UNIVERSIDADE DE SÃO PAULO
Instituto de Ciências Matemáticas e de Computação

Sumarização Automática de Atualização para a língua portuguesa

Fernando Antônio Asevedo Nóbrega

Tese de Doutorado do Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional (PPG-CCMC)

ICMC USP SÃO CARLOS

Fernando Antônio Asevedo Nóbrega

Sumarização Automática de Atualização para a língua portuguesa

Tese apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Doutor em Ciências – Ciências de Computação e Matemática Computacional. *EXEMPLAR DE DEFESA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientador: Prof. Dr. Thiago Alexandre Salgueiro Pardo

USP – São Carlos
Outubro de 2017

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados fornecidos pelo(a) autor(a)

N754s

Nóbrega, Fernando Antônio Asevedo
Sumarização Automática de Atualização para a língua portuguesa / Fernando Antônio Asevedo Nóbrega; orientador Thiago Alexandre Salgueiro Pardo. -- São Carlos, 2017.
173 p.

Tese (Doutorado - Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2017.

1. Sumarização Automática de Atualização. 2. Compressão de Sentenças. 3. Sumarização Compressiva. I. Pardo, Thiago Alexandre Salgueiro, orient. II. Título.

Fernando Antônio Asevedo Nóbrega

Update Summarization for the portuguese language

Doctoral dissertation submitted to the Institute of Mathematics and Computer Sciences – ICMC-USP, in partial fulfillment of the requirements for the degree of the Doctorate Program in Computer Science and Computational Mathematics. *EXAMINATION BOARD PRESENTATION COPY*

Concentration Area: Computer Science and Computational Mathematics

Advisor: Prof. Dr. Thiago Alexandre Salgueiro Pardo

USP – São Carlos
October 2017

Aos meus pais, Francisco e Maria Salete, que sempre lutaram muito para proporcionar oportunidades de estudos aos seus filhos.

Ao meu irmão, Francisco Jr., que sempre me motivou a seguir esse caminho.

À minha esposa, Fernanda, que me acompanhou e me deu forças durante toda minha vida acadêmica.

AGRADECIMENTOS

Primeiramente, agradeço a todo a minha família, que apesar da distância, sempre se mostrou presente. Em especial, aos meus pais, Francisco e Maria Salete, que sempre se preocuparam com a educação de seus filhos. Aos meus irmãos, sobretudo ao meu irmão mais novo, Francisco Jr., que dividiu as idas e vindas dos colégios e sempre compartilhou aprendizados e me deu conselhos. Aos meus padrinhos, Jonas e Creuza, e meu primo, Jamilson, que sempre torceram pelas minhas conquistas.

Um agradecimento muito especial à minha esposa, Fernanda, que esteve ao meu lado desde o início de minha trajetória acadêmica e sempre foi paciente e me ajudou muito durante esse caminho. Aos meus sobrinhos, David e Matheus (também afilhado) que me fazem lembrar que momentos simples também são importantes.

Aos integrantes da Rep Catalão, Edvard, Faimison (Pneu) e Rayner, que foram muito importantes para meu crescimento pessoal e acadêmico. Aos agregados, Adam, Fausto e ao casal Flávia e Miky, que sempre estiveram presentes em diversos momentos de descontração e aprendizado. Também ao Leandro, que me ajudou durante o estágio no exterior.

Deixo meu muito obrigado e um grande abraço os amigos do NILC, que proporcionaram inúmeros momentos de estudo, diversão nos happy hours, piadas excelentes... e foram muito além de um ambiente de trabalho e acadêmico.

A todos meus Professores que, de alguma forma, me impulsionaram até aqui. Em especial, ao Prof. Thiago, que foi meu orientador desde o mestrado e que sempre será uma referência para mim. Ao Prof. Alípio e ao Prof. Pavel, que me orientaram durante o estágio na Universidade do Porto. À Profa. Élen, pelas aulas de inglês.

Agradeço também a CAPES, pela bolsa de doutorado, CNPq e Fapesp, que indiretamente financiaram o andamento deste projeto. Ao ICMC-USP e NILC pela infraestrutura (técnica e humana).

Agradeço também a Deus, pois, além de ser fonte de paz e força, me proporcionou conhecer as pessoais que agradeci e que contribuíram de alguma forma, com este projeto.

"O impossível não é um fato: é uma opinião."
(Mario Sergio Cortella)

RESUMO

NÓBREGA, F. A. A. **Sumarização Automática de Atualização para a língua portuguesa**. 2017. 173 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2017.

O enorme volume de dados textuais disponível na web se caracteriza como um cenário ideal para inúmeras aplicações do Processamento de Língua Natural. Nesse ambiente, diferentes informações sobre algum assunto são publicadas por inúmeras fontes e novo conteúdo relacionado é rapidamente divulgado em meios digitais. Nesse contexto, tem-se a tarefa da Sumarização Automática de Atualização (SAA), que tem por objetivo a geração automática de resumos a partir de uma coleção textual relacionados admitindo-se que o leitor possui algum conhecimento prévio sobre os textos-fonte. Por exemplo, tendo em vista os textos já lidos por um usuário, pode-se produzir um sumário a partir de novos conteúdos visando lhe apresentar somente as informações mais relevantes, recentes, novas e atualizadas. O processo de geração de resumos implica em diversos desafios, sobretudo na seleção e síntese de conteúdo para o sumário. No contexto da seleção de conteúdo, têm-se inúmeras abordagens na literatura, com diferentes níveis de complexidade teórica e computacional. Entretanto, pouco dessas investigações fazem uso de algum conhecimento linguístico profundo, que pode auxiliar a identificação de conteúdo mais relevante. No âmbito da síntese de conteúdo, a abordagem mais empregada é a extrativa, na qual algumas sentenças dos textos-fonte são selecionadas e, sem alteração, organizadas no sumário. Tal abordagem, embora apresente resultados satisfatórios, pode limitar a informatividade do sumário, uma vez que alguns segmentos sentenciais podem conter informação redundante ou irrelevante. Dessa forma, esforços recentes foram direcionados à síntese compressiva, na qual as sentenças selecionadas para o sumário são eventualmente comprimidas previamente à inserção no sumário. Anteriormente a este trabalho, a maioria das investigações de Sumarização Automática para a língua Portuguesa foi direcionada à geração de sumários a partir de um (monodocumento) ou vários textos relacionados (multidocumento) por meio da síntese extrativa. Dado esse cenário, este trabalho de doutorado objetivou investigar a SAA por meio da síntese extrativa e compressiva com ênfase na língua Portuguesa. Dada as inúmeras abordagens para a etapa de seleção de conteúdo na SAA, a tese deste trabalho foi que enriquecer-las com conhecimentos linguísticos auxilia a produção de sumários mais informativos. Assim, investigaram-se maneiras para adicionar conhecimentos linguísticos nos métodos mais representativos de cada abordagem de SAA. Por meio dos resultados, observou-se que somente algumas abordagens foram auxiliadas pelo uso desse conhecimento. Além disso, foram propostas algumas simplificações para um modelo de distribuição de tópicos por meio da teoria linguística de Subtópicos. Tais métodos reque-

rem menor complexidade computacional e apresentaram bons desempenhos. Ressalta-se que no escopo da SAA, foram realizados experimentos para a língua Portuguesa e Inglesa, que possui recursos amplamente difundidos. Posteriormente, desenvolveram-se inúmeros métodos de Compressão Sentencial por meio de algoritmos de Aprendizado de Máquina para incorporar uma arquitetura de síntese compressiva para a SAA. O melhor método apresentou resultados superiores a um trabalho do estado da arte, que faz uso de algoritmos de *Deep Learning*. Tendo a língua Portuguesa como principal objeto de estudo, foram organizados três córpus, o CSTNews-Update, que viabiliza experimentos de SAA, e o PCSC-Pares e G1-Pares, para o desenvolvimento/avaliação de métodos de Compressão Sentencial.

Palavras-chave: Sumarização Automática de Atualização, Compressão Sentencial, Sumarização Compressiva.

ABSTRACT

NÓBREGA, F. A. A. **Update Summarization for the portuguese language**. 2017. 173 p. Tese (Doutorado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos – SP, 2017.

The huge amount of textual data in the web may be handled as the main scenery for many application of Natural Language Procession. Here, there are many sources that share distinct information about a given subject and, moreover, usually new related content is quickly produced in digital media. This scenery may be handled for the Update Summarization (US) task, which aims to produce a summary from a text collection under the assumption the user has some previous knowledge. For instance, if the user has already read some texts previously, we may produce a summary from new related documents in order to show him the most salient, recent and updated content. The production of summaries automatically requires many research challenges, mainly during the content selection and synthesis of the summary stages. For the first one, it has been proposed many and distinct approaches, which requires different theories and computational complexity. However, most of them do not use some deep linguistic knowledge, which may assist the identification of the most salient and updated information in the source-texts. Aiming the content synthesis of summaries, the most of the systems use an extractive approach, in which some sentences from the source-texts are picked and organized in the output without rewriting operations. Even with satisfactory results, the extractive synthesis may to reduce the informativeness of the summaries, once some irrelevant or redundant information may occur in some segments of the selected sentences. Thus, recent investigations have focused on the compressive approach, in which the systems may use short versions of some picked sentences in the output summaries. Before this work, most of the investigations related to Automatic Summarization for the Portuguese language were focused on single and multi-document summarization by use the extractive approach. Given this background, we have investigated US methods by use the extractive and compressive approaches for Portuguese. Given the amount of approaches for content selection, our main hypothesis is based on the idea to enrich them with some linguistic theories can to produce more informative summaries. Thus, we have proposed distinct ways to use this kind of information for the most relevant US methods of different approaches. The results show the used linguistic knowledge assists just some US approaches. Furthermore, we also have proposed some topic model simplifications based on the Subtopic theory. These methods require less computational power and have shown good results. It is important to say we have performed experiments for US over datasets on Portuguese and English, in which there are more resources. After that, we have investigated Sentence Compression

methods by use Machine Learning in order to use them into a compressive architecture for US. The proposed method with the highest evaluation scores beats a state of art system, which is based on Deep Learning techniques. Once we have focused on Portuguese, we also have proposed three corpora for this language, the CSTNews-Update, which enables the investigation of US, and the PCSC-Pairs and G1-Pairs datasets, which are used to produce and evaluate the sentence compression methods.

Keywords: Update Summarization, Sentence Compression, Compressive Summarization.

LISTA DE ILUSTRAÇÕES

LISTA DE QUADROS

LISTA DE ALGORITMOS

LISTA DE TABELAS

SUMÁRIO

CAPÍTULO

1

INTRODUÇÃO

1.1 Contexto e motivação

A quantidade e a velocidade com que informações são publicadas, sobretudo na web e por mídia textual, apresentam-se como a principal motivação para diversas áreas de pesquisa em Processamento da Língua Natura (PLN). Gantz e Reinsel (2012), por exemplo, em artigo publicado pelo IDC[1], estimaram que 10 mil *exabytes* de dados *online* seriam publicados em 2014, e que 40 mil *exabytes* estarão disponíveis em 2020. Nesse contexto, aliado ao tempo cada vez mais escasso devido às atividades cotidianas para as pessoas assimilarem os dados disponíveis, uma área do PLN que se torna muito relevante é a Sumarização Automática (SA) de textos, que consiste na produção automática de uma versão condensada de um ou mais texto-fonte, seu sumário ou, como mais conhecido, seu resumo (MANI, 2001). Por exemplo, um usuário/leitor, por meio de uma ferramenta de busca de notícias *online*, encontraria aproximadamente 1.590.000 páginas[2] relacionadas à "Mundial de F1 2017". A leitura e compreensão desse volume de dados em tempo hábil é praticamente inviável. Assim, métodos de SA poderiam ser empregados para reduzir o tempo de leitura por meio da construção de um resumo das informações mais relevantes ao leitor.

O primeiro foco de investigações em SA foi direcionado à produção automática de sumários a partir de um único texto-fonte, referenciada como Sumarização Automática Monodocumento, tais como os trabalhos de Edmundson (1969), Marcu (1997), Barzilay e Elhadad (1997), Carbonell e Goldstein (1998), Rino *et al.* (2004), Mihalcea e Ceylan (2007). Posteriormente, esse objetivo foi ampliado para a Sumarização Automática Multidocumento, cujo objetivo é a construção de sumários que relatem as informações contidas

[1] http://www.emc.com/leadership/digital-universe/index.htm
[2] Busca realizada por meio do Google News®(sem configurações específicas de usuário) em 13 de julho de 2017.

em uma coleção estática de textos relacionados (MCKEOWN; RADEV, 1995; RADEV; MCKEOWN, 1998), como pode ser observado nos trabalhos de Barzilay, McKeown e Elhadad (1999), Daumé III e Marcu (2006), Boudin, El-Bèze e Moreno (2008), Antiqueira *et al.* (2009), Bhaskar e Bandyopadhyay (2010), Camargo *et al.* (2013), Cardoso e Pardo (2016). Por exemplo, produzir um único sumário a partir das 10 primeiras páginas/notícias relacionados ao "Mundial de F1 2017" que foram retornadas por alguma ferramenta de busca. Dessa forma, por meio de um bom sumário, um usuário humano que deseja ler as principais informações sobre esse tema despenderia muito menos tempo de leitura.

No Quadro 1, são apresentados três sumários monodocumento que foram criados a partir de três textos jornalísticos distintos que estão presentes no córpus CSTNews (ALEIXO; PARDO, 2008; CARDOSO *et al.*, 2011). Cada sumário possui aproximadamente 30% do número de palavras dos respectivos textos-fonte, que são relacionados à Liga Mundial de Vôlei de 2006. É importante ressaltar que os sumários são constituídos pelas informações mais relevantes dos respectivos textos-fonte e que, tendo em vista que os textos-fonte abordam o mesmo assunto, algumas informações são comuns aos três sumários. Nesse cenário, do ponto de vista do leitor, o trabalho de leitura foi reduzido de 597 palavras dos três textos-fonte para 176 dos três sumários.

Quadro 1 – Exemplo de sumários monodocumento para a língua Portuguesa.

Sumário monodocumento 1
A seleção brasileira masculina de vôlei conseguiu a sétima vitória consecutiva na Liga Mundial ao derrotar a Finlândia por 3 sets a 0. Assim, segue invicto na competição e lidera o grupo B. Brasil e Finlândia se enfrentarão novamente; nas duas últimas rodadas receberá portugueses e finlandeses; a fase final da competição deste ano acontecerá na Rússia.
Sumário monodocumento 2
A seleção brasileira masculina de vôlei venceu a Finlândia por 3 sets a 0 e manteve sua invencibilidade na Liga Mundial-06. Este foi o sétimo triunfo consecutivo dos brasileiros na competição. Esse resultado deixou o Brasil perto de conquistar a única vaga do Grupo B.
Sumário monodocumento 3
Brasil arrasa Finlândia no primeiro confronto, nesta sexta-feira, na cidade de Tampere, pela Liga Mundial de vôlei 2006, por 3 sets a 0, com parciais de 25/17, 25/22 e 25/21 . Amanhã, as equipes voltam a se enfrentar no mesmo local. Com o resultado, o Brasil continua invicto na competição, mantendo a liderança do Grupo B da Liga, com sete vitórias em sete partidas. O time está perto da classificação para a próxima fase.

Os principais desafios presentes na tarefa de SA monodocumento envolvem, principalmente, as etapas de seleção de conteúdo e síntese do sumário. Na primeira, um método automático deve identificar quais as informações mais relevantes presentes no texto-fonte. Posteriormente, esse conteúdo deve ser inserido no sumário de forma que a leitura seja fluída e sucinta. Ou seja, um bom sumário monodocumento deve ser constituído por infor-

mações não redundantes e as sentenças devem apresentar um encadeamento lógico. Por exemplo, mesmo identificado as sentenças mais importantes em um texto, a ordenação dessas sentenças no sumário pode apresentar erros de referências, como pronomes, que podem alterar o significado do sumário ou torna-lo pouco compressível.

Um sumário multidocumento, de forma mais complexa do que a SA monodocumento, deve conter as principais informações sobre um determinado fato ou evento presente em uma coleção de textos relacionados e estática. Por exemplo, no Quadro 2, é apresentado um sumário multidocumento produzido a partir dos 3 textos-fonte que foram utilizados para produzir os sumários monodocumento dispostos no Quadro 1.

Quadro 2 – Exemplo de sumário multidocumento para língua Portuguesa.

Sumário multidocumento
A seleção brasileira masculina de vôlei conseguiu, nesta sexta-feira, a sétima vitória consecutiva na Liga Mundial ao derrotar a Finlândia por 3 sets a 0 - parciais de 25/17, 25/22 e 25/21 -, em jogo realizado na cidade de Tampere, na Finlândia.O resultado de hoje deixou o Brasil perto de conquistar a única vaga do Grupo B da Liga Mundial, que classifica o melhor de cada uma das quatro chaves, a Rússia (país-sede) e mais um time convidado pela Federação Internacional de Vôlei, para a fase final, de 23 a 27 de agosto, em Moscou (Rússia).

Do ponto de vista do leitor desse sumário (ou, usuário de um sistema de SA), a produção de um bom sumário multidocumento, composto pelas informações mais relevantes nos três textos-fonte, reduz significativamente o trabalho de leitura. Especificamente para esse exemplo, reduz-se em 83% o número de palavras em relação aos três textos da coleção. Comparando o exemplo do Quadro 2 com os exemplos do Quadro 1, nota-se que o sumário multidocumento reduz ainda mais o trabalho de leitura, com aproximadamente 101 palavras contra 176 dos três sumários monodocumento, de forma que o leitor não é privado do conteúdo mais relevante dos textos-fonte.

A SA multidocumento, em relação à SA monodocumento, implica em maiores desafios de pesquisa, que se pautam principalmente na necessidade de tratar fenômenos multidocumento (MAZIERO; Castro Jorge; PARDO, 2014), tais como: i) redundância, que é uma característica natural do cenário multidocumento e considerada um grande desafio da SA, que ocorre quando os textos-fonte apresentam informações idênticas ou muito semelhantes; ii) contradição de informação, que ocorre quando os textos-fonte apresentam informações divergentes sobre um mesmo fato ou evento; iii) estilos de escrita distintos, que se caracteriza pela possibilidade dos textos-fonte serem compostos por variações linguísticas (voz ativa ou passiva, diferentes tempos verbais, sinônimos, etc.); e iv) ordenação temporal dos fatos, que se faz presente quando a sequência cronológica das informações está inserida em diversos textos; entre outros.

No Quadro 3, são dispostas algumas sentenças extraídas do córpus CSTNews

(ALEIXO; PARDO, 2008; CARDOSO *et al.*, 2011), que exemplificam os desafios supracitados. Ressalta-se que para cada fenômeno multidocumento apresentado, são utilizadas sentenças de textos relacionados pelo assunto.

Quadro 3 – Exemplo de dificuldades encontrados na SA multidocumento.

Redundância 1
1 - A TAM afirma que "o procedimento não configura qualquer obstáculo ao pouso da aeronave". 2 - De acordo com a empresa aérea, o reversor estava travado, mas argumentou que a aeronave tinha condições de pouso normais, mesmo sem ele.
Redundância 2
1 - A maior tragédia ferroviária da história do Egito ocorreu em fevereiro de 2002, após o incêndio de um trem que cobria o trajeto entre Cairo e Luxor (sul), lotado de passageiros, e que deixou 376 mortos, segundo números oficiais. 2 - A maior tragédia ferroviária da história do Egito ocorreu em fevereiro de 2002, após o incêndio de um trem que cobria o trajeto entre Cairo e Luxor, lotado de passageiros, e que deixou 376 mortos, segundo números oficiais.
Contradição
1 - Em nota enviada após a exibição da reportagem, a TAM afirma "que não teve registro de qualquer problema mecânico neste avião no dia 16 de julho". 2 - Um dia antes do acidente, na segunda-feira, 16, o avião também teria apresentado problemas ao aterrissar em Congonhas, durante o vôo 3215, procedente de Belo Horizonte (Confins), só conseguindo parar muito próximo do final da pista.
Ordenação temporal
1 - O problema teria sido detectado pelo sistema eletrônico de checagem do próprio avião, e ainda assim a aeronave da TAM, um Airbus A320, continuou voando, com o reverso direito desligado. 2 - Um dia antes do acidente, na segunda-feira, 16, o avião também teria apresentado problemas ao aterrissar em Congonhas, durante o vôo 3215, procedente de Belo Horizonte (Confins), só conseguindo parar muito próximo do final da pista.
Mesma informação com variação de forma
1 - Renan afirmou que ele próprio pediu essa investigação ao procurador. 2 - "Eu pedi para o procurador me investigar. Eu me dispus a abrir meu sigilo", afirmou.

Nos exemplos de redundâncias no Quadro 3, notam-se que as sentenças dos respectivos textos diferentes apresentam a mesma informação. No primeiro caso, as duas sentenças, apesar de distintas, fazem referência à afirmação da empresa TAM acerca da viabilidade de pousos de uma de suas aeronaves que apresentava problemas mecânicos. Já no segundo, as sentenças são praticamente iguais, tendo como diferença apenas os item lexical "(sul)''. O fenômeno de redundância, por um lado, indica quais informações são relevantes nos textos-fonte. Entretanto, deve ser evitado no sumário, pois se admite que um bom sumário deva ser sucinto, de forma que se constitua da maior quantidade de in-

formação possível dos textos-fonte, respeitando um tamanho previamente estipulado para o sumário. Evidentemente, nesses exemplos de sentenças redundantes, o segundo caso, no qual as sentenças são praticamente idênticas, é menos complexo de ser identificado do que o primeiro.

No exemplo de contradição no Quadro 3, notam-se duas sentenças compostas por informações sobre o estado de manutenção mecânica de uma aeronave. Entretanto, na primeira sentença, descreve-se a ausência de falhas mecânicas da aeronave e, na segunda, apresenta-se a possibilidade de uma falha. A inserção dessas duas sentenças em um sumário torna o seu conteúdo incoerente. Assim, um método de SA deve identificar informações como essas para, posteriormente, ponderar qual dessas deverá compor o sumário, ou se ambas serão descartadas ou, eventualmente por meio de uma abordagem de geração de língua natural, tratar essa contradição. No primeiro caso, pode-se utilizar informação extratextual, nem sempre disponível, como: credibilidade da fonte em que a sentença foi publicada; preferência do usuário em relação às fontes de informação; qual conteúdo é mais frequente; etc. Na terceira abordagem, por sua vez, o método de SA faz uso de técnicas de geração de língua natural para modificar a contradição, como, por exemplo, modalizar contradições numéricas. No exemplificação de contradição do Quadro 3, poder-se-ia modalizar a sentença, de forma que se reportasse a possibilidade de uma falha mecânica na aeronave, mas que a empresa responsável não reportou registros de manutenção ou falha.

No exemplo de ordenação temporal, a sentença 2 apresenta fatos que ocorreram antes dos acontecimentos descritos na primeira sentença. Assim, caso essas duas sentenças fossem inseridas em um sumário, a sentença 2 deveria ser reportada antes da sentença 1. Tal ordenação seria natural se as sentenças pertencessem a um mesmo texto. Entretanto, elas ocorrem em textos diferentes. Nesse contexto, mesmo que um método de SA identifique as sentenças como relevantes para o sumário, se a ordenação temporal dessas sentenças não for identificada corretamente, o conteúdo produzido pode ser considerado incoerente.

Por fim, no exemplo de estilo de escrita, notam-se duas sentenças com o mesmo conteúdo, porém, a primeira foi redigida em discurso direto e a segunda em discurso indireto. Com isso, o processo de identificar a similaridade de conteúdo entre as sentenças pode falhar, e, consequentemente, as duas sentenças podem ser inseridas no sumário, tornando-o redundante. Além disso, o estilo de escrita exige um processo de normalização textual, por exemplo: entidades (países, pessoas, etc.) podem ser referenciadas pelos respectivos nomes, abreviações ou sinônimos; os textos podem ser redigidos em diferentes tempos verbais, de forma que um mesmo fato possa ser descrito no presente em algum texto ou no passado em outro.

A SA multidocumento parte do princípio que a coleção de textos é estática. Entretanto, dada a dinâmica dos meios de comunicação atuais, principalmente *online*, novos

textos relacionados ao tema dessa coleção textual eventualmente são produzidos. Com isso, o sumário produzido poderá se desatualizar rapidamente. No âmbito da SA multidocumento, que considera dados estáticos, poder-se-ia empregar duas abordagens para prover sumários atualizados aos leitores: i) produzir um sumário apenas dos textos recentes, que não são conhecidos pelo leitor; e ii) produzir um sumário a partir de todos os textos, os recentes e os anteriores (que já são conhecidos pelo leitor). No primeiro caso, eventualmente, os textos recentes podem apresentar informações também contidas nos textos anteriores e, consequentemente, o sumário terá informações redundantes ao usuário, ou seja, informações que já foram previamente lidas nos primeiros textos ou no respectivo sumário. Já no segundo cenário, além da possibilidade da redundância com o conhecimento do leitor, o processamento computacional é demasiado, uma vez que os textos anteriores deverão ser processados novamente.

Dado o cenário supracitado e a necessidade de processamento de dados dinâmicos, a comunidade da SA ampliou o foco de pesquisa para uma nova tarefa, a Sumarização Automática de Atualização (SAA) – referenciada em inglês como *Update Summarization* – que possui o objetivo de gerar sumários admitindo que os leitores tenham conhecimento prévio acerca dos tópicos presentes nos textos-fonte. Por exemplo, assume-se que os usuários tenham lido previamente alguns textos relacionados ao conteúdo que será sumarizado. Tendo em vista que essa tarefa é mais pertinente em um cenário dinâmico, sobretudo na web, em que há diversas fontes (produtores de conteúdo) e novos textos são publicados rapidamente, a SAA é considerada como um avanço natural da área de Sumarização Automática (DELORT; ALFONSECA, 2012).

