER. **Scandinavian Workshop on Algorithm Theory**. [S.l.], 1996. p. 53–65. Citado na página 49.

CHU, E.; LIU, P. Meansum: a neural model for unsupervised multi-document abstractive summarization. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2019. p. 1223–1232. Citado na página 46.

CILIBRASI, R. L.; VITANYI, P. M. The google similarity distance. **IEEE Transactions on knowledge and data engineering**, IEEE, v. 19, n. 3, p. 370–383, 2007. Citado nas páginas 42 e 43.

CONDORI, R. E. L. **Sumarização automática de opiniões baseada em aspectos**. Tese (Doutorado) — Universidade de São Paulo, 2014. Citado nas páginas 17, 25, 26, 28, 30, 45, 56, 61, 62, 65, 67, 72 e 74.

CORTEZ, M. C. A.; MONDO, T. S. Comentários on-line: formação de expectativa e decisão de compra de consumidores hoteleiros. **Rosa dos Ventos**, Universidade de Caxias do Sul, v. 10, n. 1, p. 119–136, 2018. Citado na página 17.

DAGAN, I.; LEE, L.; PEREIRA, F. Similarity-based methods for word sense disambiguation. **arXiv preprint cmp-lg/9708010**, 1997. Citado na página 49.

DANG, H. T. Overview of duc 2005. In: **Proceedings of the document understanding conference**. [S.l.: s.n.], 2005. v. 2005, p. 1–12. Citado nas páginas 38 e 45.

DANG, H. T.; OWCZARZAK, K. Overview of the tac 2008 opinion question answering and summarization tasks. In: **Proc. of the First Text Analysis Conference**. [S.l.: s.n.], 2008. v. 2. Citado na página 49.

DASGUPTA, A.; KUMAR, R.; RAVI, S. Summarization through submodularity and dispersion. 2013. Citado na página 49.

DEVLIN, J.; CHANG, M.-W.; LEE, K.; TOUTANOVA, K. Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2018. Citado nas páginas 52, 53 e 57.

DONG, L.; YANG, N.; WANG, W.; WEI, F.; LIU, X.; WANG, Y.; GAO, J.; ZHOU, M.; HON, H.-W. Unified language model pre-training for natural language understanding and generation. **Advances in Neural Information Processing Systems**, v. 32, 2019. Citado na página 53.

DRUCKER, H.; BURGES, C. J.; KAUFMAN, L.; SMOLA, A.; VAPNIK, V. Support vector regression machines. **Advances in neural information processing systems**, v. 9, 1996. Citado na página 24.

ERKAN, G.; RADEV, D. R. Lexrank: Graph-based lexical centrality as salience in text summarization. **Journal of artificial intelligence research**, v. 22, p. 457–479, 2004. Citado nas páginas 19, 24, 46 e 69.

ESULI, A.; SEBASTIANI, F. Sentiwordnet: A publicly available lexical resource for opinion mining. In: **Proceedings of the Fifth International Conference on Language Resources and Evaluation (LREC'06)**. [S.l.: s.n.], 2006. Citado na página 50.

FERNÁNDEZ, J.; GÓMEZ, J. M.; BARCO, P. M. Evaluation of web information retrieval systems on restricted domains. **Procesamiento del Lenguaje Natural**, v. 45, p. 273–276, 2010. Citado na página 47.

FREITAS, C.; MOTTA, E.; MILIDIÚ, R.; CÉSAR, J. Vampiro que brilha... rá! desafios na anotaçao de opiniao em um corpus de resenhas de livros. **Encontro de Linguística de Corpus**, v. 11, p. 22, 2012. Citado na página 45.

GANESAN, K.; ZHAI, C.; HAN, J. Opinosis: A graph based approach to abstractive summarization of highly redundant opinions. Coling 2010, 2010. Citado nas páginas 45 e 46.

GANESAN, K.; ZHAI, C.; VIEGAS, E. Micropinion generation: an unsupervised approach to generating ultra-concise summaries of opinions. In: **Proceedings of the 21st international conference on World Wide Web**. [S.l.: s.n.], 2012. p. 869–878. Citado na página 48.

GANESAN, K. A. **Opinion driven decision support system**. [S.l.]: University of Illinois at Urbana-Champaign, 2013. Citado nas páginas 25, 51 e 57.

GEHRMANN, S.; DENG, Y.; RUSH, A. Bottom-up abstractive summarization. In: **Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing**. Brussels, Belgium: Association for Computational Linguistics, 2018. p. 4098–4109. Disponível em: <https://aclanthology.org/D18-1443>. Citado na página 53.