A SAA surgiu na *Document Understanding Conference* (DUC) de 2007[3] (WITTE; KRESTEL; BERGLER, 2007), com a seguinte descrição: dada três coleções de textos sobre um mesmo assunto, o objetivo era a produção de um sumário, com não mais de 100 palavras, para cada coleção; desses sumários, o primeiro seria um sumário multidocumento tradicional e os demais seriam de atualização, ou seja, deveriam apenas reportar informações novas ou mais recentes em relação às coleções anteriores. Em 2008, durante a *Text Analysis Conference* (TAC)[4], novo nome assumido pela DUC, a tarefa de SAA foi descrita como uma simplificação da edição de 2007, com a utilização de apenas duas coleções de documentos com 10 textos jornalísticos cada.

No Quadro 4, apresentam-se dois sumários com aproximadamente 100 palavras, sendo o primeiro um sumário multidocumento tradicionais e o segundo um de atualização. Esses sumários foram construídos manualmente por meio dos três textos jornalísticos do córpus CSTNews (ALEIXO; PARDO, 2008; CARDOSO *et al.*, 2011). Para representar o cenário da SAA, os dois textos menos recentes foram empregados para a construção

[3] Disponível em: http://duc.nist.gov/duc2007/tasks.html#pilot
[4] Disponível em: http://www.nist.gov/tac/

do primeiro sumário. Assim, somente um texto da coleção, o mais recente, foi utilizado para produzir o sumário de atualização, admitindo-se que o leitor tenha conhecimento do conteúdo dos textos menos recentes. Letras e números entre colchetes foram utilizados como identificadores das sentenças, de forma que todos os identificadores iniciados com a letra M são do sumário multidocumento e com a letra A são as sentenças do sumário de atualização.

Quadro 4 – Exemplo de sumário multidocumento e de atualização.

Sumário 1 – Multidocumento Tradicional
[M1] Uma colisão entre dois trens de passageiros provocou a morte de pelo menos 80 pessoas e deixou 165 feridas. [M2] O acidente ocorreu no delta do Nilo, ao norte de Cairo, no Egito. [M3] Mais de 45 ambulâncias foram enviadas ao local do acidente para socorrer as vítimas. [M4] O governador de Qaluibiya, Adli Hussein, que se deslocou junto com outros vários altos funcionários egípcios ao local do acidente, afirmou que a colisão ocorreu quando o trem número 808 que circulava em alta velocidade se chocou com a parte traseira de outro, que vinha de Qaliub.
Sumário 2 – de Atualização
[A1] O ministro da Saúde egípcio, Hatem El-Gabaly, informou nesta segunda-feira que 57 pessoas morreram e 128 ficaram feridas no choque entre dois trens de passageiros no delta do Nilo, ao norte do Cairo. [A2] Fontes policiais e médicas informaram anteriormente que pelo menos 80 pessoas tinham morrido no acidente. [A3] O acidente ocorreu às 7h15 (1h15 em Brasília) entre um trem procedente da cidade nortista de Lardo e outro que esperava na estação Qalyoub, 20 quilômetros ao norte da capital do Egito.

No Quadro 4, percebe-se que as primeiras sentenças de cada sumário são bastante similares, mais a informação contida em A1 é mais atual, o que é ressaltado na sentença A2, na qual se reporta um possível equívoco anterior sobre o número de afetados no acidente. Com relação às sentenças M1 e A1, nota-se algum tipo de contradição que deve ser ponderada pelo método de SAA. Especificamente nesse exemplo, a sentença A1 pode ser considerada uma atualização por meio de informações extra e intratextuais, como a data mais recente do texto empregado no sumário de atualização e a presença da sentença A2. Por fim, a sentença A3 apresenta informações novas e complementares ao leitor, pois acrescenta dados sobre o horário, origem dos dois trens envolvidos no acidente e dados geográficos, que não foram reportados anteriormente, na sentença M4, por exemplo. Nesse exemplo, percebe-se o quão complexa é a tarefa da SAA, com objetivo de identificar quais informações atualizem o conhecimento do leitor.

A SAA se mostra como uma tarefa ainda mais pertinente devido ao volume de dados dinâmicos disponíveis. Por exemplo, por meio da chave de busca “Mundial de F1 2017” na ferramenta de busca de notícias Google News®, foram encontradas 1.980 notícias publicadas no período de 12 a 13 de julho de 2017; e 9.020 resultados no período de 4 a 11 de julho do mesmo ano. Na primeira coleção, com textos mais recentes, há uma grande

quantidade de documentos disponíveis, embora seja bem menor em relação à segunda coleção. Além disso, muito provavelmente, esses textos mais recentes possuem informações que também foram apresentadas nos textos anteriores. Dessa forma, por exemplo, dado um usuário que tenha conhecimento dos textos mais antigos (publicados entre 4 e 11 de julho), a produção de um sumário tradicional multidocumento a partir desses novos textos poderia apresentar informação já conhecida pelo leitor. Por outro lado, um sumário de atualização poderia disponibilizar somente as informações mais relevantes e atualizadas, que, de fato, seriam mais importantes para o usuário.

Com as diretrizes da SAA, além de também tratar os desafios presentes na SA tradicional (no decorrer deste texto, a expressão SA tradicional referenciará a SA monodocumento e a multidocumento), incluem-se novos desafios à área, que ocorrem pela necessidade de considerar os textos já lidos pelo leitor durante a sumarização, no objetivo de identificar as informações relevantes e com novidade que atualizem o conhecimento do leitor acerca do tema. Por exemplo, admitindo-se que a SAA, de certa forma, é uma tarefa naturalmente multidocumento, pois se notam a presença de ao menos dois textos, um conhecido e outro não conhecido pelo leitor, os fenômenos multidocumento, apresentados anteriormente, devem ser tratados. Entretanto, no contexto da SAA, esses obstáculos mostram-se ainda maiores, pois são presentes no cenário do conteúdo disponível para o sumário (entre os textos não conhecidos pelo leitor) e na necessidade de identificar conteúdo de atualização (entre os textos conhecidos e não conhecidos pelo leitor). A seguir, são ilustrados alguns cenários de obstáculos que devem ser tratados pela SAA, tanto no contexto da produção do sumário quanto na principal característica da tarefa, que é identificar informação nova relevante.

A redundância de informação, que é um grande obstáculo para a SA tradicional, torna-se um obstáculo ainda maior para a SAA. Para um sumário de atualização, espera-se que seu conteúdo não seja redundante e que, além disso, não seja redundante para com o conhecimento do leitor sobre o tema dos textos. Para o primeiro caso, por exemplo, o sumário deve identificar e ponderar o uso de sentenças que transmitem a mesma informação, tais como as sentenças com Redundância do Quadro 3. Já para o segundo caso, por exemplo, se um leitor tem conhecimento sobre o pronunciamento da TAM sobre a viabilidade de pouso de aeronave, sentenças com essas informações, tais como as sentenças com Redundância 1 do Quadro 3, devem ser evitadas no sumário.

Como apresentado anteriormente, durante a discussão sobre os obstáculos da SA multidocumento, a Redundância deve ser evitada no sumário, mas também indica relevância de informação (partindo-se do princípio que informações frequentes são relevantes). No cenário da SAA, porém, sentenças redundantes podem representar conteúdo relevante, com atualização se forem presentes nos textos desconhecidos, mas, também podem caracterizar um conhecimento prévio do leitor que se estendeu aos textos recentes. Nesse terceiro

caso, o método de SAA deve analisar se houve alterações nessas informações, dos textos anteriores para os recentes, que possam ser consideradas mais recentes. Por exemplo, a alteração do quadro de medalhas de algum evento esportivo, como as Olimpíadas.

A presença de contradição entre os textos recentes, de forma semelhante à SA multidocumento, obriga o método de SAA a ponderar se alguma ou nenhuma dessas informações será inserida no sumário. Por outro lado, no caso de contradição entre os textos recentes e os já lidos, tendo em vista que informações anteriormente reportadas podem sofrer alterações (atualização), o método deve identificar se essa diferença é uma possível atualização, algum diferente ponto de vista ou algum eventual erro, ou seja, deve identificar se essa contradição é uma atualização relevante.

A ordenação temporal das informações, essencial para a SA multidocumento, torna-se ainda mais necessária na SAA, pois é uma informação útil para correlacionar fatos recentes não reportados nos textos anteriores. Por exemplo, com a ordenação temporal, pode ser identificada uma sequência cronológica de descrições sobre um evento. Além disso, analogamente à SA multidocumento, se as informações dos textos recentes não forem corretamente ordenadas, o sumário produzido poderá ser considerado incoerente.

Por fim, a presença de estilos de escrita distintos, de forma similar à SAA multidocumento, exige um processo de normalização textual. Além disso, em alguns casos, essa característica pode dificultar a tarefa de identificar as informações dos textos recentes, não conhecidos pelo leitor, que já foram apresentadas anteriormente. Por exemplo, os textos conhecidos pelo leitor podem apresentar informações sobre algum determinado fato ou evento no tempo presente; e esse mesmo conteúdo pode ser descrito no passado nos textos não conhecidos pelo leitor. Nesse caso, eventualmente, sem processamento adequado, o sumário de atualização compor-se-ia de conteúdo redundante.

Na literatura, em um primeiro momento, a SAA foi abordada por meio da utilização de métodos de SA tradicionais com uma fase posterior de remoção de redundância, cujo objetivo é descartar sentenças dos textos recentes que sejam muito similares aos textos conhecidos pelo leitor. Assim, assume-se que sentenças não ou pouco similares às sentenças dos textos conhecidos pelo leitor possuem informação nova. Essa abordagem, denominada sumarização sensível ao contexto, foi muito utilizada em diversos trabalhos da área (DELORT; ALFONSECA, 2012). Geralmente, métodos de SAA nessa perspectiva utilizam métricas de similaridade lexical (por exemplo, a métrica do Cosseno (SALTON; WONG; YANG, 1975)) em um processo iterativo para identificar e remover sentenças redundantes do sumário, ou seja, uma sentença candidata ao sumário é previamente comparada com todas as sentenças dos textos conhecidos do leitor e, se for considerada não redundante, é inserida no sumário.

A estratégia anterior pode eventualmente desconsiderar sentenças com informação nova acompanhada de informação já conhecida pelo leitor, que é considerado um dos

grandes desafios da SAA, (DELORT; ALFONSECA, 2012). Por exemplo, no Quadro 4, as sentenças A1 e A3, que ocorrem no sumário de atualização, são, respectivamente, parcialmente similares às sentenças M1 e M4 do sumário multidocumento. É importante ressaltar que todas essas sentenças foram extraídas sem alterações de seus respectivos textos-fonte. Assim, as sentenças do sumário de atualização também são similares à sentenças dos textos-fonte empregados na produção do sumário multidocumento. Assim, essas informações relevantes e de atualização poderiam ser desconsideradas do sumário por meio da abordagem sensível ao contexto.

Nesse contexto, em que abordagens pouco informadas, como a sensível ao contexto, não são adequadas para tratar os desafios da SAA, sobretudo no objetivo de identificar informação nova, pretende-se, neste trabalho, investigar métodos de SAA por meio de abordagens mais informadas. Essa perspectiva também é observada em uma revisão dos trabalhos de SAA da literatura, na qual se sugere que melhores resultados são alcançados quando a relação entre o conhecimento do leitor, ou textos já conhecidos por ele, e os documentos mais recentes é representada por meio de uma abordagem mais informada, tais como relações sentenciais representadas em grafos (WENJIE *et al.*, 2008; LI; DU; SHEN, 2011), tópicos textuais (HUANG; HE, 2010; DELORT; ALFONSECA, 2012), agrupamento de sentenças similares (WANG; LI, 2010), modelos de otimização (DU *et al.*, 2010; LONG *et al.*, 2010), etc.

Na literatura, diferentes propostas e abordagens para representação de conteúdo textual vêm sendo investigadas em distintas tarefas de SA. Por exemplo, Nenkova e Vanderwende (2005), Varma *et al.* (2009) modelam os textos-fonte e respectivas sentenças por meio n-gramas e algumas métricas de relevância. Alguns outros métodos empregam Grafos e algoritmos baseados nessas estruturas, dos quais podem-se citar Erkan e Radev (2004), Lin e Kan (2007), Wenjie *et al.* (2008), Li, Du e Shen (2011), Ribaldo *et al.* (2012), Qian e Liu (2013), Li e Li (2014). Visando a identificação de relações semânticas por meio de termos relevantes e reduzir a quantidade dos dados a serem processados, há algumas propostas que identificam tópicos latentes (palavras ou expressões chaves), tais como Nastase (2008), Steinberger e Jevzek (2009), Du *et al.* (2010), Li, Du e Shen (2011), Delort e Alfonseca (2012), He, Qin e Liu (2012). Essas abordagens permitem correlacionar diferentes tipos de informação e o uso de métricas específicas dessas estruturas, tais como conectividade de grafo, que podem contribuir para identificar relevância, atualização e remover redundância de conteúdo.

Além das abordagens supracitadas, tem-se também algumas investigações mais linguisticamente guiadas. Por exemplo, Vanderwende, Banko e Menezes (2004) e Leskovec, Milic-Frayling e Grobelnik (2005) propuseram o uso de redes com relações lexicais identificadas por meio de informações sintáticas. Castro Jorge (2010), Cardoso (2014) e Ribaldo, Cardoso e Pardo (2016), no âmbito da SA multidocumento para a língua portuguesa,

fazem uso de diferentes técnicas por meio de teorias discursivas

Neste trabalho, investigaram-se abordagens distintas de representação textual para as quais foram incorporados conhecimentos semânticos e discursivos, como a Teoria Discursiva Multidocumento (CST, do inglês *Cross-document Structure Theory*) (RADEV, 2000) e a segmentação de Subtópicos. Esses teorias linguísticas foram investigadas pois alguns trabalhos prévios de SA para a língua Portuguesa, como Ribaldo *et al.* (2012), Cardoso (2014), Ribaldo, Cardoso e Pardo (2016), demonstraram que essas informações contribuíram efetivamente para a produção de sumários mais informativos.

Por meio da CST, é possível correlacionar sentenças de textos diferentes por meio de relações discursivas. Dessa forma, pode-se representar o conteúdo de uma coleção textual por meio dessa teoria visando identificar e tratar os fenômenos multidocumentos (CARDOSO, 2014). Além disso, no âmbito da SAA, é possível analisar as relações discursivas entre os textos mais recentes e àqueles considerados conhecidos pelo leitor com objetivo de identificar o conteúdo mais relevante e atualizado. Por exemplo, no Quadro 4, as sentenças A1 e M1 poderiam ser correlacionadas por uma relação CST que indica Contradição; assim, assumindo-se que A1 ocorreu em um texto conhecido pelo leitor, essa relação pode indicar que a sentença M1 é mais atual. Além disso, a teoria CST prevê algumas relações que indicam ordenação temporal ou um sequenciamento de eventos, o que pode ser extramente útil para a tarefa de SAA.

Visando a produção de sumários informativos, além de modelar/tratar os fenômenos multidocumento, pode-se também representar o conteúdo do texto-fonte de forma adequada. Nesse contexto, tem-se a teoria linguística de Segmentos de Subtópicos. Esse conhecimento linguístico parte do princípio de que a ideia ou assunto principal de um texto é composto por assuntos menores, que podem ser identificados ao decorrer do corpo textual. Assim, cada Segmento de Subtópico representa um conjunto coerente de sentenças (em sequência) no qual é descrita alguma sub-ideia que compõe o assunto principal no texto. Por meio desse tipo de informação, por exemplo, podem-se produzir sumários que cubram a maioria dos temas abordados nos textos-fonte e, mais especificamente no âmbito da SAA, verificar quais os Subtópicos que ainda não foram lidos pelo leitor e quais aqueles que sofreram alterações relevantes. Uma abordagem semelhante foi empregada em (HUANG; HE, 2010; DELORT; ALFONSECA, 2012), porém, por meio de tópicos latentes.

Segundo Mani (2001), o processo de construção automática de um sumário requer três etapas principais, referenciadas como fases de Seleção, Transformação e Síntese de Conteúdo. Nas duas primeiras, os textos-fonte são processados e transformados em um modelo computacional para que seja possível identificar o conteúdo mais relevante a ser inserido no sumário. Uma vez selecionado essa informação, o sumário é produzido por algum procedimento de Síntese. Embora essa arquitetura de SA tenha sido introduzida,

sobretudo, para as tarefas de SA Mono e Multidocumento, essas fases de processamento também são necessárias à SAA. Obviamente, algumas alterações são pertinentes para adequar essas fases de processamento aos requisitos da SAA, como a necessidade de identificar conteúdo relevante, mais recente e atualizado.

Na literatura, com relação as etapas de Seleção e Transformação, encontram-se diversas abordagens e técnicas, com alguns modelos de representação textual apresentados anteriormente, que fazem uso de pouco ou muito conhecimento linguístico. Contudo, para a etapa de Síntese, comumente encontram-se métodos de SAA que empregam uma Abordagem Extrativa, na qual algumas sentenças dos textos-fonte são selecionadas e posteriormente, sem alteração de conteúdo, são justapostas no sumário final, tais SAA (HUANG; HE, 2010; DELORT; ALFONSECA, 2012; NóBREGA *et al.*, 2014). Tal abordagem também é frequentemente adotada em outras tarefas, como a SA Mono e Multidocumento (RINO *et al.*, 2004; LEITE *et al.*, 2007; MARGARIDO *et al.*, 2008; MIHALCEA; TARAU, 2004; NASTASE, 2008; HAGHIGHI; VANDERWENDE, 2009; Castro Jorge, 2010; RIBALDO *et al.*, 2012; CARDOSO, 2014).

A abordagem Extrativa, embora seja a mais empregada, sobretudo por sua simplicidade e resultados satisfatórios, pode reduzir a informatividade do sumário produzido em algumas situações. Isso ocorre porque as sentenças selecionadas para o sumário podem conter alguns segmentos pouco relevantes ou redundantes. Dessa forma, uma vez que não há alteração do conteúdo dessas sentenças e o sumário comumente é submetido a restrições de tamanho (tais como a quantidade de caracteres, palavras ou sentenças), não há como acomodar algum outro conteúdo mais importante. Por exemplo, no Quadro 5 é apresentado um sumário extrativo que foi gerado por um método automático a partir por do córpus CSTNews (ALEIXO; PARDO, 2008; CARDOSO *et al.*, 2011). Pode-se observar, em negrito, que alguns segmentos das sentenças S2 e S3 são menos relevantes para o sumário, uma vez que essas informações também estão presentes em S1. Com a remoção desses segmentos, poder-se-ia, por exemplo, adicionar mais conteúdo no sumário, tornando-o mais informativo ao usuário. Aqui, é importante observar que em um cenário multidocumento, identificar sentenças dos textos-fonte que não possuam segmentos pouco relevantes ou redundantes, seja em relação ao conteúdo já inserido no sumário ou ao conhecimento do leitor em uma tarefa de SAA, é uma tarefa complexa, dado a dificuldade de se tratar os fenômenos multidocumento.

Outra possibilidade de síntese é a Abstrativa, na qual há um procedimento de geração automática de língua natural. Em outras palavras, o conteúdo do(s) texto(s)-fonte é analisado e "interpretado" pelo método de SA que, posteriormente, cria novas sentenças para compor o sumário. Dessa forma, com um bom gerador automático de texto, os problemas existentes na síntese Extrativa poderiam ser reduzidos. Entretanto, tal abordagem é bem mais complexa e ainda requer muitos esforços de pesquisa.

Quadro 5 – Exemplo de sumário extrativo, no qual são destacados os principais problemas da abordagem extrativa.

[S1] Quase metade dos vôos previstos para decolar **na manhã desta terça-feira (24) no Aeroporto de Congonhas, na Zona Sul de São Paulo, foi cancelada**, de acordo com informações da Infraero. **[S2]** Aeroporto de Congonhas continuava fechado para pousos **na manhã desta terça-feira, 24,** por conta do nevoeiro que cobria a região **sul de São Paulo, e metade dos vôos programados entre 6 e 8 horas foram cancelados**. **[S3]** Com o fechamento **de Congonhas** e a interdição da pista principal, que está fechada desde a terça-feira, 17, por conta do acidente com o vôo 3054 da TAM, algumas empresas transferiram vôos para o Aeroporto Internacional de São Paulo (Cumbica), em Guarulhos. **[S4]** A TAM cancelou 68 vôos previstos para esta terça em Congonhas e transferiu outros 22 para Cumbica.

Visando solucionar o problema supracitado, investigações em SA direcionaram esforços para a Abordagem de Síntese Compressiva, que pode ser considerada um meio termo entre as abordagens Extrativa e Abstrativa. Na SA Compressiva, as sentenças selecionadas dos textos-fonte são eventualmente comprimidas, ou reduzidas, antes de serem adicionadas ao sumário. Uma sentença é comprimida por meio da deleção de alguns segmentos ou por algum procedimento de reescrita (ALMEIDA; MARTINS, 2013). Dessa forma, por exemplo, segmentos considerados irrelevantes de algumas sentenças selecionadas são removidos, diminuindo assim seus respectivos tamanhos, e possibilitando a inserção de mais conteúdo no sumário. Por exemplo, os segmentos demarcados no Quadro 5 poderiam ser removidos de suas respectivas sentenças e, consequentemente, novas informações poderiam ser adicionadas.

Para a produção de sumários por meio da Síntese Compressiva, em algum estágio do processo, as sentenças selecionadas dos textos-fonte devem ser comprimidas. Nesse cenário, tem-se a tarefa de Compressão Sentencial (CS), cujo objetivo é produzir uma versão de menor tamanho (menor quantidade de itens lexicais) de uma determinada sentença (KNIGHT; MARCU, 2000). Por exemplo, no Quadro 6, são apresentadas algumas sentenças com as respectivas possibilidades de versões comprimidas. Nesses exemplos, pode ser observado que as versões menores possuem menor quantidade de itens lexicais em relação às sentenças originais e apresentam, relativamente, o mesmo significado.

Usualmente, métodos de CS empregam a abordagem de Deleção, na qual a sentença comprimida é produzida após a remoção de alguns itens lexicais e/ou arcos sintáticos em uma Árvore Sintática (de sintagmas ou dependências), tais como os modelos de compressão introduzidos em (KNIGHT; MARCU, 2000; TURNER; CHARNIAK, 2005; MCDONALD, 2006; COHN; LAPATA, 2008; ALMEIDA; MARTINS, 2013; BERG-KIRKPATRICK; GILLICK; KLEIN, 2011; THADANI; MCKEOWN, 2013).

Quadro 6 – Exemplos de sentenças com respectivas versões comprimidas.

Exemplo 1	
Original:	Estão abertas as inscrições para um passeio ciclístico em Divinópolis.
Compressão:	Abertas inscrições para passeio ciclístico em Divinópolis.
Exemplo 2	
Original:	Num comunicado em inglês enviado ontem à agência Reuters, dirigindo-se aos mercados, Passos Coelho explica em cinco pontos as razões para que o PSD vote contra o PEC 4.
Compressão:	Passos Coelho explica as razões para que o PSD vote contra o PEC 4.

Dado o contexto supracitado, a investigação deste trabalho foi direcionada à Sumarização Automática de Atualização e à Síntese Compressiva por meio de métodos de Compressão Sentencial. No primeiro, investigou-se o uso de: (i) conhecimento semântico e discursivo; e (ii) abordagens variadas de representação textual para SA. Por meio dessa abordagem, estimou-se que os desafios de SAA, que se baseiam em identificar conteúdo relevante, com atualização e não redundante ao conhecimento do leitor, pudessem ser tratados de forma mais efetiva. No segundo escopo, foram investigados métodos de Compressão Sentencial visando a produção de Sumários Compressivos para, posteriormente, avaliar as diferenças de qualidade entre os sumários extrativos e abstrativos.

Para investigação da Abordagem de Síntese Compressiva nos métodos de SA propostos, uma vez que se investigaram abordagens distintas de SA, propôs-se um método de Sumarização Compressiva em que as etapas de seleção e compressão das sentenças dos textos-fonte são independentes. Dessa forma, tanto os procedimentos de síntese extrativo e compressivo puderam ser experimentados em todos os métodos de SA propostos. Além disso, uma vez que a arquitetura proposta para empregar a SA Compressiva individualiza as etapas de seleção e síntese de conteúdo, acredita-se que tal metodologia possa ser facilmente integrada em métodos de SA previamente propostos de forma a contribuir para a produção de sumários mais informativos.

Além do cenário da investigação em SAA apresentado anteriormente, é importante ressaltar que, para a língua portuguesa, sobretudo no Brasil, embora a sumarização tradicional tenha sido bastante explorada, tanto no cenário monodocumento, com (RINO *et al.*, 2004; LEITE *et al.*, 2007; MARGARIDO *et al.*, 2008), quanto multidocumento, com (Castro Jorge, 2010; RIBALDO *et al.*, 2012; CARDOSO, 2014), não se tem conhecimento de trabalhos em SAA. Para a tarefa de SA Compressiva, tem-se principalmente o córpus *Priberam Compressive Corpus for Summarization* (PCSC) (ALMEIDA *et al.*, 2014), que possui textos jornalísticos publicados em agências Portuguesa e sumários semiautomaticamente produzidos por meio da Síntese Compressiva por meio de um processo de deleção de itens lexicais. Assim, espera-se que este trabalho seja motivador para a evolução da área de SA para língua portuguesa, de forma a prover recursos e ferramentas para o de-

senvolvimento da área e que, no contexto geral, apresente contribuições quanto ao uso de conhecimento semântico discursivo no processo de SAA.