GERANI, S.; MEHDAD, Y.; CARENINI, G.; NG, R.; NEJAT, B. Abstractive summarization of product reviews using discourse structure. In: **Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)**. [S.l.: s.n.], 2014. p. 1602–1613. Citado na página 45.

GUO, S.; SANNER, S. Probabilistic latent maximal marginal relevance. In: **Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval**. New York, NY, USA: Association for Computing Machinery, 2010. (SIGIR '10), p. 833–834. ISBN 9781450301534. Disponível em: <https://doi.org/10.1145/1835449.1835639>. Citado na página 71.

HU, M.; LIU, B. Mining and summarizing customer reviews. In: **Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining**. [S.l.: s.n.], 2004. p. 168–177. Citado nas páginas 45, 46 e 66.

HU, Y.-H.; CHEN, Y.-L.; CHOU, H.-L. Opinion mining from online hotel reviews–a text summarization approach. **Information Processing & Management**, Elsevier, v. 53, n. 2, p. 436–449, 2017. Citado nas páginas 41, 42, 43, 44, 56, 60 e 63.

IM, J.; KIM, M.; LEE, H.; CHO, H.; CHUNG, S. Self-supervised multimodal opinion summarization. **arXiv preprint arXiv:2105.13135**, 2021. Citado na página 33.

JOLLIFFE, I. Principal component analysis. In: ____. **International Encyclopedia of Statistical Science**. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. p. 1094–1096. ISBN 978-3-642-04898-2. Disponível em: <https://doi.org/10.1007/978-3-642-04898-2_455>. Citado na página 51.

JONES, K. S. A statistical interpretation of term specificity and its application in retrieval. **Journal of documentation**, MCB UP Ltd, 1972. Citado na página 31.

JONES, K. S. Automatic summarizing: factors immarizing: factors and directions. **Advances in automatic text summarization**, MIT press, p. 1, 1999. Citado na página 47.

JUNG, J. J.; JO, G.-S. Template-based e-mail summarization for wireless devices. In: SPRINGER. **International Symposium on Computer and Information Sciences**. [S.l.], 2003. p. 99–106. Citado na página 25.

JUNIOR, J. H. F. Sumopinions: sumarização automática de opiniões sobre pontos turísticos. 2018. Citado nas páginas 19, 28, 29, 30, 33, 43, 44 e 56.

KAUFMANN, L.; ROUSSEEUW, P. Clustering by means of medoids. **Data Analysis based on the L1-Norm and Related Methods**, p. 405–416, 01 1987. Citado nas páginas 33, 42, 43, 63 e 64.

KIM, H. D.; PARK, D. H.; VYDISWARAN, V. V.; ZHAI, C. Opinion summarization using entity features and probabilistic sentence coherence optimization: Uiuc at tac 2008 opinion summarization pilot. In: CITESEER. **TAC**. [S.l.], 2008. Citado na página 49.

LANDIS, J. R.; KOCH, G. G. An application of hierarchical kappa-type statistics in the assessment of majority agreement among multiple observers. **Biometrics**, JSTOR, p. 363–374, 1977. Citado na página 82.

LEWIS, M.; LIU, Y.; GOYAL, N.; GHAZVININEJAD, M.; MOHAMED, A.; LEVY, O.; STOYANOV, V.; ZETTLEMOYER, L. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. **arXiv preprint arXiv:1910.13461**, 2019. Citado na página 33.

LI, B.; LIU, Y.; AGICHTEIN, E. Cocqa: co-training over questions and answers with an application to predicting question subjectivity orientation. In: **Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing**. [S.l.: s.n.], 2008. p. 937–946. Citado na página 49.

LI, L.; ZHOU, K.; XUE, G.-R.; ZHA, H.; YU, Y. Enhancing diversity, coverage and balance for summarization through structure learning. In: **Proceedings of the 18th international conference on World wide web**. [S.l.: s.n.], 2009. p. 71–80. Citado na página 19.

LIN, C.-Y. Rouge: A package for automatic evaluation of summaries. In: **Text summarization branches out**. [S.l.: s.n.], 2004. p. 74–81. Citado nas páginas 34, 35, 38, 45, 74 e 75.

LIN, H.; BILMES, J. A class of submodular functions for document summarization. In: **Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies**. [S.l.: s.n.], 2011. p. 510–520. Citado na página 49.

LIU, B. Sentiment analysis and opinion mining. **Synthesis lectures on human language technologies**, Morgan & Claypool Publishers, v. 5, n. 1, p. 1–167, 2012. Citado nas páginas 18, 27 e 28.