Nas seguintes seções deste capítulo, os objetivos, hipóteses, método de pesquisa e organização deste trabalho serão apresentados.

1.2 Objetivos e tese

Como descrito anteriormente, a tarefa de SAA é muito pertinente, sobretudo no cenário da Web, tendo em vista o volume de dados textuais já existentes e o dinamismo com que novo conteúdo é publicado e em diferentes canais de comunicação. Diversas abordagens e técnicas vêm sendo investigadas no âmbito da SAA, tais como métodos baseados em distribuição de Tópicos Latentes (STEINBERGER; JEžEK, 2009; HUANG; HE, 2010; LI *et al.*, 2012; DELORT; ALFONSECA, 2012), Características Posicionais (KATRAGADDA; PINGALI; VARMA, 2009; OUYANG *et al.*, 2010), algoritmos de Grafos (ERKAN; RADEV, 2004; LIN; KAN, 2007; WENJIE *et al.*, 2008; RIBALDO *et al.*, 2012; LI; DU; SHEN, 2013), entre outras. Entretanto, tais métodos comumente não fazem uso de conhecimentos linguísticos mais refinados, que podem modelar os fenômenos multidocumento inerentes à tarefa de SAA de forma mais efetiva. Ressalta-se que alguns trabalhos prévios destinados à SA Multidocumento, tais como Ribaldo *et al.* (2012), Cardoso (2014), Ribaldo, Cardoso e Pardo (2016), sugerem que o uso de abordagens mais informadas, que empregam informações discursivas, produzem resultados mais satisfatórios. É importante também apresentar que o uso de informações/ferramentas linguísticas vem sendo estimuladas em algumas conferências da área, como a competição organizada pela *Text Analysis Conference* (TAC) de 2010, na qual os métodos submetidos deveriam analisar Aspectos Textuais que descreviam diferentes tipos de notícias jornalísticas (OWCZARZAK; DANG, 2010).

Embora diferentes conhecimentos linguísticos possam ser utilizados, sobretudo, nas etapas de Seleção e Transformação de conteúdo, com objetivo de modelar mais precisamente os desafios multidocumentos da SAA, os sumários gerados automaticamente ainda podem não ser tão informativos. Por exemplo, comumente, métodos de SAA fazem uso de uma abordagem de Síntese Extrativa, na qual não há alteração das sentenças selecionadas para o sumário. Dessa forma, mesmo com uma modelagem mais informada (linguisticamente) dos textos-fonte, como descrito anteriormente, essa técnica de síntese pode desconsiderar algumas informações pertinentes para o sumário, pois algumas sentenças selecionadas podem conter segmentos menos relevantes de forma que se impossibilita a adição de conteúdo mais importante no sumário, dada as limitações de tamanho definidas para o sumário.

Tendo em vista as lacunas identificadas na literatura, a Tese investigada neste

trabalho foi de que o uso de informação semântica e discursiva nas etapas de Seleção e Transformação de Conteúdo, aliada a diferentes abordagens de representação textual, contribui para a construção de sumários de atualização mais informativos em relação aos métodos tradicionais, pois auxiliam, principalmente, na identificação de sentenças com conteúdo de atualização inserido em meio à informação conhecida, que é considerado um dos grandes desafios da SAA e não é devidamente tratado por meio de abordagens superficiais, que fazem uso de pouco conhecimento linguístico durante o processamento, (DELORT; ALFONSECA, 2012).

Dado o contexto supracitado, o principal objetivo deste trabalho foi investigar o uso de informação semântica e discursiva em conjunto com diferentes abordagens de representação textual, que foram amplamente investigadas no âmbito da SAA, por meio das abordagens de síntese Extrativa e Compressiva. Entretanto, o uso de conhecimento semântico/discursivo em métodos de sumarização, geralmente, implica em um maior custo computacional. Assim, como segundo objetivo deste trabalho, investigaram-se abordagens híbridas com uso de informação semântica, discursiva e superficial com o propósito de produzir métodos de SAA mais escaláveis.

No contexto da SAA baseada na Abordagem de Síntese Compressiva, o objetivo dessa pesquisa foi desenvolver uma arquitetura que viabiliza a produção de sumários compressivos a partir de um método de Sumarização Extrativo. Por meio dessa Arquitetura, analisou-se o a diferença de informatividade dos sumários produzidos por meio dessa arquitetura e aqueles dos respectivos métodos de SA extrativos. Para tanto, outro objetivo foi desenvolver e avaliar métodos de Compressão Sentencial, que almejam produzir uma versão reduzida de uma sentença de entrada.

Para a língua Portuguesa, não se tem conhecimento de trabalhos e recursos específicos à SAA anteriormente a este trabalho. Assim, as experimentações principais deste trabalho foram direcionadas para esse idioma, visando ampliar o escopo da pesquisa em SA em língua Portuguesa. Dessa forma, outro objetivo deste trabalho foi desenvolver, compilar e estender recursos e ferramentas destinados à tarefa de SAA e SA Compressiva para esse idioma.

Por meio dos experimentos realizados, pôde-se observar que, na maioria dos casos, a adição de conhecimento linguístico em modelos de representação textual, comumente empregados na literatura, produz sumários de atualização ligeiramente mais informativos em relação aos sumários gerados pelos métodos tradicionais. Esses resultados ocorreram principalmente quando foi utilizado o conhecimento de Segmentação de Subtópicos. Ressalta-se que esses experimentos foram executados em conjuntos de avaliação para a língua Portuguesa e Inglesa, na qual se encontra repositórios de SAA mais difundidos.

Tendo em vista que foram investigados diferentes abordagens de SAA, que empregam diferentes formas de representação textual e algoritmos para seleção de sentenças

para os sumários, foi também proposto um método *Ensemble*, que combina alguns dos métodos investigados neste trabalho visando ranquear/identificar as sentenças mais adequadas para o sumário. Tal método apresentou resultados muito satisfatório, superando, inclusive, alguns métodos do Estado da Arte. Tal resultado indica que diferentes abordagens de representação textual são complementares e auxiliam o processo de SAA.

No âmbito da SAA baseado na síntese compressiva, foi proposta e desenvolvida uma arquitetura na qual se pode acoplar um método de sumarização extrativo e um método de Compressão Sentencial de forma a produzir sumários por meio da síntese compressiva. Ressalta-se que tal abordagem pode ser empregada em outras tarefas no contexto da SA. Após os experimentos, observou-se que adotar somente versões comprimidas das sentenças para compor o sumário não melhora a respetiva informatividade. Provavelmente, isso ocorreu pela existência de compressões agramaticais ou mal formadas que são eventualmente produzidas pelos métodos de compressão. O melhor modelo de SAA baseada na síntese compressiva foi aquele que, em cada etapa de seleção de uma sentença para o sumário, utiliza a sentença original sem alteração, que foi extraída de algum texto-fonte, ou a respectiva menor versão comprimida pelo método automático.

Previamente aos experimentos com a arquitetura supracitada, foram investigados e desenvolvidos alguns métodos de compressão sentencial para a língua Portuguesa. Esses métodos foram produzidos por meio de distintas técnicas de aprendizado de máquina com diferentes níveis de atributos. Por meio dos resultados computados durante a etapa de avaliação dos métodos, pôde-se observar que um dos modelos desenvolvidos neste trabalho apresentou valores melhores do que o método proposto por Filippova e Alfonseca (2015), que faz uso de técnicas de *Deep Learning*.

1.3 Método de trabalho

Para atingir os objetivos e verificar as hipóteses de pesquisa deste trabalho, o método de trabalho proposto correspondeu às seguintes tarefas: revisão da literatura; implementação de alguns métodos de SAA relevantes e do estado da arte na literatura; proposta de alguns métodos novos de SAA por meio de conhecimento semântico, discursivo e representações textuais; construção e adaptação de recursos necessários para o desenvolvimento/avaliação deste trabalho para a tarefa de SAA extrativa e Compressiva; avaliação dos resultados dos métodos de SAA extrativos; desenvolvimento dos métodos de Compressão Sentencial para viabilizar a produção de sumários compressivos; inserção e avaliação dos modelos de compressão aos métodos de SAA extrativos; e avaliação dos resultados e verificação das diferenças entre os sumários compressivos e extrativos.

No escopo da SAA, durante a etapa de revisão da literatura, foram analisados os principais trabalhos, sobretudo aqueles que apresentaram avanços e novas abordagens à

área. Por meio dessa tarefa, foram selecionados alguns métodos para implementação/adaptação para a língua Portuguesa, com objetivo de realizar comparações com os métodos que foram desenvolvidos por meio da abordagem proposta neste trabalho.

Para o desenvolvimento e avaliação dos métodos de SAA, foi preparado um novo córpus, o CSTNews-Update, que consiste em uma organização diferente do córpus CSTNews (ALEIXO; PARDO, 2008; CARDOSO *et al.*, 2011). Ressalta-se que o CSTNews é amplamente empregado em trabalhos de SA mono e multidocumento e possui diversos conhecimentos linguísticos manualmente identificados, entre os quais se encontram as informações que forma utilizadas neste trabalho, a CST e a identificação de subtópicos.

No escopo da Síntese Compressiva, a revisão da literatura foi direcionada aos principais trabalhos de Compressão Sentencial. Dessa forma, foram observados os atributos comumente empregados pelos principais métodos da literatura visando o desenvolvimento de novos modelos de compressão para a língua Portuguesa por meio de algoritmos de aprendizado de máquina. A utilização de métodos de CS foi necessária, pois o objetivo deste trabalho nesse escopo foi disponibilizar uma arquitetura que possibilita produzir sumários compressivos a partir de métodos de SA extrativos.

Para a produção dos métodos de Compressão Sentencial, foram compilados dois córpus. O primeiro é constituído por um conjunto de pares sentenciais compostos por uma sentença original e sua respectiva versão comprimida por meio do córpus PCSC (Córpus de Sumarização por Compressão do Priberam, do inglês *Priberam Compressive Summarization Corpus*) (ALMEIDA *et al.*, 2014). É importante informar que o córpus PCSC também é constituído por textos jornalísticos e possui sumários multidocumentos manuais. O segundo córpus proposto foi compilado por meio da metodologia automática para criação de conjuntos de dados para a tarefa de Compressão Sentencial proposta em Filippova e Altun (2013).

No que se tem conhecimento até o momento, esse trabalho foi inédito em dois cenários. No primeiro, para a língua portuguesa, sobretudo no Brasil, pois não se tem relato de trabalhos de SAA para esse idioma. Uma provável justificativa da ausência de trabalhos de SAA nesse cenário é a inexistência de córpus com sumários de atualização para avaliações de métodos propostos e, evidentemente, por ser uma tarefa recente. Assim, prevê-se a extensão do córpus CSTNews como uma contribuição que poderá incentivar pesquisas de SAA para a língua portuguesa. Já o segundo cenário de inovação desta proposta corresponde à investigação de conhecimento semântico e discursivo, principalmente a CST, no processo de sumarização, do qual não se tem conhecimento de trabalhos anteriores que tenham a investigado.

1.4 Organização da monografia

Esta monografia é organizada em 3 capítulos além da Introdução. No Capítulo 2, as teorias, recursos, formas e métricas de avaliação de métodos de SAA e definições conceituais da área são apresentados mais detalhadamente. No Capítulo 3, uma revisão dos principais trabalhos de Sumarização Automática de Atualização e Compressão Sentencial encontrados na literatura é apresentada. Os métodos de CS e SAA propostos, bem com a metodologia e resultados da avaliação, serão apresentados nos Capítulos 4 e 5, respectivamente. Por fim, as conclusões e contribuições deste trabalho serão descritos no Capítulo 6.

CAPÍTULO

2

FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, apresentam-se o aparato teórico, os recursos e o ferramental que foram utilizados durante o desenvolvimento deste trabalho. Na Seção 2.1, as principais definições da área de SA serão apresentadas, tais como classificações de métodos de sumarização e conceitos relacionados à forma ou abordagem de produção de sumários. Na Seção 2.2, as principais abordagens de avaliação de métodos automáticos de sumarização, sobretudo no escopo da SAA, serão descritas. Por fim, na Seção 2.3, os principais recursos (córpus) e ferramentas que foram utilizados neste trabalho serão listados.

2.1 Conceitos da Sumarização Automática

Métodos de Sumarização Automática são comumente categorizados conforme diversas características, tais como o número de textos-fonte utilizados, forma de síntese do conteúdo presente no sumário, objetivo do sumário e abordagem utilizada. No contexto do número de textos-fonte, um método ou sumário é classificado como monodocumento ou multidocumento, quando, respectivamente, se utiliza um ou vários textos-fonte (MCKEOWN; RADEV, 1995; RADEV; MCKEOWN, 1998). No caso específico da SAA, nota-se a presença de ao menos dois textos, um conhecido e outro não conhecido pelo leitor, mas o conteúdo do sumário é advindo apenas dos textos não conhecidos. Assim, tendo em vista que os textos conhecidos pelo leitor interferem na produção do sumário, a SAA pode ser considerada uma tarefa de sumarização naturalmente multidocumento.

Com relação à forma de síntese de conteúdo, métodos de sumarização são classificados entre extrativo, abstrativos (MANI, 2001) ou, mais recentemente, compressivos. Na sumarização extrativa, algumas sentenças dos textos-fonte são selecionadas e, posteriormente, sem alterações, são justapostas no sumário, ou seja, não há produção de conteúdo, mas sim o ranqueamento e seleção das sentenças mais relevantes dos textos-fonte. Já na sumarização abstrativa, o conteúdo dos textos-fonte é "interpretado'' pelo método e pos-

teriormente novas sentenças são produzidas para o sumário, podendo-se reescrever partes da sentenças originais.

Uma vez que a abordagem abstrativa requer procedimentos de geração automática de língua natural (mais especificamente, a produção automática de textos), que é considerada uma tarefa muito complexa que ainda demanda grandes avanços de pesquisa (BERG-KIRKPATRICK; GILLICK; KLEIN, 2011; ALMEIDA; MARTINS, 2013), a síntese extrativa é a mais empregada na área, sobretudo por ser um processo mais simples que apresenta resultados satisfatórios. Contudo, uma vez que as sentenças selecionadas para o sumário por meio dessa abordagem não são alteradas, alguns segmentos redundantes e/ou irrelevantes podem se mostrar presente. Dessa forma, dada a limitação de tamanho estipulada para o sumário, algum outro conteúdo mais importante pode ser desconsiderado, pois não há mais como acomodá-lo.

Dadas as características supracitadas, pesquisadores têm direcionado esforços para a abordagem de Sumarização Compressiva (KNIGHT; MARCU, 2000; MADNANI *et al.*, 2007; BERG-KIRKPATRICK; GILLICK; KLEIN, 2011; LI *et al.*, 2013; QIAN; LIU, 2013; ALMEIDA; MARTINS, 2013; FILIPPOVA *et al.*, 2015), que pode ser considerada um meio terno entre as duas anteriores. Na Sumarização Compressiva, uma sentença previamente selecionada para compor o sumário é eventualmente comprimida de forma que seu tamanho é reduzido sem alteração (preferencialmente) de seu significado essencial por meio de um processo de reescrita ou deleção, no qual alguns itens lexicais são removidos. Dessa forma, segmentos de informação irrelevantes ou que já estejam presentes no sumário podem ser descartados de forma a possibilitar a adição de novas sentenças ao sumário, tornando-o mais informativo.

No Quadro 7, são apresentados três sumários multidocumento produzidos por humanos por meio das abordagens de síntese extrativa, abstrativa e compressiva a partir da coleção de textos do córpus CSTNews, que é composta por notícias sobre a Liga Mundial de Vôlei de 2006.

Nesse exemplo, pode-se perceber que, embora a quantidade de palavras dos dois sumários seja similar (respectivamente, 84 e 82), o sumário abstrativo apresenta mais informações que o sumário extrativo. Essa diferença dos sumários ocorreu devido ao trabalho de reescrita, no qual o sumarizador humano que produziu o segundo sumário optou por descartar alguns dados, como a pontuação da partida, para inserir mais conteúdo sobre o tema. Além disso, pode-se também observar que o sumário compressivo possui um tamanho menor em relação aos demais. Mesmo assim, é constituído pelas informações mais relevantes presentes no sumário extrativo. Dessa forma, alguma outra sentença poderia ser eventualmente adicionada ao sumário, tornando-o mais informativo.

Com relação á abordagem, um método de sumarização é classificado como profundo ou superficial, quando se faz uso de, respectivamente, muito ou pouco conhecimento

Quadro 7 – Exemplos de sumários extrativo, abstrativo e compressivo.

Sumário extrativo
O Brasil arrasou a Finlândia no primeiro confronto entre as seleções, nesta sexta-feira, na cidade de Tampere, pela Liga Mundial de vôlei 2006, por 3 sets a 0, com parciais de 25/17, 25/22 e 25/21. Amanhã, as equipes voltam a se enfrentar, às 12h30 (horário de Brasília), no mesmo local, com acompanhamento ao vivo do Terra Esportes. Com o resultado, o Brasil continua sendo a única equipe invicta da competição, mantendo a liderança do Grupo B da Liga, com sete vitórias em sete partidas.
Sumário abstrativo
Nesta sexta-feira, a seleção brasileira masculina de vôlei ganhou da seleção da Finlândia por 3 sets a 0. Com esta vitória, a seleção brasileira acumulou 7 vitórias consecutivas na Liga Mundial, sendo invicta na competição. O jogo de volta será neste sábado às 12h30. Lider no Grupo B, a seleção brasileira está perto de obter a única vaga do Grupo B na Liga Mundial. Nesta última fase, participaria junto com a Rússia (sede) e um time convidado pela Federação Internacional de Vôlei.
Sumário compressivo
O Brasil arrasou a Finlândia nesta sexta-feira, na cidade de Tampere, pela Liga Mundial de vôlei 2006, por 3 sets a 0. Amanhã, as equipes voltam a se enfrentar, às 12h30 (horário de Brasília), no mesmo local, com acompanhamento ao vivo do Terra Esportes. O Brasil continua sendo a única equipe invicta da competição, mantendo a liderança do Grupo B da Liga, com sete vitórias em sete partidas.

linguístico (MANI, 2001). Além disso, podem-se considerar métodos híbridos, quando conhecimentos profundos e superficiais são empregados em conjunto. É importante ressaltar que essa é uma classificação com relação ao tipo de conhecimento ou informação linguística que é utilizado, o que não considera a técnica, tal como uso de Aprendizado de Máquina. Na Figura 1, ilustram-se os níveis de conhecimento linguísticos dispostos conforme o nível de complexidade e subjetividade sugerido por Jurafsky e Martin (2009). Assim, desconsiderando-se o nível fonológico, destinado aos processos de fala, pode-se considerar um método como profundo quando esse se utiliza de conhecimentos nos níveis mais altos da figura, tais como discursivo e semântico.

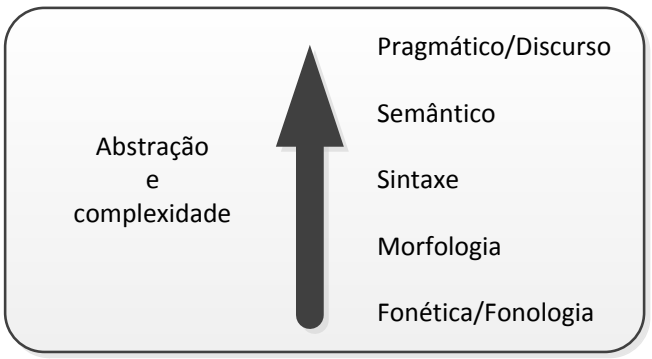

Figura 1 – Níveis de conhecimentos linguísticos.

Por exemplo, para a produção de sumários coesos e coerentes, sobretudo por meio da Abordagem Abstrativa, informações do nível Sintático podem ser utilizadas visando a produção/seleção de sentenças bem formadas. Na abordagem Extrativa, em que não há geração de novas sentenças, informação sintática pode ser adotada com objetivo de identificar um encadeamento adequado para sentenças selecionadas para o sumário. Além disso, alguns trabalhos, como Vanderwende, Banko e Menezes (2004), Leskovec, Milic-Frayling e Grobelnik (2005), fizeram uso de relações semânticas extraídas a partir das Árvores Sintáticas das sentenças presentes em um texto-fonte para identificar quais as informações mais relevantes que devem ser inseridas no sumário. Segundo esses autores, por meio dessa abordagem é possível analisar alguns eventos que são descritos nos textos.

No contexto da SA, conhecimento discursivo geralmente é utilizado para modelar os fenômenos multidocumento, como em Cardoso e Pardo (2016) e Ribaldo, Cardoso e Pardo (2016), que fazem uso de relações discursivas entre sentenças de textos diferentes, e/ou representar o conteúdo textual de forma mais adequada e informada.

Independentemente de como um método de sumarização é categorizado, o tamanho do sumário produzido geralmente é restringindo por uma Taxa de Compressão, que pode ser definida por meio de um número ou percentual de sentenças ou palavras. Nas competições das TAC 2008 e 2009, por exemplo, a taxa de compressão definida foi de 100 palavras. Nos trabalhos de Castro Jorge (2010), Ribaldo *et al.* (2012), Cardoso (2014), no âmbito da SA multidocumento para a língua portuguesa, a taxa de compressão foi de 30% em relação ao número de palavras do maior texto-fonte.

É importante ressaltar que a taxa de compressão influencia diretamente na qualidade do sumário gerado. Por exemplo, a definição de uma taxa muito restritiva (poucas palavras ou sentenças), faz com que muitas informações relevantes nos textos-fonte sejam desconsideradas no sumário, mesmo que o método de SA automática tenha uma excelente qualidade. Nesse caso, o método de sumário deve ser muito preciso, de forma que as informações realmente mais importantes sejam identificadas. Por outro lado, uma taxa de compressão muito permissiva pode contribuir para a produção de sumários que, embora mais abrangentes, dificultam a leitura das informações mais importantes por parte do leitor.

Outra classificação pertinente para a SA diz respeito ao objetivo do sumário. Nesse cenário, um sumário pode ser considerado indicativo, informativo ou crítico (MANI, 2001). Um sumário é dito como informativo quando seu objetivo é apresentar as principais informações presentes em um ou mais textos ao leitor. Sumários indicativos são utilizados para auxiliar o leitor em tarefas de seleção de conteúdo e geralmente possuem uma menor estrutura linguística. Por exemplo, o índice dos capítulos e seções desta monografia. Em uma livraria, pode-se também usar um sumário indicativo que apresente informações sobre o gênero, idioma, origem, se o livro é um *bestseller* com intuito de auxiliar o lei-

tor a selecionar um próximo livro para leitura. Por fim, um sumário é considerado crítico quando composto por uma análise ou avaliação sobre algo. Além disso, em alguns cenários, como a análise de opinião de produtos, é comum a utilização de sumários contrastivos, na qual o objetivo é apresentar as principais diferenças de informação ou características entre duas entidades ou fatos, como produtos, partidos políticos ou eventos esportivos (LERMAN; MCDONALD, 2009).

No Quadro 8, são dispostos três sumários com diferentes objetivos. Primeiramente, é apresentado um sumário do tipo informativo que, consequentemente, foi construído por meio das informações mais relevantes presentes em uma coleção de textos jornalísticos relacionados. Por outro lado, pode-se observar que no segundo sumário, que é do tipo crítico, são apresentadas as opiniões/críticas mais pertinentes de um leitor com relação a um livro. Por fim, tem-se um sumário contrastivo, no qual são dispostas as principais características de dois *Smartphones* e as respectivas diferenças entre os dois.

Quadro 8 – Exemplo de sumários com diferentes objetivos.

Sumário Informativo
A forte chuva em São Paulo deixou a cidade em estado de atenção na manhã desta segunda-feira, 16. O CGE (Centro de Gerenciamento de Emergências) registrou oito pontos de alagamento, sendo dois intransitáveis. Às 9 horas, a capital teve 113 km de lentidão, sendo que a média para esse horário é de 82 km, segundo a CET (Companhia de Engenharia de Tráfego). Mesmo após a suspensão do estado de atenção, às 9h25, o trânsito permaneceu lento. Houve registro de algumas ocorrências, sendo a mais grave na zona sul de São Paulo, onde três pessoas ficaram ilhadas em uma casa. Não há informações sobre o estado de saúde dos feridos.
Sumário Crítico
O "O outro lado da meia noite" possui uma história boa e muito envolvente. O desenvolvimento dos personagens é fantástico, o que cria um vínculo muito forte e próximo com os leitores. O final é previsível, desde o primeiro capítulo, mas a divisão da trama sobre a perspectiva de duas personagens distintas torna o desenvolvimento da história interessante, imprevisível e cheio de surpresas para entender como os universos das personagens tão diferentes se conectam na conclusão. O livro é organizado em capítulos muito grandes e em alguns trechos o desenvolvimento é lento, o que deixa a leitura monótona e desestimulante.
Sumário Contrastivo
O Aparelho 1 é configurado com SO Android e possui dimensões iguais a 155.4 x 75.2 x 7.7 mm, com 165 gramas. Além disso, possui suporte à Quad Band. O Aparelho 2, que vem equipado com o mesmo SO e tem suporte à Quad Band, é um pouco maior, com 162.5 x 74.8 x 8.6 mm e 195 gramas.