LIU, Y.; LAPATA, M. Text summarization with pretrained encoders. **arXiv preprint arXiv:1908.08345**, 2019. Citado nas páginas 53 e 57.

LLORET, E.; BALAHUR, A.; GÓMEZ, J. M.; MONTOYO, A.; PALOMAR, M. Towards a unified framework for opinion retrieval, mining and summarization. **Journal of Intelligent Information Systems**, Springer, v. 39, n. 3, p. 711–747, 2012. Citado na página 50.

LLORET, E.; BOLDRINI, E.; VODOLAZOVA, T.; MARTÍNEZ-BARCO, P.; MUÑOZ, R.; PALOMAR, M. A novel concept-level approach for ultra-concise opinion summarization. **Expert Systems with Applications**, Elsevier, v. 42, n. 20, p. 7148–7156, 2015. Citado nas páginas 47, 48 e 56.

LLOYD, S. Least squares quantization in pcm. **IEEE Transactions on Information Theory**, v. 28, n. 2, p. 129–137, 1982. Citado na página 68.

LOUVIERE, J. J.; FLYNN, T. N.; MARLEY, A. A. J. **Best-Worst Scaling: Theory, Methods and Applications**. [S.l.]: Cambridge University Press, 2015. Citado na página 46.

LUHN, H. P. A statistical approach to mechanized encoding and searching of literary information. **IBM Journal of research and development**, IBM, v. 1, n. 4, p. 309–317, 1957. Citado na página 31.

MANI, I. **Automatic summarization**. [S.l.]: John Benjamins Publishing, 2001. v. 3. Citado nas páginas 17 e 26.

MCCARGAR, V. Statistical approaches to automatic text summarization. **Bulletin of the American Society for Information Science and Technology**, v. 30, n. 4, p. 21–25, 2004. Citado nas páginas 19 e 24.

MCHUGH, M. L. Interrater reliability: the kappa statistic. **Biochemia medica**, Medicinska naklada, v. 22, n. 3, p. 276–282, 2012. Citado na página 82.

NARAYAN, S.; COHEN, S. B.; LAPATA, M. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In: **Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing**. Brussels, Belgium: Association for Computational Linguistics, 2018. p. 1797–1807. Disponível em: <https://aclanthology.org/D18-1206>. Citado nas páginas 52 e 53.

NENKOVA, A.; MCKEOWN, K. **Automatic summarization**. [S.l.]: Now Publishers Inc, 2011. Citado nas páginas 17, 23 e 25.

PANG, B.; LEE, L. *et al.* Opinion mining and sentiment analysis. **Foundations and Trends® in information retrieval**, Now Publishers, Inc., v. 2, n. 1–2, p. 1–135, 2008. Citado na página 26.

PATEL, D.; SHAH, S.; CHHINKANIWALA, H. Fuzzy logic based multi document summarization with improved sentence scoring and redundancy removal technique. **Expert Systems with Applications**, Elsevier, v. 134, p. 167–177, 2019. Citado na página 18.

POLAJNAR, T.; RIMELL, L.; CLARK, S. An exploration of discourse-based sentence spaces for compositional distributional semantics. In: **Proceedings of the First Workshop on Linking Computational Models of Lexical, Sentential and Discourse-level Semantics**. Lisbon, Portugal: Association for Computational Linguistics, 2015. p. 1–11. Disponível em: <https://aclanthology.org/W15-2701>. Citado na página 55.

PORTER, M. F. An algorithm for suffix stripping. **Program**, MCB UP Ltd, 1980. Citado nas páginas 29 e 41.

RADEV, D. R. Generating natural language summaries from multiple on-line sources. 1997. Citado na página 26.

RAMOS, J. *et al.* Using tf-idf to determine word relevance in document queries. In: CITESEER. **Proceedings of the first instructional conference on machine learning**. [S.l.], 2003. v. 242, n. 1, p. 29–48. Citado na página 32.

RATNAPARKHI, A. **Maximum entropy models for natural language ambiguity resolution**. [S.l.]: University of Pennsylvania, 1998. Citado na página 29.

RAUT, V. B.; LONDHE, D. Opinion mining and summarization of hotel reviews. In: IEEE. **2014 International Conference on Computational Intelligence and Communication Networks**. [S.l.], 2014. p. 556–559. Citado nas páginas 50 e 56.

RIBALDO, R.; AKABANE, A. T.; PARDO, T. A. S. Multi-document summarization with graph metrics. 2012. Citado na página 45.

RIBALDO, R.; PARDO, T. A.; RINO, L. H. Sumarização automática multidocumento com mapas de relacionamento. In: **STIL STUDENT WORKSHOP ON INFORMATION AND HUMAN LANGUAGE TECHNOLOGY**. [S.l.: s.n.], 2011. v. 2, p. 1–3. Citado nas páginas 19 e 24.

SCHILDER, F.; KONDADADI, R. Fastsum: Fast and accurate query-based multi-document summarization. In: **Proceedings of ACL-08: HLT, short papers**. [S.l.: s.n.], 2008. p. 205–208. Citado nas páginas 19 e 24.

SCHUBERT, E.; ROUSSEEUW, P. J. Faster k-medoids clustering: Improving the pam, clara, and clarans algorithms. In: AMATO, G.; GENNARO, C.; ORIA, V.; RADOVANOVIĆ, M. (Ed.). **Similarity Search and Applications**. Cham: Springer International Publishing, 2019. p. 171–187. ISBN 978-3-030-32047-8. Citado na página 64.

SEILLES, A. **Structuration de débats en ligne à l'aide d'Annotationssocio-sémantiquesVers une analyse de réseaux sociaux centrés sur l'interaction**. Tese (Doutorado) — Montpellier 2, 2012. Citado na página 44.

SILVA, M. M.; FILHO, L. M. Intenção de uso de comentários de viagem online na escolha de um meio de hospedagem: Fatores influenciadores. **Revista Brasileira de Pesquisa em Turismo**, Associação Nacional de Pesquisa e Pós-Graduação em Turismo, v. 8, n. 3, p. 419–434, 2014. Citado na página 17.

SIMONASSI, R. *et al.* Uma abordagem de sumarização automática de textos aplicada a debates online. Universidade Católica de Brasília, 2016. Citado nas páginas 44, 45 e 56.

SOUZA, F.; NOGUEIRA, R.; LOTUFO, R. Bertimbau: pretrained bert models for brazilian portuguese. In: SPRINGER. **Intelligent Systems: 9th Brazilian Conference, BRACIS 2020, Rio Grande, Brazil, October 20–23, 2020, Proceedings, Part I 9**. [S.l.], 2020. p. 403–417. Citado na página 83.

STEINHAUS, H. *et al.* Sur la division des corps matériels en parties. **Bull. Acad. Polon. Sci**, v. 1, n. 804, p. 801, 1956. Citado na página 68.

TADANO, R.; SHIMADA, K.; ENDO, T. Multi-aspects review summarization based on identification of important opinions and their similarity. In: **Proceedings of the 24th Pacific Asia conference on language, information and computation**. [S.l.: s.n.], 2010. p. 685–692. Citado nas páginas 14, 45, 61, 66, 67, 68, 72, 74, 75, 78 e 79.

TAY, W.; JOSHI, A.; ZHANG, X. J.; KARIMI, S.; WAN, S. Red-faced rouge: Examining the suitability of rouge for opinion summary evaluation. In: **Proceedings of the The 17th Annual Workshop of the Australasian Language Technology Association**. [S.l.: s.n.], 2019. p. 52–60. Citado nas páginas 79 e 82.

VASWANI, A.; SHAZEER, N.; PARMAR, N.; USZKOREIT, J.; JONES, L.; GOMEZ, A. N.; KAISER, Ł.; POLOSUKHIN, I. Attention is all you need. **Advances in neural information processing systems**, v. 30, 2017. Citado nas páginas 33 e 53.

WANG, L.; RAGHAVAN, H.; CARDIE, C.; CASTELLI, V. Query-focused opinion summarization for user-generated content. **arXiv preprint arXiv:1606.05702**, 2016. Citado nas páginas 48, 49, 50 e 56.

WONG, K.-F.; WU, M.; LI, W. Extractive summarization using supervised and semi-supervised learning. In: **Proceedings of the 22nd international conference on computational linguistics (Coling 2008)**. [S.l.: s.n.], 2008. p. 985–992. Citado nas páginas 54 e 57.

ZHENG, H.; LAPATA, M. Sentence centrality revisited for unsupervised summarization. **arXiv preprint arXiv:1906.03508**, 2019. Citado nas páginas 55 e 57.

ZHONG, M.; LIU, P.; CHEN, Y.; WANG, D.; QIU, X.; HUANG, X. Extractive summarization as text matching. **arXiv preprint arXiv:2004.08795**, 2020. Citado nas páginas 19, 24, 52, 53 e 57.