Além das características supracitadas, um sumário pode ser classificado conforme a relação com o leitor entre sumário genérico ou com foco no interesse do usuário. Um sumário é considerado genérico quando nenhuma preferência do leitor ou algum direcionamento é levado em consideração, tal como a Sumarização Multidocumento de textos

jornalísticos. Por outro lado, um sumário é dito com foco no interesse do usuário quando algo previamente estipulado pelo leitor direciona o processo de sumarização, tais como: uma pergunta do leitor, na qual se espera que o sumário responda esse questionamento; um direcionamento temático, em que o sumário deve ser constituído de informação mais relacionada ao tema adotado; ou, como no caso da SAA, o conhecimento do leitor sobre um assunto a ser sumarizado.

2.2 Avaliação de métodos de Sumarização Automática

Nesta seção, serão apresentadas as principais abordagens de avaliação de sumários produzidos por métodos automáticos de sumarização, sobretudo no contexto da SAA. Para avaliação de sumários, geralmente, verifica-se sua informatividade e qualidade do conteúdo produzido. Na informatividade, o objetivo é mensurar o grau de informação em um sumário. Já no âmbito da qualidade, analisam-se erros gramaticais, referências ausentes (como pronomes não resolvidos), facilidade de leitura, etc.

As abordagens automáticas (que serão apresentadas na Seção 2.2.1), comumente, são utilizadas para avaliar a informatividade de um sumário por meio de sua correlação com algumas fontes de referência. Essas fontes são frequentemente produzidas por humanos, e, entre as mais comuns, encontram-se sumários. Entretanto, há algumas variações, como no caso da métrica da Pirâmide (NENKOVA; PASSONNEAU, 2004), que faz uso de uma estrutura que organiza os conceitos que devem ocorrer no sumário ordenados por relevância. Entre as abordagens automáticas, no contexto da SAA, destacam-se: ROUGE (LIN, 2004); Elementos Básicos, referenciada como BE (do inglês *Basic Elements*) (HOVY; LIN; ZHOU, 2005); e Pirâmide (NENKOVA; PASSONNEAU, 2004).

Já as avaliações manuais (que serão apresentadas na Seção 2.2.2), que são efetuadas por julgadores humanos, avaliam informatividade e qualidade linguística do conteúdo produzido. Por exemplo, na Responsividade (traduzida do termo em inglês *Responsiveness*), os avaliadores verificam se é possível responder questões previamente criadas sobre o tema dos textos-fonte por meio do conteúdo presente no sumário. Além disso, pode-se verificar a presença de erros gramaticais ou referenciais (pronomes não resolvidos) nos sumários.

2.2.1 Avaliações automáticas

No contexto da avaliação automática, comumente, os métodos de SAA são avaliados da mesma forma que trabalhos de sumarização mono e multidocumento. Nesse contexto, o sumário produzido é comparado com um ou mais textos de referência (geralmente, sumários produzidos por humanos) por meio de alguma métrica de avaliação, tais como: ROUGE (LIN, 2004); Elementos Básicos, referenciada como BE (do inglês *Basic*

Elements) (HOVY; LIN; ZHOU, 2005); Pirâmide (NENKOVA; PASSONNEAU, 2004), e *Nouveau-ROUGE* Conroy, Schlesinger e O'Leary (2011).

A ROUGE, amplamente empregada na literatura de Sumarização Automática, mensura a qualidade de um sumário por meio do número de n-gramas que esse possui em comum com um ou mais textos de referência. Essa métrica é bastante versátil e possui diversas configurações, entre as quais, no âmbito da SAA, destacam-se as seguintes: (R-1), que verifica sobreposições de unigramas; (R-2), que verifica sobreposição de bigramas; (R-L), que verifica a maior sequência de *tokens* (ou itens lexicais) comuns entre os textos de referência e o sumário sendo avaliado; e (R-SU4), que verifica a sobreposição de todos os possíveis bigramas ordenados (de acordo com a posição na sentença) com no máximo quatro itens lexicais (ou 4-gramas) de distância no texto.

Na métrica de avaliação BE, a qualidade de um sumário é aferida pelo número de unidades coerente ou elementos básicos que esse possui em comum com as fontes de referências. Um elemento básico é composto por uma ou mais palavras que representam segmentos coerentes e semânticos, tais como: Estados Unidos da América, forte chuva, engarrafamento, etc. Os elementos básicos são identificados por meio de informações sintáticas, tais como: a *head* de um constituinte na árvore sintática e relações sintáticas de dependências. Em um primeiro momento, elementos menores são identificados (itens lexicais isolados), e, posteriormente, outras unidades são identificadas e concatenadas para compor elementos básicos maiores.

Segundo Hovy, Lin e Zhou (2005), a BE, além de permitir diversas configurações para identificar os elementos básicos, possibilita distintas abordagens para verificar quais são os elementos básicos sobrepostos entre os sumários avaliados e os textos de referência, tais como: i) correlação lexical, na qual os elementos sobrepostos são aqueles que são compostos pelos mesmos itens lexicais; ii) correlação de lemas, semelhante à primeira, porém, os itens lexicais são lematizados; iii) correlação entre sinônimos, que permite que unidades com palavras sinônimas sejam correlacionadas; iv) correlação por meio de similaridade lexical, na qual os elementos são correlacionados por meio da métrica do cosseno; e v) correlação por meio da generalização de unidades, que permite que unidades hipônimas ou hiperônimas sejam correlacionadas.

É importante ressaltar que a ROUGE e a BE, de forma distinta às demais métricas que serão apresentadas nesta seção, podem ser consideradas métodos que correlacionam o sumário produzido com os textos de referência. Em outras palavras, esses métodos discretizam os sumários e os textos de referência em componentes menores (por exemplo, n-gramas para ROUGE e elementos básicos para a BE) e posteriormente os comparam. Com isso, a quantidade de componentes sobrepostos entre sumários e textos de referência é identificada e, geralmente, esse resultado é ponderado em diferentes formas, como a Precisão, Cobertura e Medida-F.

Na precisão, o objetivo é ponderar se o conteúdo selecionado automaticamente para o sumário é realmente relevante. Na Cobertura, por um ponto de vista mais abrangente, o intuito é verificar se o conteúdo dos textos de referência está presente nos sumários. Por fim, a medida-F é uma soma ponderada com relação às métricas anteriores. Matematicamente, a precisão e a cobertura são calculadas por meio da divisão do valor aferido por alguma métrica pelo número de componentes presentes no sumário ou nos textos de referência, respectivamente. A seguir, são apresentadas as equações para precisão, cobertura e medida-F, sendo que *valor* corresponde ao valor aferido por alguma métrica, $|sum|$ indica a quantidade de componentes no sumário e $|ref|$ contabiliza a quantidade de componentes dos textos de referência.

$$\mathbf{P} = \frac{valor}{|sum|} \tag{2.1}$$

$$\mathbf{C} = \frac{valor}{|ref|} \tag{2.2}$$

$$F = \alpha * P + (1 - \alpha) * C \tag{2.3}$$

Por exemplo, para a BE, as variáveis das equações 2.1, 2.2 e 2.3 seriam substituídas da seguinte forma: *valor* = número de elementos básicos que ocorreram simultaneamente no sumário e nos textos de referência; $|sum|$ = número de elementos básicos identificados no sumário; e $|ref|$ = número de elementos básicos presentes nos textos de referência, descartando-se duplicatas. Para a ROUGE, esse cenário seria semelhante, mas os elementos básicos seriam substituídos por n-gramas conforme a configuração selecionada da ROUGE. Ressalta-se que o valor α é independente do método utilizado e pode ser estipulado como desejado. Usualmente, tendo em vista que os valores de Precisão e Cobertura são igualmente importantes para a avaliação de sumários, o valor de α é definido como 0,5.

A métrica da Pirâmide baseia-se na cobertura de unidades básicas de conteúdo EDUs (do inglês *Elementary Discourse Units*), que são palavras ou expressões que definem conceitos importantes em um texto. Essas unidades são identificadas de forma manual por meio da análise dos textos-fonte ou respectivos sumários de referência. Após essa etapa manual, as EDUs são ranqueadas em níveis de relevância conforme a presença nos sumários. Por exemplo, dado 3 textos de referência, as unidades do primeiro nível são aquelas que ocorrem nos três textos, em seguida, encontram-se as que ocorreram em apenas dois textos, e, por fim, as unidades presentes em apenas um texto. Com esse método, tem-se a organização de relevância dos conteúdos presentes nos textos de referência. Essa organização pode ser graficamente representada por meio de uma pirâmide, que nomeia a métrica, pois Nenkova e Passonneau (2004) observaram que a quantidade de unidades de conteúdo em um nível é inversamente proporcional a respectiva relevância.

Após a etapa supracitada, a qualidade de um sumário pode ser aferida pelo número e nível em que se encontram suas respectivas EDUs na Pirâmide identificada. Dessa forma, sumários que possuem unidades nos níveis mais relevantes, mais altos, são melhores do que aqueles que possuem conceitos dos níveis inferiores da pirâmide. Para se atribuir um valor numérico de qualidade ao sumário, as EDUs podem ser contabilizadas de forma ponderada em relação ao nível da pirâmide. Por exemplo, na Figura 2 são ilustradas duas possíveis representações visuais dos conceitos presentes em dois sumários já alocados na Pirâmide. O sumário A é considerado melhor do que o sumário B, pois seus respectivos conceitos de informação estão nos níveis superiores da estrutura, ou seja, são mais relevantes.

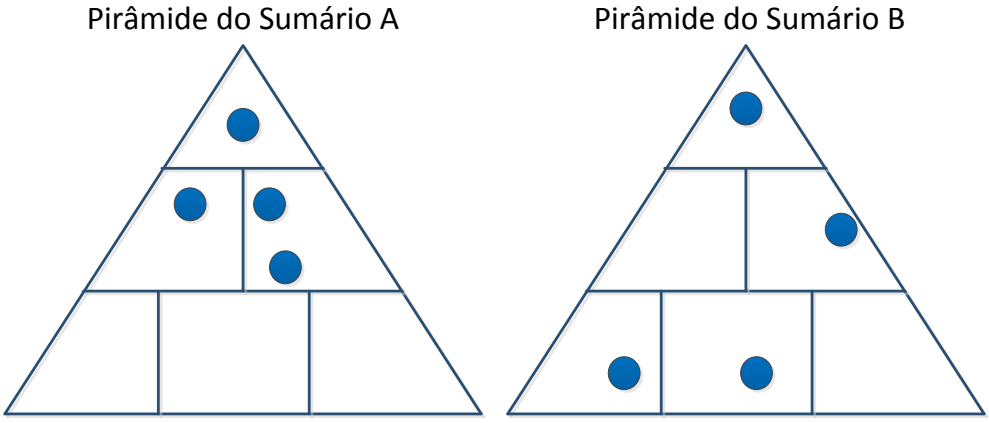

Figura 2 – Exemplo de avaliação visual da Pirâmide para dois sumários.

Segundo Conroy, Schlesinger e O'Leary (2011), os métodos de avaliação apresentados anteriormente não avaliam corretamente as características da SAA, pois foram desenvolvidas para os métodos de SA tradicionais. Nesse contexto, Conroy, Schlesinger e O'Leary (2011) propõem a utilização da Nouveau-ROUGE (N-R), com o objetivo de enfatizar a necessidade de conteúdo com atualização. Em experimentos sobre o córpus da TAC de 2008, os autores verificaram que a N-R possui uma correlação maior com as avaliações manuais em relação às demais métricas automáticas.

Conroy, Schlesinger e O'Leary (2011) partem do princípio que um bom sumário de atualização deve ser composto por mais informações dos textos não conhecidos pelo leitor. Para tanto, dado o cenário em que há sumários de referência para os textos conhecidos e não conhecidos, como disponível nas tarefas da DUC 2007 e TAC 2008, os autores propõem uma equação que pondera o valor ROUGE de um sumário para as referências dos textos não conhecidos e conhecidos pelo leitor. Essa equação é apresentada a seguir, sendo que: R_i^{AB} é o valor aferido por alguma configuração ROUGE (R-1, por exemplo) entre o sumário produzido e as referências dos textos conhecidos; R_i^{BB} análogo à anterior, porém, com as referências dos textos não conhecidos; e $\alpha_{i,0}, \alpha_{i,1}, \alpha_{i,2}$ os respectivos parâmetros da configuração ROUGE utilizada.

$$N_i = \alpha_{i,0} + \alpha_{i,1} R_i^{AB} + \alpha_{i,2} R_i^{BB} \tag{2.4}$$

Os parâmetros α, segundo os autores, foram definidos em um processo de otimização com objetivo de maximizar a correlação entre o valor aferido por essa abordagem e a métrica da Pirâmide e Responsividade (que será apresentada na Seção 2.2.2), como apresentado no Quadro 9.

Quadro 9 – Parâmetros α da *Nouveau-ROUGE*.

Configuração ROUGE	Responsividade			Pirâmide		
	$\alpha_{i,0}$	$\alpha_{i,1}$	$\alpha_{i,2}$	$\alpha_{i,0}$	$\alpha_{i,1}$	$\alpha_{i,2}$
R1	-0,0271	-7,3550	13,4227	-0,2143	-1,9011	3,1118
R2	0,9126	-5,4536	21,1556	-0,0143	-1,3499	4,3778
RSU4	1,1381	-2,6931	35,8555	0,0346	-1,1680	7,2589
RBE	1,0602	-5,0811	24,8365	0,0145	-1,3156	5,0446

As métricas automáticas apresentadas anteriormente definem a informatividade de um sumário em relação à algum modelo previamente definido. Por exemplo, para a ROUGE e a Noveau-ROUGE, comumente se utilizam sumários construídos por humanos. Por outro lado, para a métrica da Pirâmide, o sumário é analisado conforme uma hierarquia de EDUs criada manualmente. Entretanto, a produção manual desses modelos pode demandar muitos recursos financeiros e de tempo. Nesse cenário, alguns autores investigaram diferentes abordagens de avaliações automáticas que não requerem a construção manual de modelos, como sumários ou Pirâmides de EDUs.

Louis e Nenkova (2009), por exemplo, investigaram a aplicação da ROUGE e algumas outras métricas de similaridade sentencial para mensurar a informatividade de sumários considerando os respectivos textos-fonte como modelos. Segundo os autores, embora essa abordagem possa disponibilizar diferentes valores de avaliação em relação à utilização de sumários de referência, a ordenação dos métodos mantém-se equivalente pois há uma alta correlação entre essa abordagem de avaliação com metodologias com modelos manuais, como a métrica da Pirâmide. Em outras palavras, se um determinado método $m1$ apresenta resultados melhores do que outro $m2$ por meio da utilização de sumários de referência, essa diferença se mantém quando os textos-fonte são definidos como referência.

2.2.2 Avaliações subjetivas

Nas avaliações subjetivas, cada característica previamente estipulada do sumário é avaliada por um ou mais julgadores humanos. Os avaliadores geralmente realizam uma etapa anterior de treinamento e, durante a avaliação, devem ponderar cada item em um nível da qualidade, tal como, por exemplo, a escala empregada nas edições da DUC e posteriores TAC: 1) muito ruim; 2) ruim; 3) aceitável; 4) bom; 5) muito bom.

No contexto da Qualidade Linguística (QL), segundo as diretrizes da TAC (DANG, 2005), um sumário deve ser avaliado por meio de cinco características principais: i) gra-

maticalidade; ii) não redundância; iii) clareza referencial; iv) foco textual; e v) estrutura textual e coerência.

Na gramaticalidade, a ausência de erros de formatação, de capitalização (nomes próprios não iniciados por letra maiúscula, por exemplo) ou segmentos gramaticalmente incorretos é verificada. Assim, espera-se que um sumário com alta pontuação nesse item possui poucos erros com esses.

No quesito de não redundância, admitindo-se que um bom sumário deve apresentar a maior quantidade de informação possível respeitando a taxa de compressão estipulada, um sumário é ponderado conforme a ausência de repetições não necessárias de informações. Essa análise deve ocorrer em diferentes níveis, tais como, a redundância de dados/fatos sobre algum evento, de sentenças e de nomes (uma pessoa, lugar ou entidade pode ser, quando possível, referenciada por meio de pronomes).

Com relação à clareza referencial, um sumário é bem ranqueado quando as respectivas referências textuais (pronomes, nomes, etc.) não são ambíguas. Um sumário também deve possuir foco (item iv) de forma que todas as sentenças devem ser relacionadas ao tema geral abordado.

Por fim, por meio do último item sugerido pela TAC, um sumário também ser avaliado pela estrutura e coerência textual que apresenta. Por exemplo, espera-se em um bom sumário informativo não seja constituído por informações divergentes sobre um mesmo fato ou evento.

A Responsividade (traduzida do termo em inglês *Responsiveness*) foi proposta com objetivo de avaliar o conteúdo do sumário com relação à quantidade e relevância da informação que o compõe. Para tanto, segundo as diretrizes da DUC, dado alguns direcionamentos prévios, como perguntas ou descrições dos tópicos que o sumário deve responder ou abordar, o juiz deve ponderar se o conteúdo do sumário satisfaz esses direcionamentos iniciais. Em um primeiro momento, o avaliador deve verificar a quantidade de informação e, posteriormente, a qualidade e fluidez de leitura do sumário sendo avaliado.

Em geral, as metodologias subjetivas ou manuais para avaliação de sumários são muito mais refinadas e precisas do que as abordagens automáticas. Contudo, tais processos demandam mais recursos de tempo e, eventualmente, financeiros. Por exemplo, para avaliação de um conjunto com vários sumários por meio da Qualidade Linguística, deve-se considerar uma equipe com número razoável de avaliadores que, eventualmente, devem receber um treinamento inicial para esclarecer as características da tarefa e as questões definidas em (DANG, 2005). Além disso, o tempo necessário e/ou número de avaliadores requisitados pode ser maior considerando a experimentação de inúmeros métodos.

Dado o cenário supracitado, justifica-se o uso frequente de metodologias automáticas de avaliação, das quais se destaca o uso da ROUGE. Embora essas abordagens

possuem algumas limitações, são amplamente difundidas e empregadas na literatura.

2.3 Recursos e ferramentas

Nesta seção, serão apresentados os principais recursos e ferramentas que foram empregados neste trabalho. Na Seção 2.3.2, descrevem-se os seis córpus que foram aplicados para desenvolver e avaliar os métodos de SAA e Compressão de Sentenças propostos neste trabalho, sendo que três destes possuem textos em língua portuguesa e os demais são para a língua inglesa. Na Seção 2.3.3, disserta-se sobre as ferramentas de processamento de língua natural, sobretudo para extração ou identificação de conteúdo linguístico, como segmentação de subtópicos, informação sintática e relações discursivas, que foram utilizados neste trabalho.

2.3.1 Recursos teóricos

Radev (2000) propões a Teoria Discursiva Multidocumento (CST, do inglês, *Cross-document Structure Theory*), com objetivo de modelar as relações discursivas entre pares de sentenças de textos distintos. Essa teoria pode ser representada por um grafo discursivo, no qual os vértices (nós) encapsulam as sentenças dos textos e as relações discursivas entre essas são representadas pelas arestas, que podem identificar similaridades, contradições, sequências de conteúdo, variações de escrita, etc.

Originalmente, os autores propuseram um conjunto de 28 relações discursivas, que estão dispostas no Quadro 10. Essas relações indicam correlações de conteúdo, quando são consideradas as informações contidas nas sentenças, e de forma, quando se observa a redação das sentenças. Por exemplo, as relações de conteúdo *Contradiction*, *Description* e *Identity* ocorrem entre duas sentenças quando, respectivamente, há informação divergente sobre um mesmo fato, uma sentença reporta descrições sobre algo apresentado em outra e quando as duas sentenças são similares. Por outro lado, as relações de forma, como *Indirect speech* e *Translation*, são presentes quando duas sentenças reportam uma mesma informação em discursos diferentes (uma em discurso direto e outra em indireto) e quando uma sentença tem algo da outra em outra língua.

Quadro 10 – Relações discursivas da CST propostas por Radev (2000).

Agreement	*Attribution*	*Change of perspective*	*Citation*
Contradiction	*Contrast*	*Cross-reference*	*Description*
Elaboration	*Equivalence*	*Follow-up*	*Fulfillment*
Generalization	*Historical background*	*Identity*	*Indirect speech*
Judgment	*Modality*	*Parallel*	*Reader profile*
Refinement	*Subsumption*	*Summary*	*Translation*

A CST foi investigada por diversos pesquisadores, que propuseram variações ou diferentes conjuntos de relações discursivas. Para a língua portuguesa, por exemplo, Maziero, Castro Jorge e Pardo (2010) optou o uso de apenas 14 relações do conjunto inicial, pois, durante o processo de anotação manual do córpus CSTNews (vide Seção 2.3.2), identificaram algumas relações muito similares e outras que não se mostraram presentes no córpus.

Na Figura 3, é disposta a hierarquia CST proposta por Maziero, Castro Jorge e Pardo (2010). No primeiro nível, as relações foram divididas em dois grupos principais: relações baseadas em conteúdo e de forma (redação). A primeira parte dessa divisão, relações de conteúdo, foi arranjada em três menores, que correspondem às relações de relações de redundância (subdividida entre total e parcial), complementaridade (subdividida entre temporal e não temporal) e contradição. Já a segunda parte inicial, relações de forma, foi configurada entre relações correlacionadas ao estilo de redação ou fonte/autora.

A taxonomia de Maziero, Castro Jorge e Pardo (2010) é interessante por permitir diferentes níveis de refinamento da descrição discursiva multidocumento. Por exemplo, uma vez identificado que relações de complementaridade são eficazes para uma determinada tarefa, um método computacional pode ser desenvolvido para categorizar relações desse tipo em um nível mais amplo, sem distinguir *Hitorical-Background*, *Follow-up* e *Elaboration*, o que diminuiria a complexidade de classificação e, provavelmente, contribuiria para uma melhor acurácia do resultado.

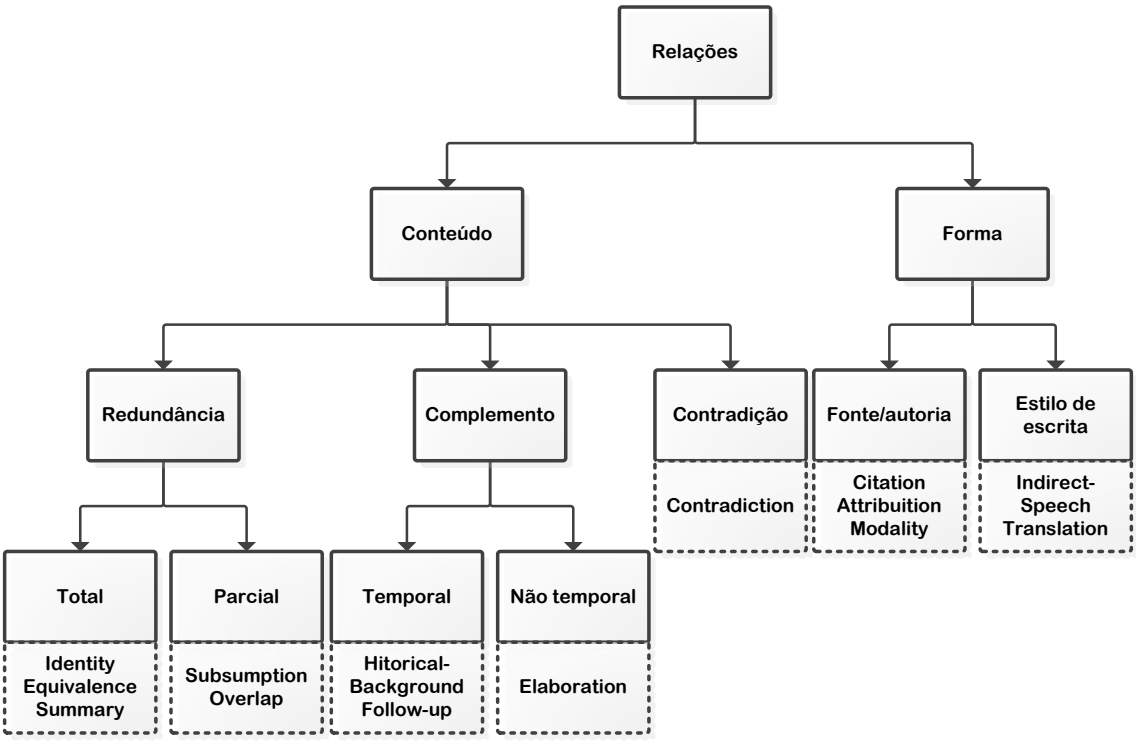

Figura 3 – Taxonomia de relações CST segundo Maziero, Castro Jorge e Pardo (2010).

Na Figura 4, são dispostas exemplos de relações discursivas, segundo a taxonomia proposta por Maziero, Castro Jorge e Pardo (2010), entre quatro sentenças de dois textos do córpus CSTNews. Pode-se observar três relações CST, sendo uma dessas uma relação de forma, a *Atribuition*, entre as últimas sentenças listadas de cada texto. A primeira sentença do texto 1 possui duas relações de conteúdo para o texto 2, uma de *Subsumption* e outra de *Elaboration*.

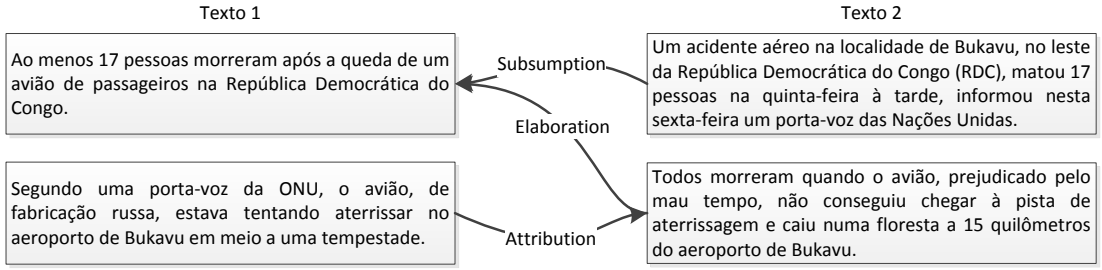

Figura 4 – Exemplo de relações CST entre textos.

A CST foi investigada em diversas tarefas e, para a língua portuguesa, principalmente no contexto da SA, sobretudo no cenário multidocumento. Nesse contexto, podem-se citar os métodos de SA de (Castro Jorge, 2010; CARDOSO, 2014) e um método de alinhamento entre sentenças dos textos-fonte e respectivos sumários que foi proposto por Agostini, Condori e Pardo (2014). Especificamente neste trabalho, investigaram-se quais relações CST contribuem para a produção de bons sumários de atualização, com objetivo de, principalmente, identificar informação nova e tratar os desafios do cenário multidocumento, também presentes na SAA.

É importante ressaltar que, na CST, utiliza-se um grafo bi-partido, ou seja, somente é possível haver arestas (relações) entre vértices de sentenças de textos diferentes. Assim, essa teoria mostra-se como um recurso para auxiliar o tratamento dos problemas do cenário multidocumento ou intertextuais.

Ainda no âmbito da análise discursiva de um texto, Koch (2009) propõe que um texto coerente é composto por diversos segmentos textuais relacionados que, juntos, compõem o tema ou tópico principal do texto. Em geral, esses segmentos são formados por sequências sentencias agrupadas, que reportam um mesmo subtópico inserido no assunto geral do texto.

No Quadro 11, é apresentado um exemplo de texto do CSTNews, que foi manualmente segmentado e rotulado com o respectivo subtópico. Em cada segmento, é apresentado em negrito e entre os caracteres $<$ e $>$ um rótulo e um identificador ao respectivo subtópico associado. Por exemplo, para o primeiro segmento, foi atribuído o rótulo "acidente aéreo" e identificador igual a 1. Esse identificar ocorre novamente no terceiro segmento, no qual se pode perceber que, embora possua um rótulo diferente, aborda um subtópico muito similar ao do primeiro segmento.

Quadro 11 – Exemplo de segmentos de subtópicos em um texto.

Ao menos 17 pessoas morreram após a queda de um avião de passageiros na República Democrática do Congo. Segundo uma porta-voz da ONU, o avião, de fabricação russa, estava tentando aterrissar no aeroporto de Bukavu em meio a uma tempestade. A aeronave se chocou com uma montanha e caiu, em chamas, sobre uma floresta a 15 quilômetros de distância da pista do aeroporto. **<t LABEL="acidente aéreo" TOP="1">**
Acidentes aéreos são freqüentes no Congo, onde 51 companhias privadas operam com aviões antigos principalmente fabricados na antiga União Soviética. **<t LABEL="histórico" TOP="2">**
O avião acidentado, operado pela Air Traset, levava 14 passageiros e três tripulantes. Ele havia saído da cidade mineira de Lugushwa em direção a Bukavu, numa distância de 130 quilômetros. **<t LABEL="avião acidentado" TOP="1">**
Aviões são usados extensivamente para transporte na República Democrática do Congo, um vasto país no qual há poucas estradas pavimentadas. Em março, a União Européia proibiu quase todas as companhias aéreas do Congo de operar na Europa. Apenas uma manteve a permissão. Em junho, a Associação Internacional de Transporte Aéreo incluiu o Congo num grupo de vários países africanos que classificou como "uma vergonha" para o setor. **<t LABEL="histórico" TOP="2">**

Com a identificação desses segmentos textuais, é possível desenvolver métodos de SA que maximizem a quantidade de subtópicos que compõem o tema geral dos textos-fonte. Assim, o sumário gerado trará maior abrangência de conteúdo. Além disso, essa abordagem pode também ser utilizada para identificar o fluxo dos subtópicos, tais como: i) que ocorreram apenas nos textos conhecidos pelo leitor; ii) que são presentes apenas nos textos não conhecidos pelo leitor; e iii) que foram alterados. No âmbito da SAA, os itens (ii) e (iii) podem ser bons indicativos de conteúdo de atualização.

Um outro exemplo de teoria discursiva muito empregada na tarefa da SA, sobretudo no cenário monodocumento, é a Teoria da Estrutura Retórica (RST, do inglês *Rhetorical Structure Theory*) (MANN; THOMPSON, 1987). De forma distinta à CST, a RST foi desenvolvida para modelar as relações discursivas presentes em um único texto. Nessa teoria, o conteúdo textual é representado em uma árvore retórica, na qual os componentes externos indicam segmentos textuais e as relações entre esses segmentos são dispostas nos componentes internos da árvore. Cada nó da árvore é categorizado entre nuclear ou satélite, quando, respectivamente, apresentam informação considerada principal e secundária (ou complementar).

A seguir, será apresentado o conceito de tópicos latentes, que possui uma acepção diferente para o termo tópico que foi dissertado anteriormente. Assim, para padronização e remoção da ambiguidade, no decorrer deste trabalho, o termo tópico, quando utilizado

no significado de tema ou assunto principal de um texto, será substituído pela palavra tema e o termo subtópico será empregado para referenciar os fragmentos, ou subtemas, que compõem o tema ou assunto principal de um texto. A palavra tópico, então, fará referência ao conceito de tópicos latentes.

A identificação de tópicos latentes é outra possível abordagem para representação textual, bastante difundida em diversas áreas de pesquisa, inclusive na SAA. Um tópico latente, geralmente, é constituído por uma palavra ou expressão que foi identificada em uma coleção textual com respectivo valor de relevância para cada item (sentenças, textos, documentos, etc.) dessa coleção. Essa é uma abordagem muito utilizada para reduzir a dimensionalidade dos dados, ou seja, textos ou coleções de documentos são representados por conjuntos de palavras ou termos chaves.

Duas técnicas muito utilizadas para identificar tópicos latentes são a *Latent Semantic Analysis* (LSA) (LANDAUER; DUTNAIS, 1997) e a *Latent Dirichlet Allocation* (LDA) (BLEI; NG; JORDAN, 2003). A LSA é um modelo que utiliza decomposições matriciais para identificar quais palavras podem ser considerados termos importantes e qual a respectiva relevância desses termos para os documentos. Assim, após a extração de uma matriz M com dimensão igual $|V|x|D|$, em que V representa o tamanho do vocabulário (número de palavras distintas) e D a quantidade de documentos, tal que uma célula $M_{i,j}$ indica a frequência ou presença da palavra i no documento j, decompõe-se essa matriz no seguinte modelo $M = U\Sigma V^T$, tal que U represente os termos, V^T a matriz transposta com a relevância dos documentos, e Σ a matriz que indica quais os termos que compõem um tópico latente que são relevantes para um dado documento. A LDA emprega uma abordagem muito semelhante à LSA. Entretanto, faz uso de um modelo probabilístico para identificar os tópicos e emprega uma distribuição a priori para os termos e tópicos.

2.3.2 Córpus

Nesta seção, serão descritos os córpus que serviram como base para a realização de experimentos e a produção de novos recursos neste trabalho. Para a língua Portuguesa, utilizaram-se o córpus CSTNews e o *Priberam Compressive Summarization Corpus* (PCSC). Para a língua inglesa, foram utilizados os córpus disponibilizados nas competições de SAA da DUC 2007 e TAC 2008 e 2009.

O córpus CSTNews (ALEIXO; PARDO, 2008; CARDOSO *et al.*, 2011) possui 140 textos jornalísticos organizados por assunto em 50 coleções, com dois ou três textos cada. Os textos são notícias reportadas no Brasil e capturadas de 5 portais de notícias *online* (Folha de São Paulo, Estadão, O Globo, Gazeta do Povo e Jornal do Brasil). Cada coleção é classificada em uma de seis possíveis categorias (Ciência, Cotidiano, Dinheiro, Esportes, Mundo e Política).

No córpus CSTNews, cada texto-fonte possui um respectivo sumário monodocumento, que foi produzido por um humano. Além disso, para cada coleção, há 14 sumários multidocumentos, sendo que 12 foram produzidos por humanos, de forma que metade desses são sumários extrativos e os demais abstrativos, e os outros 2 sumários foram automaticamente gerados por meio do método CSTSumm (Castro Jorge, 2010). Além de sua vasta aplicação em investigações de SA para língua portuguesa, esse córpus possui inúmeros conhecimentos linguísticos anotados que contribuíram para o andamento deste trabalho, tais como: relações discursivas CST (RADEV, 2000) entre pares de textos-fonte de cada coleção; desambiguação de sentido de substantivos comuns (NóBREGA; PARDO, 2012) e verbos (Sobrevilla Cabezudo *et al.*, 2014); e segmentação de subtópicos (CARDOSO *et al.*, 2011).

O Córpus de Sumarização por Compressão do Priberam (PCSC, do inglês *Priberam Compressive Summarization Corpus*) (ALMEIDA *et al.*, 2014) é constituído por 801 textos jornalísticos que foram extraídos de 8 portais de notícias de Portugal (Diário de Notícias, Jornal de Notícias, Correio da Manhã, Rádio e Televisão de Portugal, TSF Rádio de Notícias, Jornal de Negócios, Record e O Jogo) e são relacionados à eventos ou fatos sobre desastres naturais, esporte, política, economia e outros que foram reportados em Portugal. Esse córpus é organizado por assunto em 80 coleções com 10 textos-fonte cada, exceto uma que possui 11, de forma que a metade dessas possuem notícias que foram reportadas entre os anos de 2010 e 2011 e a outra entre 2012 e 2013.

Em cada coleção de textos do PCSC, há dois sumários multidocumentos gerados semiautomaticamente por meio de uma abordagem de síntese compressiva e com não mais de 100 palavras. Em um primeiro momento, para cada coleção, um algoritmo foi aplicado para remover sentenças pouco relevantes dos textos-fonte. Posteriormente, humanos deveriam remover sentenças ou palavras do sumário resultante até que a taxa de compressão fosse atingida. Entretanto, no caso de remoção de partes de sentenças, a sentença resultante deveria ser gramaticalmente correta.

No córpus empregado na competição de SAA realizada na *Document Understand Conference* (DUC) de 2007, há 250 textos jornalísticos agrupados por tema em dez coleções. Em cada coleção, os textos são cronologicamente ordenados e organizados em três grupos, nomeadamente A, B e C, de forma que a ordenação dos textos é mantida. Dessa forma, por exemplo, todos os textos do grupo A foram publicados antes dos textos do grupo B e C. Em cada coleção, há 15 sumários abstrativos com não mais de 100 palavras que foram produzidos por humanos, sendo 5 para cada grupo, de forma que para os sumários dos grupos B e C, assumiu-se que o leitor tinha conhecimento das coleções A e B, respectivamente. Dessa forma, para cada grupo, há um sumário multidocumento tradicional (grupo A) e dois sumários de atualização (grupos B e C). No decorrer do texto, esse córpus será referenciado como DUC 2007.

Os córpus empregados nas competições da *Text Analysis Conference* (TAC) de 2008 e 2009 possuem estrutura similar ao da DUC 2007. Contudo, há apenas dois grupos de textos, A e B, em cada coleção. No córpus da TAC 2008, há 48 coleções de textos-fonte relacionados por assunto, nas quais há dois grupos, A e B, com 10 textos cada, de forma que àqueles disponíveis na coleção A possuem data de publicação inferior àqueles da coleção B. Para cada grupo, há 4 sumários multidocumentos e abstrativos produzidos por humanos, sendo que para o grupo B, os sumários são de atualização admitindo-se que o usuário teria conhecimento dos textos do grupo A.

O Córpus da TAC 2009 possui estrutura idêntica à apresentada no córpus da TAC 2008, com a mesma quantidade de textos e grupos por coleção, bem como a quantidade e restrições de sumários abstrativos humanos. Entretanto, foram disponibilizadas apenas 44 coleções de textos em vez de 48, como na TAC 2008.

Na Tabela 1, é apresentado um resumo comparativo entre as características dos córpus supracitados. Na primeira coluna, são listadas as características que foram comparadas. Na segunda e terceira colunas, respectivamente, são apresentados os valores analisados de cada córpus.

Tabela 1 – Resumo comparativo entre os córpus CSTNews, PCSC, DUC 2007, TAC 2008 e TAC 2009.

Características	**CSTNews**	**PCSC**	**DUC 2007**	**TAC**	
				2008	**2009**
Coleções	50	80	10	48	44
Textos-fonte	140	801	250	480	440
Textos/coleção	2–3	10–11	6–11	20	20
Sum./coleção	14	2	5-10	4	5
Síntese	extrativa e abstrativa	compressiva	abstrativa	abstrativa	abstrativa
Idioma	Português (BR)	Português (PT)	Inglês	Inglês	Inglês

2.3.3 Ferramentas

O CSTParser (MAZIERO; Castro Jorge; PARDO, 2014) é o atual estado da arte em identificação automática de relações discursivas segunda a CST (RADEV, 2000) para a língua portuguesa. Sua acurácia, aferida por uma abordagem de *ten-fold cross-validation*, é de 68%. O CSTParser utiliza apenas um subconjunto de 14 relações discursivas dentre as 28 propostas originalmente, que foram identificadas por Maziero, Castro Jorge e Pardo (2010) (vide Figura 3).

O CSTParser utiliza uma abordagem híbrida, com regras criadas manualmente e outras identificadas por meio de algoritmos de aprendizado de máquina. Para sua utili-

zação, é possível acessar uma versão *online*[1] ou instalar uma versão *offline*[2] em algum computar que disponha do interpretador Perl®.

O analisador sintático PALAVRAS (BICK, 2000), que foi desenvolvido por meio de regras criadas manualmente, possui diversas funcionalidades para identificar informações sintáticas diversas, como árvores de constituintes e de dependência, e é considerado um dos melhores *parsers* para a língua portuguesa. Segundo o autor, essa ferramenta possui acurácia de aproximadamente 98% para a língua Portuguesa. Entretanto, é importante ressaltar que durante o uso para processar textos jornalísticos, seu desempenho aparentemente mostra-se inferior ao valor reportado. Nesse cenário, também é importante ressaltar que a ferramentas PALAVRAS, frequentemente, é atualizada e que seu criador recebe/responde sugestões advindas dos membros do grupo de pesquisa NILC, do qual o autor deste trabalho faz parte. Assim, possíveis erros são corrigidos, contribuindo para uma melhor qualidade na identificação de informações sintáticas.

Outra opção para identificação automática de informação morfossintática dos textos-fonte é o MXPOST, que faz uso de um algoritmo independente de língua (após treinamento do modelo). Seu funcionamento baseia-se em um modelo probabilístico pautado no cálculo da máxima entropia (RATNAPARKHI, 1986). Para o Português do Brasil, Aires (2000), em experimentos com aproximadamente 100 mil palavras e um conjunto de 17 etiquetas morfossintáticas, reportaram uma performance de 97% de acurácia para o MXPOST.

A segmentação de subtópicos é a tarefa de identificar segmentos textuais coesos que abordam um mesmo subtópico. Umas das ferramentas mais utilizadas para automatizar essa tarefa é a TextTiling (HEARST, 1997), cujo funcionamento baseia-se na identificação de troca de vocabulário entre sentenças. Assim, quando o vocabulário entre duas sentenças é considerado suficientemente distinto, uma quebra ou segmentação ocorre entre essas. Por outro lado, se não há diferença significante de vocabulário entre duas sentenças, essas são agrupadas e consideradas de um mesmo subtópico.

Cardoso, Taboada e Pardo (2013) realizaram diversos experimentos de segmentação de subtópicos para língua portuguesa por meio do córpus CSTNews e propuseram um método de segmentação baseado em informação discursiva, a RST (MANN; THOMPSON, 1987). Em experimentos reportados por esses autores, a TextTiling (adaptada para a língua portuguesa) obteve cobertura de 0,405 e precisão de 0,773. Já a melhor configuração do método proposto por Cardoso, Taboada e Pardo (2013), apresentou cobertura de 0,908 e precisão igual a 0,353.

[1] Disponível em http://www.nilc.icmc.usp.br/CSTParser/

[2] Disponível em http://www.icmc.usp.br/pessoas/taspardo/sucinto/files/CSTParser%20-%20standalone.rar

2.4 Considerações finais

Neste capítulo, foram apresentados os recursos, ferramentas (técnicas e teóricas) que viabilizaram o desenvolvimento deste trabalho. Além disso, dissertou-se também sobre os principais conceitos relacionados à SA, como as classificações utilizadas na literatura, para distinguir as diferentes abordagens, e os métodos comumente empregados na avaliação de métodos de SAA.

Dada as definições para classificação de sumários que foram apresentadas na Seção 2.1, os sumários produzidos nos experimentos realizados neste trabalho podem ser considerados informativos e com foco no usuário (admitindo-se que SAA é uma tarefa naturalmente focada no usuário pois se considera alguns textos conhecidos pelo leitor). Além disso, foram investigados os métodos de Síntese Extrativo e Compressivo.

Com relação aos recursos, foram introduzidos cinco córpus distintos com características particulares e com textos e sumários compostos em dois idiomas diferentes. Para a língua Portuguesa, foram apresentados os córpus CSTNews (ALEIXO; PARDO, 2008; CARDOSO *et al.*, 2011), que é bastante empregado em investigações de SA Mono e Multidocumento nesse idioma, e o PCSC (ALMEIDA *et al.*, 2014), em que os respectivos sumários foram produzidos por meio da Síntese Compressiva. Para a língua inglesa, foram descritos os córpus disponibilizados nas competições de SAA da DUC de 2007 e TAC de 2008 e 2009. Esses conjuntos de dados foram utilizados para a realização de experimentos para avaliar os métodos de SAA desenvolvidos e para compilação de outros recursos que auxiliaram esta investigação.

CAPÍTULO

3

REVISÃO DA LITERATURA

Neste capítulo, uma revisão da literatura relacionada às tarefas de Sumarização Automática de Atualização e Compressão de Sentenças será apresentada. Nas Seções 3.1 e 3.2, são apresentadas os principais trabalhos de SAA e Compressão Sentencial, respectivamente. Com o objetivo de descrever o avanço desses dois campos de pesquisa, as referências foram organizadas em ordem cronológica.

3.1 Sumarização Automática de Atualização

Nesta seção, para uma melhor organização, os trabalhos foram agrupados em 3 categorias, conforme as seguintes abordagens: (i) sensível ao contexto; (ii) análise de tópicos textuais; e (iii) otimização.

Os métodos sensíveis ao contexto, que serão listados na Seção 3.1.1, são aqueles que fazem uso de técnicas tradicionais de SA e identificam atualização por meio da remoção de sentenças redundantes em relação às sentenças dos textos conhecidos pelo leitor. Na Seção 3.1.2, serão apresentados os trabalhos que fazem análise de tópicos textuais, tais como aqueles que identificam tópicos latentes. Por fim, na Seção 3.1.3, disserta-se sobre trabalhos que fazem uso de métodos de otimização. No final deste capítulo, um resumo e discussão sobre os trabalhos listados serão apresentadas na Seção 3.1.4.

Todos os métodos de SAA que serão apresentados neste capítulo avaliaram os resultados por meio de um ou mais córpus disponíveis pela DUC 2007 e edições da TAC de 2008 a 2011. Assim, uma visão geral desses eventos será apresentada a seguir.

A Sumarização Automática de Atualização (SAA) ocorreu como foco principal nas tarefas da *Document Understanding Conference* (DUC) de 2007 e *Text Analysis Conference* (TAC)[1] de 2008, que são conferências semelhantes às competições, na qual os

[1] Novo título atribuído aos eventos da DUC

métodos submetidos são avaliados sobre o mesmo conjunto de dados e devem seguir as regras e restrições previamente estipuladas. Nas edições posteriores, exceto 2013 e 2014, a SAA também esteve presente, mas não como foco principal. A quantidade de trabalhos submetidos, relacionados à SAA, para a DUC 2007 e TAC de 2008 a 2011 é apresentada na Tabela 2.

Duc 2007	**TAC 2008**	**TAC 2009**	**TAC 2010**	**TAC 2011**
19	31	23	23	25

Tabela 2 – Quantidade de trabalhos submetidos aos eventos DUC e TAC entre 2007 a 2011.

Na DUC de 2007, a tarefa da SAA foi definida pela produção de três sumários multidocumento a partir de três coleções textuais sobre um mesmo assunto e cronologicamente ordenadas, de forma que os dois últimos deveriam ser de atualização, admitindo que as coleções anteriores fossem conhecidas pelo leitor. Por exemplo, o terceiro sumário deveria atualizar o conhecimento do leitor sobre o que já foi lido na primeira e segunda coleções. Posteriormente, na TAC de 2008, a tarefa de SAA foi definida de forma semelhante, porém, reduziu-se o número de coleções textuais de três para dois. Assim, dois sumários deveriam ser produzidos pelos métodos, um multidocumento tradicional e um de atualização, admitindo que o leitor conheça a primeira coleção (DANG; OWCZARZAK, 2008).

Nas posteriores edições da TAC, de 2009 a 2011, sumários de atualização também deveriam ser produzidos. Porém, novas características foram adicionadas à tarefa, de forma que o foco principal foi estendido ou alterado. Por exemplo, na TAC de 2009, a tarefa de SAA foi executada de forma semelhante à edição anterior, porém com o direcionamento de que os sumários deveriam satisfazer uma *query* (consulta) previamente estipulada (DANG; OWCZARZAK, 2009). Essa extensão parte do cenário em que um usuário deseja ler mais conteúdo sobre um determinado assunto, do qual ele já possui algum conhecimento prévio, e, para tanto, utiliza-se de ferramentas de busca, como o Google Search®e Bing Search®, por meio de consultas para encontrar essas informações.

Em 2010 e 2011, com objetivo de incentivar o uso de conhecimento linguístico nos métodos de SA, o foco dos eventos foi a Sumarização Guiada, na qual, para cada tipo de assunto disponível no córpus de avaliação, o sumário deveria ser composto de segmentos de informação específicas, referenciados como aspectos (OWCZARZAK; DANG, 2010; OWCZARZAK; DANG, 2011). Por meio dos aspectos, identificam-se componentes de conteúdo relacionados ao assunto principal de um texto, tais como: *WHAT*, o que aconteceu ou foi relatado; *WHEN*, quando aconteceu; *WHERE*, onde ocorreu; *WHY*: razões ou motivos que fez com que o evento ou fato acontecesse; etc. Para cada tipo de assunto, eram sugeridos os aspectos que deveriam ser identificados e inseridos no sumário.

Nas próximas seções, os trabalhos selecionados durante a revisão da literatura

serão apresentados.

3.1.1 Trabalhos baseados em abordagem sensível ao contexto

Reeve e Han (2007) assumem que um sumário deve apresentar uma distribuição de termos (ou unidades) semelhante à distribuição presente nos textos-fonte. Dessa forma, espera-se que o sumário produzido seja constituído pela maioria das informações presentes nos textos-fonte de maneira que aquelas consideradas mais relevantes nos textos-fonte também sejam no sumário. É importante ressaltar que os autores não apresentam o método utilizado para identificar os termos ou unidades dos textos.

Com a hipótese supracitada, Reeve e Han (2007) propõem o método FreqDistUpdate, que é composto por duas principais fases de processamento. Na primeira etapa, para cada coleção de textos (recente e anterior), é calculada a distribuição de frequência dos termos presentes nos textos. Já na segunda fase, em um processo iterativo, que é executado enquanto a taxa de compressão não tenha isso atingida, as sentenças dos textos-fonte são ranqueadas e a primeira dessa ordenação é inserida no sumário. A pontuação para uma sentença é alta se sua inserção no sumário mantém a distribuição de termos do sumário semelhante à distribuição dos textos-fonte.

No FreqDistUpdate, com o objetivo de identificar as informações com atualização, a frequência dos termos presentes nos textos recentes é inicializada com a respectiva frequência nos textos conhecidos pelo leitor. Em outras palavras, se um item lexical t ocorreu 5 e 10 vezes, respectivamente, nos textos recentes e anteriores, sua frequência nos textos não conhecidos pelo leitor será igual a 15. Com isso, espera-se que sentenças compostas por palavras pouco frequentes sugiram informação nova, pois são constituídas por unidades que pouco ocorreram anteriormente. Além disso, para evitar sentenças redundantes no sumário, os autores descartam as sentenças muito similares às presentes já selecionadas.

O método FreqDistUpdate obteve resultados medianos na avaliação da DUC de 2007. Entretanto, os autores salientam que essa abordagem emprega uma estratégia simples para seleção das sentenças que poderia ser aprimorada com uma possível etapa de simplificação textual. Por exemplo, em alguns casos, sentenças consideradas relevantes eram descartadas, pois a respectiva inserção no sumário extrapolaria a taxa de compressão (100 palavras) e outras sentenças menores e menos relevantes eram utilizadas.

Boudin, El-Bèze e Moreno (2008) salientam que um grande problema da SA multidocumento, também compartilhado pela SAA, é a necessidade de processar todos os textos-fonte em conjunto para a produção de novos sumários. Os autores referenciam esse problema como processamento em "lote". Nesse contexto, Boudin, El-Bèze e Moreno (2008) propõem maneiras de SAA iterativas e escaláveis por meio da abordagem

de Relevância Marginal Máxima (CARBONELL; GOLDSTEIN, 1998), que consiste no ranqueamento de informação (ou sentenças) com base em dois pontos de referência, de forma que a informação seja similar com um e dissimilar com outro.

Para a produção de sumários, a cada iteração do algoritmo, todas as sentenças que ainda não foram selecionadas são ranqueadas e a sentença mais bem pontuada é inserida no sumário. Os autores propuseram algumas variações para esse processo, mas a melhor configuração foi dada pelo ranqueamento efetuado por meio da similaridade da sentença ao tópico dos textos multiplicada pelo respectivo maior valor de redundância em relação às sentenças dos textos conhecidos pelo leitor e aquelas que já foram introduzidas no sumário. Em outras palavras, a sentença que é selecionada para o sumário é aquela mais similar com o tópico textual (disponível no córpus usado para avaliação) e mais distinta das sentenças do histórico (textos conhecidos e atuais sentenças do sumário).

Por meio do córpus da DUC 2007, Boudin, El-Bèze e Moreno (2008) reportam que o método proposto obteve 0,363 de R-1, 0,102 de R-2 e 0,138 de R-SU4 de desempenho. Com esses valores, o método não foi superior aos melhores trabalhos submetidos à edição, porém, foi superior a média dos sistemas avaliados na edição. Como enfatizado pelos autores, essa abordagem pode ser empregada em um contexto *online*, no qual novos textos são processados conforme são publicados, mas ainda possui a necessidade de refazer o ranqueamento das sentenças a cada iteração.

Wenjie *et al.* (2008) apresentam um método de SAA por meio de reforço positivo e negativo baseado em grafo. Na proposta de Wenjie *et al.* (2008), chamada PNR^2 (Ranqueamento com Reforço Positivo e Negativo, do inglês *Ranking with Positive and Negative Reinforcement*), cada sentença é representada como um vértice em um grafo e as arestas identificam relações (positivas ou negativas) entre sentenças. Sentenças recebem reforços positivos quando se relacionam (são similares) com sentenças de uma mesma coleção e reforço negativo quando se relacionam com sentenças de outra coleção. Em outras palavras, o reforço positivo identifica conteúdo relevante dentro de um mesmo grupo de textos e, por outro lado, o reforço negativo visa diminuir a redundância entre os textos recentes com os anteriores o que, consequentemente, auxilia a identificação de informação nova.

O método PNR^2, assim como o PageRank (BRIN; PAGE, 1998), assume a intuição de que sentenças relevantes são relacionadas com outras sentenças relevantes. Entretanto, os autores salientam que o algoritmo PageRank emprega somente o reforço positivo, ou seja, no âmbito da SAA, não diferencia informação conhecida de informação nova. Na proposta dos autores, duas sentenças eram consideradas relacionadas se a similaridade do Cosseno (SALTON; WONG; YANG, 1975) entre as duas era maior do que um limiar de 0,81.

Em experimentos realizados sobre o córpus da DUC 2007, o PNR^2 obteve resultados de R-1, R-2 e R-SU4 de, respectivamente, 0,361, 0,089 e 0,129. Com isso, o método

mostrou-se superior ao resultado médio dos trabalhos submetidos à DUC 2007, mas não superior ao primeiro colocado do evento. Nesse cenário, embora o PNR^2 faça uso de um método refinado para ponderar relevância de conteúdo, as informações com atualização ainda são selecionadas por meio da não redundância com os textos conhecidos pelo leitor. De fato, essa abordagem mostra-se mais promissora do que somente o uso de métricas de similaridade, mas a identificação de informação com atualização ainda não é tão bem explorada.

Bawakid e Oussalah (2008) propõem um método de SAA baseado no ranqueamento de sentenças por meio de características textuais de diferentes naturezas, tais como atributos posicionais (posição da sentença no texto), número de entidades nomeadas e similaridade semântica entre sentenças. (como referenciado pelos autores, mas que, basicamente, trata-se de uma análise de padrões sintáticos). A proposta dos autores pauta-se no uso desses atributos para ranquear as sentenças com objetivo de produzir sumários com mais informações relevantes, que satisfaçam e/ou sejam similares a uma *query* e tópico definidos previamente (dados disponíveis no córpus da TAC 2008, no qual os autores realizaram experimentos). Além disso, é importante ressaltar que os autores não apresentam mecanismos para tratar efetivamente a necessidade de identificar conteúdo com atualização ou novidade.

Após uma etapa de pré-processamento dos textos (identificação de sentenças; tokenização; remoção de caracteres ou marcações desnecessárias; identificação de entidades nomeadas e aplicação de parser sintático), os atributos para cada sentença dos textos-fonte são computados. Posteriormente, as sentenças são ranqueadas para cada texto-fonte individualmente por meio de um equacionamento que considera os seguintes pontos: similaridade semântica da sentença em relação à *query* e ao tópico; a quantidade de sentenças que possuem pontuação superior a um limiar previamente estipulado; quantidade de entidades nomeadas presentes na sentença e em seu respectivo texto e a quantidade de sentença do texto-fonte.

Para computar a similaridade semântica, os autores consideram os pares de substantivos com adjetivos, verbos com advérbios e substantivos ou verbos com quantificadores linguísticos (os autores exemplificam com itens lexicais que enfatizam a importância ou irrelevância de um substantivo ou verbos, tais como as palavras *very* ou *less*) presentes nas sentenças, de forma que adjetivos e advérbios são expandidos por meio dos respectivos sinônimos presentes na WordNet de Princeton (FELLBAUM, 1998). Além disso, os autores também consideram a similaridade entre itens lexicais isolados, para qual reportam três possíveis algoritmos da literatura. Assim, a similaridade entre uma sentença e uma *query* é alta quando ambas são compostas por pares de itens lexicais semelhantes (ex.: substantivos com adjetivos próximos ou quantificadores de igual direcionamento).

Bawakid e Oussalah (2008) reportam experimentos, com algumas variações no cál-

culo da similaridade entre itens lexicais isolados, sobre o córpus da TAC de 2008, nos quais obtiveram R-1, R-2 e R-SU4, respectivamente, iguais a 0,341, 0,080, e 0,117. Com esses valores, os autores não superam os melhores métodos da edição. Entretanto, ressalta-se que a proposta de Bawakid e Oussalah (2008) apresenta alto valor de cobertura, o que sugere que o método é factível para identificar informação relevante, porém é pouco eficiente para remover conteúdo redundante (os autores afirmam que essa etapa do processamento não foi concluída durante os experimentos).

Katragadda, Pingali e Varma (2009) apresentam uma abordagem de detecção de relevância por meio da posição sentencial para a SAA. Primeiramente, os autores investigaram em que posições no corpo textual as sentenças mais relevantes ocorrem por meio da análise do córpus da DUC 2007. Para tanto, os autores utilizaram as Unidades Conceituais (empregadas pela métrica da Pirâmide (NENKOVA; PASSONNEAU, 2004) e disponíveis no córpus utilizado) para contabilizar a relevância das sentenças. Posteriormente, para cada posição textual, os valores aferidos para cada respectiva sentença foram somados, normalizados e identificados como a relevância para aquela posição sentencial. Por exemplo, a relevância da posição 1 (primeira sentença do texto) foi contabilizada pela média da relevância de todas as sentenças que ocorreram na primeira posição dos textos do córpus. É importante ressaltar que, dada a discrepância no tamanho dos textos, os autores classificaram-nos entre pequenos ou grandes e normalizaram o resultado de forma distinta para cada categoria.

Com a análise supracitada, os autores identificaram um ranqueamento ótimo de relevância por posição sentencial (OPP, do inglês *Optimal Position Policy*). Com esse ranqueamento, os autores propuseram um método de SAA que seleciona as sentenças conforme o ranqueamento OPP até que a taxa de compressão seja atingida. Em experimentos por meio do córpus da TAC 2008, os autores reportaram 0,091 de R-2, 0,124 de R-SU4 e 0,350 de Pirâmide. Com esses valores, a proposta de Katragadda, Pingali e Varma (2009) foi superior ao *baseline* empregado na TAC 2008, que produz sumários de atualização por meio das 100 primeiras palavras do texto-fonte mais recente. Os autores enfatizam a simplicidade do método e, como proposto por eles, trata-se de uma abordagem que deve ser usada como um *baseline* mais refinado para a SAA.

Varma *et al.* (2009) propõem o uso do algoritmo *Support Vector Regression* (SVR) para identificar os melhores atributos sentenciais para a produção de sumários de atualização. Os autores utilizam atributos posicionais e o fator de novidade (definido pelos autores), cujo objetivo é ponderar o nível de conteúdo de atualização de uma sentença. Na abordagem de Varma *et al.* (2009), as sentenças mais bem ranqueadas são inseridas no sumário, porém, respeitando a ordem em que ocorrem nos textos-fonte, visando evitar problemas de ordenação sentencial, que podem diminuir a facilidade de leitura do sumário. Além disso, para produzir sumários mais informativos, os autores desconsideram sentenças

candidatas (que poderiam ser inseridas no sumário) consideradas redundantes em relação às sentenças já inseridas no sumário. Entretanto, o método utilizado para essa tarefa não é especificada no artigo.

Varma *et al.* (2009) utilizaram dois atributos posicionais, SL1 e SL2. O primeiro assume que o conteúdo mais relevante de um texto se encontra nas três primeiras sentenças dos textos-fonte. Assim, admitindo-se que um texto possui no máximo 1000 sentenças, uma sentença é pontuada com $1 - \frac{n}{1000}$, caso a sentença seja uma das três primeiras do texto, ou $\frac{n}{1000}$, caso contrário. Dessa forma, as três primeiras sentenças recebem valores bem maiores que as demais. O segundo parâmetro posicional, SL2, apenas utiliza a posição sentencial para o treinamento do algoritmo SVR.

O Fator de Novidade (NF), como definido pelos autores, visa identificar conteúdo de atualização partindo do princípio de que uma sentença possui atualização se é constituída por itens lexicais mais frequentes nos textos desconhecidos do que nos conhecidos pelo autor. Para tanto, por meio da equação a seguir, cada sentença é valorada por meio do somatório do NF de cada palavra (removendo-se *stopwords*) que a constitui, normalizado pelo tamanho da sentença, onde: $|nd_t|$ e $|pd_t|$ indicam, respectivamente, a quantidade de textos desconhecidos e conhecidos pelo autor em que w ocorre, e $|D|$ a quantidade de textos-fonte.

$$nf(w) = \frac{|nd_t|}{|pd_t| + |D|} \tag{3.1}$$

Para treinamento do modelo, os autores, primeiramente, estimaram a relevância de cada sentença no conjunto de treinamento por meio da similaridade da sentença com sumários modelos (produzidos por humanos) dos respectivos textos-fonte por meio da ROUGE-2. Após essa etapa, os atributos das sentenças foram extraídos e, em conjunto com a respectiva relevância da sentença, foram submetidos ao algoritmo SVR, para identificar a proeminência de cada atributo. Dessa forma, pesos foram atribuídos aos atributos e, posteriormente, utilizados para ranquear as sentenças conforme suas características.

Varma *et al.* (2009) reportam experimentos com duas configurações de sua proposta por meio do conjunto de dados da TAC 2009. Na primeira, o treinamento do modelo ocorreu com os dados da TAC 2008 por meio dos atributos SL1 (posição sentencial, com ênfase nas três primeiras sentenças dos textos-fonte) e fator de novidade. Essa configuração obteve, na produção de sumários de atualização, R-2 igual a 0,101, R-SU4 de 0,138, Pirâmide e Responsividade, respectivamente, iguais a 0,307 (melhor resultado na TAC de 2009) e 4,614. Já na segunda configuração, os autores utilizaram os dados da DUC 2007 (o *dataset* para SAA) e os atributos SL2 (posição da sentencial) e fator de novidade. Nessa disposição, a avaliação do método apresentou R-2 de 0,095, R-SU4 igual a 0,136, Pirâmide e Responsividade, respectivamente, iguais a 0,299 e 4,568.

Após os experimentos supracitados, Varma *et al.* (2009) investigaram mais alguns atributos, que visavam produzir sumários mais direcionados ao tópico indicado nos textos-fonte e com ênfase maior em conteúdo novo. Os resultados não se mostraram superiores, exceto para uma pequena melhoria em R-SU4 (com valor 0,140), que ocorreu quando também se considerou o número de palavras novas (não presentes nos textos conhecidos pelo leitor) das sentenças dos textos-fonte por meio da fórmula a seguir, onde: s é uma sentença, w um item lexical, $F_{clusA}(w)$ é 0 se w ocorreu em algum texto conhecido pelo leitor e sua frequência nos textos novos, caso contrário, e $|s|$ o número de itens lexicais da sentença.

$$Score(s) = \frac{\sum_{w \in s} F_{clusA}(w)}{|s|} \tag{3.2}$$

A proposta de Varma *et al.* (2009), embora simples com relação aos atributos utilizados, apresenta bons resultados e se mostra bastante eficiente, uma vez que o modelo tenha sido treinado. Os resultados reportados pelos autores indicam a relevância de atributos posicionais, também presente em outros trabalhos da literatura (KATRAGADDA; PINGALI; VARMA, 2009; OUYANG *et al.*, 2010), e a melhor qualidade dos sumários quando conteúdo novo é enfatizado nos sumários. Além disso, as investigações posteriores dos autores, no objetivo de melhorar os resultados, por meio da análise de palavras novas nas sentenças dos textos-fonte mais recentes poderia ser incrementada, por exemplo, considerando-se palavras sinônimas e não somente itens lexicais isolados.

Ouyang *et al.* (2010) apresentam um método de sumarização baseado em características posicionais. Por característica posicional, nesse contexto, entende-se a posição em que a sentença ou item lexical ocorreu no texto. Assim, nessa abordagem, a relevância de uma sentença é mensurada por meio da respectiva posição no texto e pelo somatório das pontuações posicionais atribuídas aos itens lexicais que a compõem. Para pontuar os itens lexicais, os autores propõem quatro possíveis métricas. Três dessas métricas diminuem (de formas distintas) o peso dos itens lexicais conforme a frequência no texto, de forma que as primeiras ocorrências tenham pesos maiores dos que as demais, e a outra pontua apenas a primeira ocorrência do item lexical. Assim, em um processo iterativo, as sentenças mais bem ranqueadas são inseridas no sumário até que a taxa de compressão seja atingida.

Ouyang *et al.* (2010) fazem experimentos com nove variações de combinações das métricas supracitadas por meio do córpus da TAC 2008 e 2009. Na primeira, as sentenças são ranqueadas conforme a frequência dos itens lexicais que as compõem. Já para as demais configurações, as sentenças são ranqueadas por meio de uma das quatro métricas de pontuação de itens lexicais supracitadas, considerando ou não a posição da sentença.

No primeiro conjunto de avaliação, os autores reportaram que a configuração mais bem avaliada do algoritmo proposto, que não fez uso da posição sentencial durante o ranqueamento, obteve 0,375 de R-1 e 0,394 para R-2. No segundo cenário de avaliação, para

R-1 de 0,373, a configuração mais bem ranqueada do algoritmo foi análoga ao primeiro experimento. Entretanto, para R-2 com 0,095, o maior resultado ocorreu por meio do algoritmo que fez uso da posição sentencial.

Com os valores supracitados, a proposta de Ouyang *et al.* (2010) mostra-se superior à média dos sistemas submetidos às duas edições da TAC (2008 e 2009). Além disso, trata-se de uma abordagem muito simples, que não faz uso de métodos para identificar conteúdo com atualização. A utilização dessa abordagem aliada à proposta de Reeve e Han (2007), que visa identificar atualização por meio da frequência dos itens lexicais presentes nos textos conhecidos e não conhecidos pelo leitor, poderia atingir resultados mais satisfatórios.

3.1.2 Trabalhos de análise de tópico

Os métodos de SAA apresentados nesta seção empregam técnicas de identificação e análise de tópicos textuais para selecionar o conteúdo mais adequado para o sumário e, eventualmente, remover informação redundante. Dessa forma, embora alguns métodos aqui apresentados fazem uso da abordagem Sensível ao Contexto, que foi introduzida na seção anterior, optou-se por organiza-los em uma seção específica, pois o mecanismo principal para a produção de sumários desses métodos é diferentes dos trabalhos listados na Seção 3.1.1.

Steinberger e Ježek (2009) propõem um método de SAA baseado nas distinções de tópicos latentes (geralmente, constituídos por palavras isoladas e consideradas mais representativas) entre os textos recentes e os já conhecidos pelo leitor. Para tanto, os autores fazem uso da Análise Semântica Latente (LSA, do inglês, *Latent Semantic Analysis*) (LANDAUER; DUTNAIS, 1997) para identificar os tópicos presentes em uma coleção de textos ou sentenças e, além disso, para ponderar os tópicos mais proeminentes para cada sentença ou texto.

Na etapa de pré-processamento dos textos-fonte, a LSA é aplicada nos textos conhecidos e não conhecidos pelo leitor com objetivo de identificar os tópicos latentes presentes em cada cenário. Após essa análise, o nível de atualização dos tópicos latentes dos textos recentes é calculado por meio da diferença entre a respectiva proeminência nos textos recentes e a redundância (tópicos similares) com tópicos latentes presentes nos textos anteriores. Para a produção do sumário, a cada iteração do algoritmo, a sentença mais correlacionada com o tópico latente mais bem pontuado é selecionada para o sumário e a pontuação dos tópicos latentes é recalculada. Esse processo é executado até a taxa de compressão ser atingida.

Em experimentos realizados por Steinberger e Ježek (2009), por meio do córpus da TAC 2008, o método proposto obteve resultados satisfatórios, com valores aferidos

de Pirâmide, Qualidade Linguística, Responsividade, R-2, R-SU4 e BE, respectivamente, iguais a 0,28, 2,83, 2,29, 0,08, 0,12 e 0,05. Com esses valores, o método apresentando mostrou-se superior ao *baseline* e à média de pontuação dos sistemas participantes da edição. Por outro lado, não obteve resultados próximos ao método com maior pontuação. Entretanto, é importante ressaltar que a abordagem proposta faz uso de uma técnica escalável e independente de língua. Além disso, pode-se considerar que a proposta de analisar a distinção de tópicos entre os textos conhecidos e não conhecidos pelo leitor é promissora, mas deve ser mais explorada.

Huang e He (2010) propõem a identificação da correlações entre os tópicos latentes presentes nos textos conhecidos e não conhecidos pelo leitor para a SAA. Para tanto, os autores utilizam o algoritmo *Latent Dirichlet Allocation* (LDA) (BLEI; NG; JORDAN, 2003) para identificar os tópicos latentes e os classificam em uma das quatro categorias: (i) emergentes, que estão presentes somente nos textos recentes (não conhecidos pelo leitor); (ii) ativos, que estão presentes em todos os textos, mas com maior ênfase nos textos recentes; (iii) não ativos; que estão presentes em todos os textos, mas com pouco relevância nos textos recentes; e (iv) extintos, que estão presentes somente nos textos anteriores.

Posteriormente à análise de identificação de classificação dos tópicos latentes conforme as categorias supracitadas, somente as sentenças relacionadas aos tópicos latentes classificados como emergentes (i) ou ativos (ii) são considerados para o sumário. Eventualmente, caso a taxa de compressão não tenha sido atingida, sentenças dos tópicos não ativos (iii) e extintos (iv), nessa ordem, também são consideradas para o sumário. As sentenças são ranqueadas conforme a respectiva relação com os tópicos latentes, e esses, por sua vez, são ponderados conforme a frequência de ocorrência nos textos.

Os autores avaliaram o método proposto no conjunto de dados da TAC 2008 e de 2009. No primeiro conjunto, os valores aferidos de R-1, R-2 e R-SU4 foram, respectivamente, de 0,367, 0,097 e 0,131. Já na segunda avaliação, os valores de R-1, R-2 e R-SU4 foram, respectivamente, de 0,368, 0,097 e 0,1359. Nesse cenário, pode-se observar que o método de Huang e He (2010) obteve resultado superior à proposta de Steinberger e Ježek (2009), que também realiza a análise de tópico entre os textos. Provavelmente, isso ocorre, pois Huang e He (2010) analisa de forma mais refinada, com relação ao trabalho de Steinberger e Ježek (2009), o fluxo dos tópicos entre os textos conhecidos e não conhecidos.

Wang e Li (2010) propõem o uso de agrupamento hierárquico de sentenças com objetivo de detectar atualização e evitar o processamento em lote dos textos-fonte, ressaltado com um dos principais problemas da SA (BOUDIN; EL-BÈZE; MORENO, 2008). Nessa abordagem, as sentenças são agrupadas por meio da similaridade de conteúdo ou tópico que abordam de forma a representar uma taxonomia de conteúdo. Assim, quando

uma sentença é inserida em um novo ramo ou nó dessa estrutura, tem-se a identificação de uma atualização, pois o conteúdo dessa sentença não foi encontrado anteriormente, ou seja, não houve grupos com os quais essa nova sentença tenha sido similar.

Em um processo iterativo de agrupamento, para cada nova sentença analisada, o algoritmo computa sua similaridade com as demais sentenças já agrupadas. Assim, caso um grupo de sentenças similares seja encontrado, a sentença é aglutinada, caso contrário, um novo ramo ou grupo é criado para alocar a sentença. Ao final desse processo, o algoritmo seleciona os grupos mais relevantes conforme a taxa de compressão e seleciona as sentenças mais representativas de cada grupo para o sumário.

Wang e Li (2010) realizaram experimentos por meio do córpus da TAC 2008 e obtiveram valores de R-1 e R-2, respectivamente, iguais a 0,371 e 0,093. Além disso, os autores verificaram outros algoritmos, como agrupamento baseado em centroides, que correspondem aos itens (sentenças) mais representativos de um grupo, uma abordagem baseada em identificação de tópicos por meio da LSA, mas todas se mostraram inferiores ao algoritmo proposto. Os autores salientam que, além do resultado satisfatório, a proposta é computacionalmente eficiente e pode ser usada para processamento *online*, de forma que novos textos podem ser processados conforme sejam publicados.

Delort e Alfonseca (2012) apresentam o método DualSum, cuja abordagem baseia-se em um modelo probabilístico de tópicos, que, de forma semelhante ao trabalho de Reeve e Han (2007), parte do princípio que a distribuição de conteúdo textual presente no sumário deve ser semelhante à distribuição de informação presente nos textos-fonte. Especificamente no DualSum, essa distribuição é estipulada por uma distribuição de categorias de tópicos latentes previamente estipulada. Os autores enfatizam que a utilização dessa abordagem diminui a redundância e produz sumários com maior abrangência de conteúdo.

No DualSum, cada texto é representando por uma *bag-of-words* e cada palavra é associada a um tópico latente. Posteriormente, os tópicos latentes, identificados por meio da LDA (BLEI; NG; JORDAN, 2003), são ponderados em relação às seguintes quatro categorias conforme a direcionalidade da relevância: (i) relevância para o texto; (ii) relevância para a coleção de textos conhecida pelo leitor; (iii) relevância para a coleção não conhecida pelo leitor; e (iv) relevância geral. É importante ressaltar que, para representar corretamente o modelo de atualização, tópicos latentes da classe (iii) não são considerados possíveis na coleção de textos conhecida pelo leitor. Ou seja, tópicos dessa categoria podem somente serem identificados a partir dos textos mais recentes, que não são conhecidos pelo leitor.

No DualSum, um sumário é produzido de forma que sua respectiva distribuição de tópicos latentes se aproxime de uma distribuição previamente dada. Evidentemente, no contexto da SAA, espera-se que os sumários produzidos componham-se de sentenças

mais relevantes para os tópicos latentes da categoria (iii), que correspondem aos tópicos de atualização. Entretanto, uma combinação dessas categorias pode ser utilizada. Para determinar a melhor configuração dessa distribuição, Delort e Alfonseca (2012) utilizaram o córpus da TAC 2008 e 2009 para definir os parâmetros de distribuição por meio da análise das distribuições dos textos-fonte e respectivos sumários de referência.

Na etapa de geração do sumário, as classes dos tópicos latentes de cada sentença é computada em uma função de custo, que representa uma distribuição desejada. Assim, por meio de um algoritmo guloso e não determinístico, um subgrupo de sentenças que maximize o valor dessa função é selecionado para o sumário. Em experimentos realizados nos córpus da TAC 2011, os autores obtiveram R-1 de 0,357, R-2 de 0,092 e R-SU4 de 0,128. Com esses resultados, o DualSum apresentou-se superior, respectivamente, ao primeiro, terceiro e segundo colocados na TAC de 2011.

Li *et al.* (2012) apresentam uma abordagem de SAA pautada na análise do comportamento dinâmico dos tópicos (uma ou mais palavras representativas) presentes nos textos conhecidos e não conhecidos pelo leitor. Para tanto, cada tópico identificado é classificado entre uma das seguintes categorias: (i) geral, que são tópicos presentes nos textos anteriores e recentes; (ii) de coleção, que consiste em tópicos que ocorrem apenas nos textos anteriores ou recentes; e (iii) de texto, que são tópicos existentes em apenas um texto. Os tópicos textuais são identificados por meio de um processo dirichlet hierárquico (HDP de *Hierarchical Dirichlet Process*) estendido para o contexto da SAA, o qual os autores referenciam como h-Uhdp.

Li *et al.* (2012) salientam que computar o dinamismo (surgimento, divisão em dois ou mais tópicos, término) dos tópicos é muito importante para identificar informações relevantes e com atualização. Assim, propõem o uso de agrupamento hierárquico para identificação de tópicos e, posteriormente, por meio de um algoritmo guloso, identificam as sentenças dos textos-fonte que mais bem satisfaçam essa proposição.

Li *et al.* (2012) reportam os resultados do método proposto sobre os córpus da TAC de 2010 e 2011. Para o conjunto de testes da TAC 2010, os autores reportaram R-2 e R-SU4, respectivamente, de 0,085 e 0,125. Já para a TAC de 2011, para as mesmas métricas, os valores aferidos foram de 0,101 e 0,163. Com esses valores, o método de Li *et al.* (2012) se fez superior aos melhores sistemas submetidos à edição.

Louis (2014) propõem um método de SAA por meio do conceito de Surpresa Bayesiana (BALDI; ITTI, 2010), que se baseia na análise da diferença entre duas distribuições de probabilidades. Para tanto, o método de Louis (2014) computa a probabilidade dos termos (palavras) presentes em dois conjuntos textuais (que variam conforme as variações do algoritmo) e, posteriormente, mensura a relevância (surpresa) de cada termo por meio da divergência de Kullback Leibler (KL) entre as duas distribuições. Com esses valores, as sentenças dos textos-fonte são ranqueadas por meio dos valores das respectivas palavras

e inseridas no sumário. A intuição desse método é de que, dado dois conjuntos (textos conhecidos e desconhecidos pelo leitor), um termo possui alto valor de surpresa quando a probabilidade de ocorrência no segundo conjunto é maior do que no primeiro. Com esse princípio, Louis (2014) almeja identificar conteúdo menos redundante para o sumário e, sobretudo, mais novo.

Louis (2014) apresentou quatro abordagens para ranquear as sentenças para o sumário. Na primeira, os termos eram pontuados por meio da divergência KL entre as respectivas probabilidades nos textos conhecidos e o sumário. Dessa forma, Louis (2014) afirma que, embora seja uma aplicação simples do método, o sumário é produzido com objetivo de ser menos redundante com relação ao conhecimento do leitor (textos conhecidos) e ao próprio conteúdo. Na segunda, após a identificar palavras chave presentes nos textos já conhecidos, as sentenças eram ranqueadas por meio do somatório das respectivas relevâncias dos termos chaves com ou sem normalização pelo tamanho da sentença (número de *tokens*). A terceira proposta é bem similar à primeira, porém a análise é realizada com os termos dos textos desconhecidos e o sumário. Por fim, a última proposta do autor faz uso da terceira e algumas das anteriores. Assim, dessa forma o método considera tanto o conhecimento do leitor (textos já lidos) quanto o conteúdo dos textos novos.

Em experimentos reportados pelo autor por meio do córpus da TAC de 2009, a melhor configuração de seu método para o cenário da SAA se deu pelo uso da quarta proposta supracitada, na qual as sentenças foram ranqueadas por meio da seguinte expressão:

$$SR_{avg}(s) - KL_{inp}(s)$$

sendo $SR_{avg}(s)$ a pontuação da sentença s computada por meio do somatório das respectivas palavras que são palavras chave dos textos conhecidos pelo leitor e normalizado pelo tamanho de s; e $KL_{inp}(s)$ o valor de surpresa da sentença para o sumário. Nesse caso, espera-se que as sentenças selecionadas para o sumário sejam aquelas com pouca similaridade com os textos já conhecidos e que também não sejam redundantes com as sentenças já inseridas no sumário. Os valores de R-1 e R-2 reportados pelo autor para essa configuração do método foram de 0,337 e 0,076.

3.1.3 Trabalhos baseados em otimização

Gillick, Favre e Hakkani-Tur (2008) propõem um método de SAA baseado em otimização linear, cujo modelo é composto por variáveis que representam o tamanho e relevância das sentenças. Para identificar relevância, os autores ponderam as sentenças pela soma da frequência dos respectivos conceitos, identificados por bigramas com itens lexicais radicalizados. Gillick, Favre e Hakkani-Tur (2008) sugerem que, idealmente, os conceitos sejam identificados da mesma forma que as UCs da métrica da Pirâmide, mas salientam que tal tarefa é complexa e que, comumente, é executada de forma manual.

Após a identificação das mais bem sentenças ranqueadas, os autores utilizaram uma estratégia bem simples para ordenar as sentenças no sumário, que se baseia na data e posição da sentença no respectivo texto. Como melhorias do modelo, Gillick, Favre e Hakkani-Tur (2008) adicionaram outra configuração que remove segmentos sentencias pouco relevantes identificados por meio de regras. Em outras palavras, os autores propuseram uma abordagem extrativa e outra semi-abstrativa, pois apesar de não gerarem conteúdo novo, alteram o conteúdo das sentenças dos textos-fonte.

Gillick, Favre e Hakkani-Tur (2008), para o método puramente extrativo, reportam R-1, R-2 e BE de 0,359 , 0,088 e 0,013, respectivamente. Já para o método dois, que faz uso de técnicas abstrativas, os valores reportados foram de, respectivamente, 0,362, 0,096 e 0,061. Com esses valores, o método de Gillick, Favre e Hakkani-Tur (2008) mostra-se como uma abordagem muito eficiente, por apresentar bons resultados e ser relativamente simples, por fazer uso de apenas conhecimento superficial (frequências de palavras). Ressalta-se novamente que, como enfatizado pelos autores, melhores resultados poderiam ser alcançados se os conceitos textuais fossem identificador por meio de uma abordagem mais adequada, que considerasse conhecimento semântico.

Gillick *et al.* (2009) enfatizam problemas de qualidade linguística em sumários extrativos, tais como problemas de referências como anáforas e catáforas, que ocorreram na proposta anterior de Gillick, Favre e Hakkani-Tur (2008). Nesse cenário, empregam um algoritmo que ordena as sentenças com relevância por meio de atributos posicionais e descartam sentenças com pronomes não resolvidos. Para identificar pronomes não resolvidos, os autores utilizaram classificadores treinados em córpus por meio de algoritmos de aprendizado de máquina, cuja acurácia não foi divulgada. É importante ressaltar que os autores também não especificaram como realizaram a extração de características posicionais das sentenças.

Em experimentos por meio do córpus da TAC 2009, o método proposto pelos autores obtiveram R-2 de 0,101, BE de 0,06, Responsividade de 4,75, QL de 5,52 e Pirâmide de 0,30. Com esses resultados, Gillick *et al.* (2009) observaram que a remoção de sentenças com pronomes não resolvidos não interferiram, significativamente, na qualidade de informação do sumário.

Du *et al.* (2010) apresentam um algoritmo guloso por meio do uso de ranqueamento com pontos fixos para a SAA. Nessa abordagem, cada sentença é representada por um ponto no espaço de ranqueamento e pode ser considerada livre ou fixa, quando, respectivamente, poderá ou não ser inserida no sumário. Em uma primeira iteração do algoritmo, todas as sentenças dos textos conhecidos pelo leitor são consideradas pontos fixos e todas as restantes como pontos livres. Posteriormente, a sentença mais bem ranqueada é inserida no sumário e toda a pontuação é recalculada enquanto a taxa de compressão não seja atingida.

Na proposta de Du *et al.* (2010), uma sentença é ranqueada conforme um parâmetro α que pondera sua dissimilaridade para com os pontos fixos e, eventualmente, sua similaridade com um tópico previamente estipulado. Para tanto, operações matriciais são executadas iterativamente até a convergência do modelo de forma a considerar as seguintes informações: similaridade sentencial (aferida por meio da métrica de Cosseno (SALTON; WONG; YANG, 1975)); e estado de liberdade da sentença (livre ou fixa). Após a convergência do modelo, a sentença mais bem ranqueada é inserida no sumário e o processo se repete enquanto tamanho do sumário seja atingido.

Os autores realizaram experimentos no conjunto de dados da TAC de 2008 e 2009, por meio da R-2 e R-SU4 e compararam com os respectivos *baselines* e melhores métodos das edições. Na edição de 2008, o método de Du *et al.* (2010) obteve 0,102 de R-2 e 0,137 de R-SU4, sendo superior aos *baselines* e ao maior resultado de TAC 2008. Na edição de 2009, o método não foi superior ao método primeiro colocado da TAC, entretanto, os autores enfatizam que esse método emprega abordagens abstrativas, que geram sumários melhores que técnicas extrativas, utilizada pelos autores. Por essa razão, os autores analisam os resultados com o método puramente extrativo mais bem colocado da TAC 2009, e mostraram-se superiores, com 0,099 de R-2 e 0,131 de R-SU4.

O ranqueamento proposto por Du *et al.* (2010) faz uso de um processo iterativo que considera, basicamente, a similaridade entre as sentenças dos textos, de forma que o ranqueamento é executado apenas sobre essa informação. Apesar do resultado satisfatório da abordagem, outras informações, tais como posição sentencial, poderiam ser adicionadas no modelo com objetivo de melhorar a identificação de relevância das sentenças. Da forma que é proposto, desconsiderando-se a similaridade da sentença para um tópico dado, a abordagem dos autores pode ser definida como um algoritmo iterativo com objetivo de encontrar a sentença menos similar com as demais a cada etapa de execução.

Long *et al.* (2010) propõem a criação de sumários com a menor redundância possível. Para tanto, fazem uso da teoria de distância de informação baseada na Complexidade de Kolmogorov (LI; VITáNYI, 2007), que permite aferir valores de dissimilaridade e similaridade entre conjuntos de informações (sentenças). Por meio da Complexidade de Kolmogorov, os autores propõem uma função de custo para mensurar a redundância do sumário. É importante ressaltar que, os autores fazem uso de duas abordagens de aproximação da Complexidade de Kolmogorov, uma baseada em compressão e outra em sobreposição de unidades semânticas (palavras, entidades nomeadas, etc.), pois tal teoria da distância da informação é não computável.

Na etapa inicial do algoritmo, os autores utilizam a teoria de distância de informação supracitada para identificar e descartar sentenças dos textos recentes que sejam muito similares às sentenças dos textos conhecidos pelo leitor. Com as sentenças restantes, todas as permutações possíveis para o sumário, respeitando-se a taxa de compressão, são gera-

das e a função de custo supracitada é aplicada. Por fim, o sumário com maior relevância ou menor custo (menor redundância segundo a Complexidade de Kolmogorov) é selecionado como sumário final. Embora o algoritmo proposto seja uma abordagem de força bruta, os autores afirmam que é factível devido à taxa de compressão utilizada (somente 100 palavras) e que, geralmente, após a primeira etapa do algoritmo, várias sentenças são descartadas e, portanto, poucas sentenças são empregadas no processo de permutações de sumários.

Os autores realizaram experimentos por meio dos córpus de testes da DUC 2007 e TAC 2008 e 2009, nos quais obtiveram R-1 de aproximadamente 0,37. Com esse valor de avaliação, o método de Long *et al.* (2010) mostra-se superior no córpus da DUC 2007, porém, não supera outros métodos nos demais cenários. Assim, embora tenha obtido ótimos resultados, a complexidade computacional do algoritmo é alta e outras abordagens escaláveis produzem resultados semelhantes ou superiores.

Li, Du e Shen (2011) fazem uso de Relevância Marginal Máxima (MMR, do inglês, *Maximal Marginal Relevance*) (CARBONELL; GOLDSTEIN, 1998). Nessa abordagem, uma sentença possui alta relevância se é similar a um grupo de informação e dissimilar com outro, ou seja, no contexto da SAA por meio dessa abordagem, uma sentença é bem pontuada se é similar às sentenças dos textos recentes e dissimilar às sentenças dos textos conhecidos pelo leitor.

Por meio dos conceitos supracitados, Li, Du e Shen (2011) propuseram uma função de custo para o sumário com objetivo de produzir sumários com a Relevância Marginal Máxima. Assim, em um processo iterativo, que é executado enquanto a taxa de compressão não seja atingida, a sentença que maximiza o ranqueamento do sumário é selecionada.

Em experimentos reportados pelos autores por meio do conjunto de dados da TAC 2008 e 2009, o método proposto obteve R-1, R-2 e BE, respectivamente, iguais a 0,360, 0,085 e 0,201. Para o córpus da TAC de 2009, para as mesmas métricas anteriores, os valores aferidos foram de 0354, 0,083 e 0,192. Com essa avaliação, a proposta de Li, Du e Shen (2011) mostrou-se superior aos métodos submetidos aos eventos e alguns outros trabalhos também avaliados sobre o mesmos córpus.

He, Qin e Liu (2012) propõem o uso de ranqueamento múltiplo e agrupamento por meio de grafos de similaridade sentencial. Primeiramente, um grafo é construído para representar as sentenças dos textos fontes (vértices) e o grau de similaridade entre essas (arestas) por meio da métrica do Cosseno (SALTON; WONG; YANG, 1975). Na segunda etapa, o grafo construído é utilizado em dois processos paralelos, um para ranquear as sentenças e outro para agrupá-las conforme o tópico (tema) que abordam. Com isso, os autores almejam produzir sumários que abranjam os conteúdos dos textos fonte por meio das sentenças mais relevantes para cada tópico presente nos textos.

Para ranquear as sentenças, He, Qin e Liu (2012) fazem uso de duas informações (eventualmente três, caso um tópico para direcionar o sumário seja estipulado). São elas: (i) as sentenças dos textos anteriores; (ii) as primeiras sentenças de cada texto da coleção não conhecida pelo leitor; e (iii) as sentenças presentes nos sumários dos textos já conhecidos. Com as informações na primeira categoria, os autores partem do princípio que sentenças dos textos recentes mais distintas dos textos já lidos provavelmente possuem conteúdo com novidade.

Com a segunda categoria de informação, os autores visam identificar relevância. Para tanto, eles partem da hipótese de que as sentenças nas primeiras posições textuais são mais relevantes e, posteriormente, utilizam a similaridade de outras sentenças com essas para mensurar relevância. Em outras palavras, uma sentença é relevante se considerada similar com aquelas que ocorrem nas primeiras posições textuais. Tal estratégia foi foco de investigação nos trabalho de Katragadda, Pingali e Varma (2009) e Ouyang *et al.* (2010).

Por fim, por meio da terceira categoria supracitada, He, Qin e Liu (2012) almejam identificar quais as informações já conhecidas pelo leitor que, provavelmente, terão atualizações nos textos recentes. Para tanto, os autores determinam que as sentenças dos textos conhecidos que foram inseridas nos respectivos sumários compõem-se de conteúdo relevante e que, eventualmente, terão atualizações ou estarão presentes nos textos mais recentes.

Após os dois processos supracitados, o algoritmo de He, Qin e Liu (2012) mescla os resultados de ambos e ordena os subtópicos. Com isso, identificam-se os subtópicos e as respectivas sentenças mais bem ranqueadas que devem ser inseridas no sumário. Em experimentos realizados no córpus da TAC 2008, a melhor configuração do algoritmo obteve bons resultados, atingindo 0,879 de R-2, 0,123 de RSU-4 e 0,522 de BE. Com esses valores, o algoritmo He, Qin e Liu (2012) supera o método mais bem colocado da edição.

Li, Du e Shen (2013) apresentam algumas alterações para o método proposto por Du *et al.* (2010), que, segundo os autores, são necessárias, pois esse método, bem como alguns outros algoritmos de SAA baseados em grafo, permitem que sentenças dos textos conhecidos pelo leitor interfiram na relevância das sentenças dos que serão sumarizados. Assim, no intuito de amenizar esse problema, Li, Du e Shen (2013) adicionam um parâmetro ao modelo de otimização para que explicitamente desconsidere a saliência das sentenças dos textos recentes que sejam similares às sentenças já conhecidas pelo leitor. Dessa forma, sentenças são melhores ranqueadas quando mais semelhantes aos textos recentes.

O modelo de otimização proposto por Li, Du e Shen (2013) é um problema NP Difícil, embora seja viável considerando a quantidade de textos e o tamanho do sumário (100 palavras como definido nas tarefas da DUC e TAC). Assim, para diminuir o tempo computacional do método, os autores propuseram uma aproximação para o algoritmo. Os

autores reportaram experimentos e resultados para os métodos originais e aproximado sobre os conjuntos de dados da TAC de 2008 e 2009. No primeiro conjunto, o modelo original obteve R-1 de 0,370, R-2 igual a 0,088 e 0,201 de BE; o modelo aproximado, respectivamente para as mesmas métricas anteriores, apresentou 0,366, 0,085 e 0,200. Já para o conjunto de dados da TAC 2009, o modelo original pontuou 0,362 de R-1, R-2 de 0,085 e 0,192 de BE; o método aproximado, respectivamente, obteve 0,361, 0,086 e 0,188.

Com os resultados supracitados, os algoritmos de Li, Du e Shen (2013) não superaram os melhores métodos para os respectivos conjuntos de dados. Entretanto, mostraram-se com resultados satisfatórios e superiores aos trabalhos que apresentam o problema que os autores pretendiam amenizar, tais como as propostas de Wenjie *et al.* (2008) e Du *et al.* (2010), entre outros algoritmos que empregam ranqueamento por meio de grafos. Além disso, é importante ressaltar que o método original dos autores possui alta complexidade computacional, mas que o algoritmo aproximado, embora com resultados inferiores, possui desempenho satisfatório, que também supera os trabalhos referenciados anteriormente nesse parágrafo.

3.1.4 Discussão e considerações sobre os métodos de SAA

Como apresentado anteriormente, a SAA tem sido investigada por diversas abordagens, que incluem desde a aplicação de técnicas de SA multidocumento com a remoção de sentenças redundantes e técnicas mais sofisticadas, que visam produzir sumários com a máxima quantidade de informação relevante e com novidade por meio de algoritmos variados. Nesta seção, será apresentada uma discussão sobre esse cenário da SAA.

Para uma visão geral dos métodos apresentados neste capítulo, no Quadro 12, os trabalhos são listados em ordem cronológica com a respectiva abordagem e método de identificação de novidade. Por exemplo, o método de Reeve e Han (2007), na segunda linha do quadro, foi rotulado como da abordagem Sensível ao Contexto e emprega atributos de frequência para identificar sentenças com atualização.

Com objetivo de contrastar a qualidade dos métodos apresentados, na Tabela 3, os resultados dos métodos apresentados agrupados para os respectivos conjuntos de dados é listada. Os valores da avaliação da Pirâmide e QL não são listados nessa tabela, pois poucos trabalhos os divulgaram. Os trabalhos são listados por meio do valor da R-2 em ordem decrescente. Para as demais métricas, os valores em negrito e em itálico correspondem aos métodos primeiro e segundo mais bem ranqueados, respectivamente. Por exemplo, para o conjunto de dados da TAC 2008, o método com maior R-2 foi Wang e Li (2010) e com maior R-1 foi Ouyang *et al.* (2010).

Os métodos baseados na abordagem sensível ao contexto, em geral, são aqueles que se propõem a identificar conteúdo com novidade por meio da remoção de sentenças

Quadro 12 – Trabalhos apresentados cronologicamente ordenados.

Autores	**Abordagem**	**Novidade**
Reeve e Han (2007)	Sensível ao contexto	Atributos de frequência
Bawakid e Oussalah (2008)	Sensível ao contexto	Atributos posicionais
Boudin, El-Bèze e Moreno (2008)	Sensível ao contexto	Sentenças não redundantes
Gillick, Favre e Hakkani-Tur (2008)	Otimização	Atributos posicionais
Wenjie *et al.* (2008)	Sensível ao contexto	Reforço positivo e negativo em um grafo
Gillick *et al.* (2009)	Otimização	Atributos posicionais
Katragadda, Pingali e Varma (2009)	Posicional	Atributos posicionais das sentenças
Steinberger e Ježek (2009)	Análise de tópico	Sentenças de tópicos não redundantes
Du *et al.* (2010)	Otimização	Sumário com menor redundância
Varma *et al.* (2009)	Otimização	Análise de Vocabulário
Huang e He (2010)	Análise de tópico	Distribuição tópicos
Long *et al.* (2010)	Otimização	
Ouyang *et al.* (2010)	Sensível ao contexto	Atributos posicionais e de redundância
Wang e Li (2010)	Análise de tópico	Agrupamento hierárquico
Li, Du e Shen (2011)	Otimização	Sumário com menor redundância
Delort e Alfonseca (2012)	Análise de tópico	Análise do comportamento dos tópicos
He, Qin e Liu (2012)	Otimização	Relevância Marginal Máxima
Li *et al.* (2012)	Análise de tópico	Agrupamento hierárquico
Li, Du e Shen (2013)	Otimização	Sumário com menor redundância
Louis (2014)	Análise de tópico	Kullback Leibler

redundantes em relação aos textos conhecidos pelo leitor. Como salientado por Delort e Alfonseca (2012), tal estratégia não é adequada para identificar conteúdo com novidade que ocorre junto com informações conhecidas. Contudo, pode-se notar que em alguns casos esses métodos apresentam resultados muito satisfatórios. Por exemplo, sobre o conjunto de dados da DUC 2007 e TAC 2009, os trabalhos mais bem ranqueados, Boudin, El-Bèze e Moreno (2008) e Gillick *et al.* (2009), são dessa abordagem. Entretanto, é importante ressaltar que esses dois métodos apresentam propostas mais sofisticadas para selecionar as possíveis sentenças para o sumário. Boudin, El-Bèze e Moreno (2008) utiliza o algoritmo

Tabela 3 – Avaliação dos métodos por *dataset*.

	Autores	Abordagem	R-1	R-2	R-SU4	BE
DUC 2007	Boudin, El-Bèze e Moreno (2008)	Sensível ao cont.	*0,363*	0,102	**0,139**	
	Reeve e Han (2007)	Sensível ao cont.		0,090		
	Wenjie *et al.* (2008)	Sensível ao cont.	0,362	0,086	0,129	
	Witte, Krestel e Bergler (2007)	Sensível ao cont.	0,296	0,053	0,096	0,024
	Long *et al.* (2010)	Otimização	**0,370**			
TAC 2008	Wang e Li (2010)	Análise de tópico	*0,371*	0,930	0,135	
	He, Qin e Liu (2012)	Otimização		0,879	0,124	0,052
	Ouyang *et al.* (2010)	Sensível ao cont.	**0,375**	0,394		
	Du *et al.* (2010)	Otimização		0,102	*0,138*	
	Huang e He (2010)	Análise de tópico	0,367	0,097	0,132	
	Gillick, Favre e Hakkani-Tur (2008)	Otimização	0,362	0,096		0,061
	Katragadda, Pingali e Varma (2009)	Posicional		0,093	0,128	
	Li, Du e Shen (2013)	Otimização	0,370	0,088		0,201
	Li, Du e Shen (2011)	Otimização	0,360	0,085		0,201
	Steinberger e Ježek (2009)	Análise de tópico		0,081	0,121	0,059
	Bawakid e Oussalah (2008)		0,341	0,080	0,117	
	Long *et al.* (2010)	Otimização	0,370			
TAC 2009	Gillick *et al.* (2009)	Otimização		0,104		0,061
	Varma *et al.* (2009)	Otimização		0,101	**0,138**	
	Du *et al.* (2010)	Otimização		0,099	*0,131*	
	Huang e He (2010)	Análise de tópico	0,369	0,097	0,136	
	Ouyang *et al.* (2010)	Sensível ao cont .	**0,373**	0,096		
	Li, Du e Shen (2013)	Otimização	0,362	0,085		0,192
	Li, Du e Shen (2011)	Otimização	0,354	0,084		0,192
	Louis (2014)		0,337	0,076		
	Long *et al.* (2010)	Otimização	*0,370*			
2011	Li *et al.* (2012)	Análise de tópico		0,101	0,163	
	Delort e Alfonseca (2012)	Análise de tópico	0,358	0,092	0,128	

MMR (CARBONELL; GOLDSTEIN, 1998), e Gillick *et al.* (2009) enfatizam melhorias do algoritmo de Gillick, Favre e Hakkani-Tur (2008) para maior qualidade linguística dos sumários.

Após 2010, os trabalho de SAA foram mais direcionados à tarefa de Sumarização de Linha Temporal, TS, do inglês *Timeline Summarization*, como pode ser observado nos trabalhos de Gao, Li e Zhang (2013), Li e Li (2013). Nessa nova tarefa no âmbito da SA, tem-se o objetivo de produzir uma sequência de sumários de atualização a partir de um conjunto de textos-fonte relacionados por tema e distribuídos em diferentes momentos temporais em um determinado intervalo de tempo. Por exemplo, todos as notícias reportadas sobre algum evento esportivo com duração de semanas, de forma que para cada dia é gerada um sumário de atualização.

3.2 Compressão de Sentenças

A tarefa de Compressão Sentencial tem por objetivo a produção automática de uma paráfrase de menor tamanho (número de itens lexicais) a partir de uma sentença de entrada (KNIGHT; MARCU, 2000). Essa tarefa é muito relevante para inúmeras aplicações, sobretudo, no campo do Processamento de Língua Natural, tais como a Sumarização Automática de textos propostos nos trabalhos de (KNIGHT; MARCU, 2000; MADNANI *et al.*, 2007; BERG-KIRKPATRICK; GILLICK; KLEIN, 2011; LI *et al.*, 2013; QIAN; LIU, 2013; ALMEIDA; MARTINS, 2013; FILIPPOVA *et al.*, 2015), produção automática de títulos (FILIPPOVA; ALTUN, 2013), e melhoria de legendas de formas automática (LUOTOLAHTI; GINTER, 2015).

Geralmente, métodos de Compressão Sentencial (CS) baseiam-se em uma Abordagem de Deleção, na qual alguns itens lexicais ou arcos de uma árvore sintática são removidos da sentença de entrada, como empregado nos trabalhos de Knight e Marcu (2000), Turner e Charniak (2005), McDonald (2006), Cohn e Lapata (2008), Almeida e Martins (2013), Berg-Kirkpatrick, Gillick e Klein (2011), Thadani e McKeown (2013). Nesse cenário, a tarefa de CS pode ser interpretada como uma sequência de n decisões sobre uma sentença s com n itens lexicais ou arcos sintáticos. Dessa forma, em cada decisão i, decide-se se o i-ésimo item lexical ou arco sintático será ou não removido.

Knight e Marcu (2000) propuseram dois métodos de CS baseados em diferentes algoritmos de Aprendizado de Máquina. Para o primeiro método, Knight e Marcu (2000) treinaram um modelo Noisy-Channel com objetivo de identificar os padrões sintáticos mais recorrentes durante o processo de compressão de uma sentença. Para tanto, os autores fizeram uso da respectiva árvore sintática das sentenças de entrada e alguns modelos sintáticos de compressão, que foram definidos empiricamente. Por exemplo, os autores definiram que a sequência de sintagmas SN (Sintagma Nominal) SV (Sintagma Verbal) SP (Sintagma Preposicional), comumente é comprimida para o padrão SN SP. Aqui, ressalta-se que esses modelos sintáticos de compressão utilizados pelos autores não necessariamente ocorrem nos córpus investigados por eles. Além disso, visando a produção de sentenças reduzidas sem erros gramaticais, os autores aplicaram um modelo de língua para ranquear as sentenças produzidas e desconsiderar aquelas com pontuações baixas. Para o segundo método de CS proposto, Knight e Marcu (2000) treinaram um modelo de Árvore de Decisão que representa operações de compressão sobre a árvore sintática da sentença de entrada. Após o treinamento, em um processo *bottom-up* (dos itens lexicais para a estrutura da árvore), o método processa cada item lexical sequencialmente, aplicando as operações aprendidas para produzir uma árvore sintática reduzida em relação à original. Nos experimentos reportados pelos autores, o modelo baseado no algoritmo de Noisy-Channel produziu sentenças reduzidas com maior gramaticalidade. Porém, o segundo método apresentou sentenças com maior informatividade.

Turner e Charniak (2005) introduziram alguns aprimoramentos ao método supracitado baseado no algoritmo de Noisy-Channel. Primeiramente, Turner e Charniak (2005) utilizaram um modelo de língua mais robusto, que possui mais informação sintática incorporada. Dessa forma, o sistema proposto poderia desconsiderar sentenças com erros gramaticais com mais eficiência. Para treinamento do modelo de Noisy-Channel, os autores também utilizaram modelos sintáticos para compressão, mas somente aqueles que ocorreram no conjunto de dados utilizados para treinamento do modelo. Segundo eles, essa restrição faz com que o método proposto produza sentenças comprimidas com menor incidência de erros gramaticais. Porém, uma vez que tal restrição pode reduzir a quantidade de padrões sintáticos para compressão aprendidos pelo modelo, algumas instâncias não supervisionadas foram inseridas no conjunto de treinamento por meio de uma metodologia conservadora. Em outras palavras, essas novas instâncias possuem uma taxa de compressão inferior (menos itens lexicais eram removidos), partindo da hipótese de que a adição de sentenças com poucas operações de compressões no córpus pode contribuir para que o modelo aprenda novos padrões de compressão sem diminuir sua gramaticalidade. Nos experimentos reportados, Turner e Charniak (2005) indicam que o método proposto gerou sentenças comprimidas mais relevantes e gramaticais, porém maiores, em relação ao trabalho de Knight e Marcu (2000).

Kawamoto e Pardo (2010) investigaram diferentes algoritmos de Aprendizagem de Máquina para a construção de um método de CS para a língua Portuguesa. De forma distinta aos trabalhos anteriores, os modelos apresentados por esses autores se baseiam na tarefa de Classificação. Dessa forma, uma dada sentença s com n itens lexicais é comprimida após n decisões do classificador, sendo que em cada decisão i, o classificador decide se o i-ésimo item da sentença será removido ou não. Os autores investigaram diferentes níveis de atributos superficiais, sintáticos e semânticos que foram extraídas após o pré-processamento das sentenças por meio do parser sintático PALAVRAS (BICK, 2000). Além disso, eles também propuseram atributos baseados nos respectivos documentos em que as sentenças ocorriam, tais como a frequência do item lexical no documento e a respectiva posição da sentença no documento. Kawamoto e Pardo (2010) propuseram atributos interessantes para a tarefa de CS, porém, o conjunto de dados utilizado era relativamente pequeno em relação aos córpus empregadas para a língua Inglesa, de forma que seus resultados não foram muito satisfatórios.

Martins e Smith (2009) propuseram um modelo de Programação Linear Inteira (ILP, do Inglês *Integer Linear Programming*) que verificar atributos superficiais e sintáticos das sentenças. O método proposto por esses autores também é baseado em operações de deleções, porém, sobre os arcos de relações de dependência presentes em uma árvore sintática. Como atributos superficiais, Martins e Smith (2009) utilizaram informações para cada item lexical individualmente, tais como a respectiva posição na sentença, se o item lexical é uma *stopword* ou algum modificador (negação ou temporal). Como atributos

sintáticos, os autores utilizaram os rótulos sintáticos da respectiva árvore sintática da sentenças. Além do treinamento do modelo sob um córpus, os autores também incorpora algumas restrições manuais linguisticamente guiadas para indicar algumas compressões que não devem ser executadas, tal como remover o verbo principal da sentença.

Almeida e Martins (2013) estenderam o modelo de ILP originalmente apresentado em Martins e Smith (2009) e o utilizaram em um método de Sumarização Automática. Tendo em vista esse cenário para aplicação do sistema de CS, a principal diferença para o sistema de CS de Almeida e Martins (2013) para a proposta original é que o treinamento do modelo de ILP para compressão foi executado em conjunto com um modelo para a tarefa de SA. Além disso, eles também utilizaram alguns atributos superficiais, que foram extraídos a partir dos textos-fonte em que as sentenças ocorriam, tais como a frequência de ocorrência e a posição da sentença no textos-fonte. Nos experimentos reportados por Almeida e Martins (2013), foi demonstrado que esses novos atributos auxiliaram à produção de sentenças comprimidas com maior qualidade no cenário de aplicação da CS na Sumarização Automática.

Thadani e McKeown (2013) também propuseram um modelo de ILP para a tarefa. O método proposto aplica operações de deleções de itens lexicais e arcos sintáticos da sentença de entrada. Segundo esses autores, essa abordagem faz com que a estrutura sintática da sentença seja mais preservada na sentença comprimida. Thadani e McKeown (2013) utilizaram quatro distintos de atributos, que eles referenciaram como informatividade, fluência, fidelidade e pseudo-normalização. No primeiro grupo, o modelo proposto analisa informações sintáticas das sentenças, como etiqueta morfossintática (POS) do item lexical sendo analisado e dos três itens em torno dele; raiz do verbo mais próximo; uma indicação se o item lexical é uma palavra de negação; se o item lexical ocorre entre parênteses; e se o item lexical faz parte de uma cadeia de item lexical em caixa alta. Para o segundo grupo, o método analisa informação da estrutura da árvore sintática. No segundo grupo de atributos, os autores utilizaram um modelo de língua com o objetivo de descartar sentenças produzidas que apresentam baixa gramaticalidade. Por fim, os autores adicionaram um atributo de penalização, que indica a relevância para a remoção ou não de uma palavra dado o tamanho da sentença. Os autores indicaram que esse atributo foi necessário para indicar ao modelo de ILP alguma restrição de tamanho desejado para a sentença comprimida

Filippova *et al.* (2015) investigaram alguns métodos de CS baseado em deleção de itens lexicais por meio de técnicas de *Deep Learning*. Para tanto, os autores treinaram três diferentes arquiteturas de Redes Neurais com neurônios do tipo LSTM (*Long Short-Term Memory*). Dessa forma, como é frequentemente empregada nesse tipo de abordagem, os autores modelaram cada item lexical por meio de uma representação vetorial, comumente referenciado como *Embedding*. Filippova *et al.* (2015) utilizaram vetores com 256

dimensões que foram obtidos pelo modelo *skip-gram* proposto por Mikolov *et al.* (2013). O primeiro modelo apresentado utiliza apenas dois atributos para cada item lexical, a respectiva representação vetorial e uma indicação se o item lexical anterior foi ou não removido da sentença. Os demais modelos propostos, referenciados como LSTM+PAR e LSTM+PAR+PER, além dos atributos anteriores para cada item lexical da sentença, foi incorporado o respectivo *embedding* do item lexical com dependência sintática. Além disso, no modelo LSTM+PAR+PER, foram adicionadas dois atributos com valores lógicos: a primeira é verdadeira caso a dependência sintática do item tenha sido removido da sentença; já a segunda, é verdadeira quando a respectiva dependência sintática do item lexical não tenha sido ainda processada pelo método. Nos experimentos reportados em Filippova *et al.* (2015), pode-se observar que a arquitetura LSTM+PAR+PER alcançou os melhores resultados, com cerca de 82% de F-1). Contudo, segundo os autores, o primeiro modelo proposto, que só utiliza os *embeddings* como atributos, é muito relevante, pois não requer processamento sintático das sentenças e alcançou 80% de F-1.

3.2.1 Considerações finais sobre os métodos de Compressão Sentencial

Os trabalhos de Compressão Sentencial apresentados anteriormente, em geral, fazem uso de diferentes atributos superficiais (que podem ser capturadas a partir de um pré-processamento simples da sentença de entrada) e sintáticas. Algumas desses atributos podem ser interpretadas como uma intuição natural para a tarefa de CS. Por exemplo, algumas marcações como parênteses e palavras de negação podem ser bons indicativos para remoção ou não de um ou mais itens lexicais de alguma sentença. Além disso, é importante ressaltar que os trabalhos que fazem uso de conhecimento sintático, usualmente fazem uso de um parser de dependências sintáticas. Esse tipo de estrutura sintática é relativamente mais simples do que uma Árvore de Constituintes, porém é também muito informativa. Além disso, como descrito por alguns autores (FILIPPOVA *et al.*, 2015), comumente, parsers de dependência sintáticas são menos suscetíveis a erros do que a identificação de sintagmas.

Independentemente das abordagem ou dos tipos de atributos utilizados, pode-se observar também que a maioria dos trabalhos analisam decisões prévias para decidir de um item lexical ou arco sintático será removido. Por exemplo, no trabalho de Filippova *et al.* (2015), o modelo de melhor resultados nos experimentos dos autores faz uso de dois atributos dessa categoria para decidir remoção de um determinado item lexical t, sendo eles: a verificação se o item lexical logo anterior t foi removido da sentença, que também é utilizado pelo modelo mais simples do trabalho supracitado; e a indicação se o item lexical identificado como dependência sintática de t foi removido ou não da sentença.

É interessante observar o uso de um modelo de língua em alguns trabalhos visando

a produção de sentenças comprimidas com alta gramaticalidade, tais como (KNIGHT; MARCU, 2000; TURNER; CHARNIAK, 2005; THADANI; MCKEOWN, 2013). Possivelmente, essa abordagem foi empregada pelos autores pois seus respectivos sistemas produzem mais de uma versão comprimida para uma mesma sentença de entrada, de forma que o modelo de língua é utilizado para identificar àquela mais adequada. Além disso, eventualmente, os autores podem ter considerado que o conjunto de dados disponíveis para treinamento não era suficiente para a produção de versões comprimidas de qualidade. Por exemplo, no método baseado em *Deep Learning* introduzido em Filippova *et al.* (2015), que não faz uso de modelo de língua diretamente para ranquear as sentenças, havia disponível um conjunto de dados para treinamento com mais de 10 mil pares de sentenças originais e versões compridas. Aqui, é importante ressaltar que é dito "diretamente", porque os modelos de representação vetoriais podem ser considerados um modelo de língua em alguns cenários.

Quadro 13 – Resumo dos trabalhos de CS apresentados.

Autores	**Técnica**	**Recursos**	**Deleção sob**
Knight e Marcu (2000)	Árvore de Decisão e Naïve-Bayes	Árvore Sintática, Modelo de Língua, Padrões de Compressão	Arcos sintáticos
Turner e Charniak (2005)	Naïve-Bayes	Árvore Sintática, Modelo de Língua, Padrões de Compressão	Arcos sintáticos
Kawamoto e Pardo (2010)	Árvore de Decisão	Árvore Sintática, Informações Superficiais	Itens Lexicais
Martins e Smith (2009)	ILP	Árvore Sintática, Informações Superficiais, Posição Sentencial, Restrições Manuais	Arcos sintáticos
Almeida e Martins (2013)	ILP	Árvore Sintática, Informações Superficiais, Posição Sentencial, Restrições Manuais	Arcos sintáticos
Thadani e McKeown (2013)	ILP	Árvore Sintática, Informações Superficiais	Arcos sintáticos
Filippova *et al.* (2015)	*Deep Learning*	*Embeddings* e Relações de Dependências Sintáticas	Itens Lexicais

CAPÍTULO

4

MÉTODOS DE COMPRESSÃO SENTENCIAL

Neste Capítulo, serão introduzidos os dois córpus para Compressão Sentencial empregados neste trabalho, bem como a metodologia utilizada para compilá-los e uma análise detalhada de suas características. Esses recursos foram necessários para desenvolvimento, experimentação e avaliação dos métodos automáticos de Compressão Sentencial propostos.

Os dois córpus, referenciados como Pares-PCSC e Pares-G1, serão descritos na Seção 4.1. Posteriormente, na Seção 4.2, será apresentada a investigação inicial para a construção de métodos de Compressão Sentencial, bem como a avaliação desses métodos desenvolvidos. Nesse primeiro momento, o objetivo dos experimentos foi à identificação do algoritmo de Aprendizagem de Máquina mais adequado para a tarefa de CS e analisar diferentes níveis de atributos que podem contribuir para essa tarefa. Por fim, na Seção 4.3.5, dissertara-se as propostas de aprimoramento desses modelos iniciais. Para esses novos métodos, também serão descritas a metodologia de avaliação e comparação com outros trabalhos da literatura.

4.1 Córpus de Compressão Sentencial compilados

Nos dois córpus que serão introduzidos em seguida, há diversas instâncias para a tarefa de CS. Cada instância é composta por um par de sentenças, (s, s_c), sendo que s uma sentença original, que foi extraída de algum texto jornalístico, e s_c, a respectiva versão comprimida de s.

Uma vez que foram desenvolvidos métodos de CS por meio da abordagem de Deleção, que é a mais frequente utilizada na literatura (KNIGHT; MARCU, 2000; TURNER; CHARNIAK, 2005; MCDONALD, 2006; COHN; LAPATA, 2008; ALMEIDA; MARTINS, 2013; BERG-KIRKPATRICK; GILLICK; KLEIN, 2011; THADANI; MCKEOWN, 2013),

assumiu-se que uma sentença s_c é uma versão comprimida de alguma outra sentença s se, e somente se, é possível produzir a primeira por meio de uma sequência de zero ou mais deleções de itens lexicais em s. Dessa forma, não foi considerado abordagens de Compressão Sentencial que fazem uso de procedimentos de alteração, simplificação ou reescrita de conteúdo. Por exemplo, no Quadro 14, são dispostas três pares de sentenças, sendo apenas o primeiro considerado válido para este trabalho. No segundo caso, é utilizado um sinônimo que não permite gerar a versão comprimir por apenas deleções da sentença original. No terceiro par, é necessário alterar a localização de alguns sintagmas.

Quadro 14 – Exemplo de instâncias de compressão.

s:	Estão abertas as inscrições para um passeio ciclístico em Divinópolis .
s_c:	Abertas inscrições para passeio ciclístico em Divinópolis.
s:	Estão abertas as inscrições para um passeio ciclístico em Divinópolis .
s_c:	Abertas **matrículas** para passeio ciclístico em Divinópolis.
s:	Estão abertas as inscrições para um passeio ciclístico em Divinópolis .
s_c:	Em Divinópolis estão abertas as inscrições para um passeio ciclístico.

Dada à definição supracitada de uma sentença comprimida, pode-se perceber que foram consideradas válidas as instâncias em que a sentença comprimida é igual à sentença original. Ou seja, quando não há nenhuma deleção sobre a sentença original. Essa restrição foi mantida baseando-se no pressuposto de que tais exemplos são importantes para representar situações em que uma sentença qualquer não deve ser reduzida. Além disso, o uso de instâncias com exemplos positivos e negativos (comprimir ou não uma dada entrada) é importante para o treinamento de métodos baseados em Aprendizado de Máquina, que foram investigados neste trabalho.

Os dois córpus compilados, o Pares-PCSC e Pares-G1, bem como a metodologia para construí-los serão detalhados nas seções 4.1.1 e 4.1.2, respectivamente. Posteriormente, uma análise quantitativa e comparativa desses recursos será apresentada na seção 4.1.3.

4.1.1 Córpus Pares-PCSC

Como apresentado no Capítulo 2, o córpus PCSC possui 160 sumários produzidos semiautomaticamente por meio da Síntese Compressiva baseada na Abordagem de Deleção, sendo dois sumários para cada coleção. Dessa forma, uma vez que os sumários produzidos por meio dessa abordagem são constituídos por sentenças extraídas dos textos-fonte que eventualmente são comprimidas, assume-se que qualquer sentença s_{sum} que compõem algum sumário no PCSC é uma versão reduzida de uma ou mais sentenças s_{texto} dos respectivos textos-fonte.

Quadro 15 – Exemplo de sentenças que foram alinhadas com duas versões comprimidas no córpus PCSC. Aqui, as sentenças em negrito foram selecionadas para compor o córpus empregado neste trabalho.

Fonte	Segundo o último balanço do Governo, o sismo de magnitude 8.8 na escala de Richter provocou pelo menos 711 mortos.
Compressão #1	**O sismo de magnitude 8.8 na escala de Richter provocou pelo menos 711 mortos.**
Compressão #2	Segundo o último balanço, o sismo de magnitude 8.8 na escala de Richter provocou pelo menos 711 mortos.
Fonte	No total, 20 Estados indianos de um total de 29 foram afetados, de acordo com a AFP.
Compressão #1	**20 Estados indianos de um total de 29 foram afetados.**
Compressão #2	No total, 20 Estados de um total de 29 foram afetados.
Fonte	As peças do acidente ferroviário, que fez 78 mortos, estão à guarda da polícia num navio na freguesia de Escravitude, na Corunha.
Compressão #1	**As peças do acidente estão à guarda da polícia na Corunha.**
Compressão #2	As peças do acidente estão à guarda da polícia num navio, na Corunha.

Dada às características supracitadas, realizou-se um procedimento automático para identificar os pares correspondentes de sentenças originais dos textos-fonte e respectivas compressões nos sumários do PCSC. Primeiramente, todas as sentenças foram normalizadas com relação à codificação e sinais de pontuação, que podem ter sido alterados durante a construção dos sumários devido ao uso de ferramentas distintas (Sistema Operacional, editores de textos, etc.). Essa etapa foi necessária, pois foram identificados alguns pares sentenciais não alinhados devido à incongruência de sinais de pontuação, tais como o aspas simples (‘ e ’) e dupla (“ e ”). Após a etapa de normalização, foram alinhados todos os pares de sentenças compostos por uma sentença do texto fonte (s_{texto}) e uma outra do respectivo sumário (s_{sum}) considerada uma redução da primeira.

Posteriormente a etapa supracitada, foram identificadas 78 sentenças dos textos-fonte foram alinhadas com duas ou mais versões comprimidas. Esse fenômeno era esperado, pois uma vez que há dois sumários multidocumentos para cada coleção com 10 textos-fonte, há compressões distintas para uma mesma sentença ou sentenças muito similares de diferentes textos no córpus. No Quadro 15, são apresentados alguns exemplos desses casos. Assim, com objetivo de desenvolver métodos de CS que produzam o maior número de deleções possíveis de alguma sentença de entrada, optou-se por selecionar apenas a menor versão reduzida alinhada.

Após as etapas de normalização, alinhamento e filtro supracitadas, foram selecionados 874 instâncias de compressão consideradas válidas para o córpus. Uma vez que essas instâncias foram selecionadas a partir do córpus PCSC, esse córpus de compressão será referenciado como Pares-PCSC ao decorrer desta monografia.

4.1.2 Córpus Pares-G1

Uma vez que a quantidade de instâncias de compressão no córpus Pares-PCSC não é tão expressiva, sobreuto com relação a trabalhos mais recentes para a língua inglesa. Por exemplo, Filippova *et al.* (2015) fizeram uso de um córpus com 12 mil instâncias de compressão. Assim, tendo em vista que o tamanho do córpus influencia diretamente na qualidade dos métodos desenvolvidos, sobretudo quando se emprega técnicas de Aprendizado de Máquina, utilizou-se a metodologia para criação automática de um córpus para a tarefa de CS proposta em (FILIPPOVA; ALTUN, 2013).

A abordagem supracitada parte da hipótese de que títulos de textos jornalísticos publicados em portais de notícias eventualmente ocorrem como a uma compressão da respectiva primeira sentença do texto. Dessa forma, inicialmente para este trabalho, se fez uso de um conjunto de 1.008.356 textos jornalísticos que foram coletados do portal de notícias online G1[1]. Posteriormente, para cada notícia nesse conjunto, extraíram-se os respectivos títulos e primeiras sentenças como possíveis instâncias candidatas para o córpus.

Previamente à seleção das instâncias adequadas para o córpus, optou-se por normalizar todos os candidatos com objetivo de aumentar o número de pares válidos. Essa etapa é importante, pois foram observados alguns padrões recorrentes que inviabilizariam a identificação de algumas instâncias válidas, tais com a alteração dos símbolos de parênteses por hífen. Assim, utilizaram-se as seguintes regras de normalização:

- sentenças não finalizadas com algum sinal pontuação, receberam o símbolo de ponto final (.). De fato, essa regra não interfere no processo de identificação de instâncias de compressão válidas. Porém, optou-se por utiliza-la com intuito de manter a estrutura gramatical das sentenças;
- embora incomum em textos de língua Portuguesa, sobretudo no Brasil, encontrou-se números com as casas decimais separadas pelo sinal de ponto (.), que é comum em língua Inglesa. Assim, esses casos foram normalizados para utilização do símbolo de vírgula como separador de casa decimal;
- com objetivo de minimizar o número de instâncias desconsideradas por causa de problemas de tokenização, adicionaram-se espaços em branco em torno de todos os sinais de pontuação;
- todos os segmentos textuais circundados por aspas duplas foram normalizados para utilização de aspas simples; e
- sinais de pontuação que demarcam siglas de estados brasileiros foram alterados para símbolos de parênteses. Assim, por exemplo, “–SP–” é normalizado para “(SP)”.

[1] <http://www.g1.com.br>

Após todas as etapas supracitadas, foram removidas 32 instâncias válidas selecionadas, pois foram consideradas irrelevantes para a tarefa de Compressão Sentencial. Essas instâncias apresentam alguma das seguintes características: sentenças originais (não comprimidas) menores do que 6 tokens; pares em que o conteúdo da sentença original é a apenas um nome próprio; sentenças com marcações do portal de notícia utilizado (tais como: atualização; links; infográficos; e outros); e sentenças em enumeradas, tais como listas com taxas de investimento. Posteriormente as etapas supracitadas, foram selecionadas 7024 instâncias válidas. Uma vez que esse córpus foi compilado a partir de um conjunto de notícias extraídas do Portal de Notícias G1, esse córpus será referenciado como Pares-G1 no decorrer desta monografia.

4.1.3 Visão geral sobre os córpus Pares-PCSC e Pares-G1

Como apresentado nas Subseções anteriores, os dois córpus para SC compilados neste trabalho, Pares-PCSC e Pares-G1, são constituídos por sentenças extraídas de textos jornalísticos. Entretanto, a forma com que as sentenças dos dois córpus foram comprimidas visam diferentes objetivos, embora sejam baseadas na Abordagem de Deleção. Essa característica é de fato relevante, pois possibilita a investigação da CS em diferentes cenários.

No córpus Pares-PCSC, as sentenças foram comprimidas no âmbito da Sumarização Compressiva. Dessa forma, muito provavelmente, informações relacionadas aos textos-fonte e respectivos sumários das sentenças de entrada influenciaram o processo de compressão manual empregado no córpus, tais como: as sentenças já adicionadas no sumário; a respectiva posição da sentença no sumário; a presença de siglas que já foram expandidas previamente; entre outras.

Por outro lado, no córpus Pares-G1, as sentenças comprimidas foram selecionadas a partir de títulos de notícias. Assim, possivelmente, regras de estilo de escrita ou de edição estipuladas pelo portal de notícias G1 foram utilizadas durante o processo de titulação das notícias. É importante ressaltar que, embora existam diferenças entre os córpus compilados, assumiu-se que ambos os córpus podem ser utilizados para o desenvolvimento de métodos de Compressão por meio de técnicas de Aprendizado de Máquina.

Na Tabela 4, são dispostas informações numéricas que resumem as características e diferenças entre os córpus construídos. É importante ressaltar que para contabilizar o número de tokens em uma sentença, consideram-se também os sinais de pontuação. Ressalta-se que as sentenças (originais e versões reduzidas) presentes no Pares-PCSC são ligeiramente maiores do que àquelas presentes no córpus Pares-G1. Além disso, a Taxa de Compressão (CR, do inglês *Compression Rate*) das sentenças comprimidas presentes no primeiro córpus é menor. O valor da CR indica a proporção de tokens removidos de uma sentença para produzir uma respectiva versão reduzida por meio da seguinte equação:

$\frac{|s|-|s_c|}{|s|}$, onde: $|s|$ indica a quantidade de tokens na sentença original; e $|s_c|$ corresponde à quantidade de tokens da sentença comprimida. Dessa forma, um alto valor de CR indica que mais tokens foram removidos da sentença original.

Tabela 4 – Características numéricas sobre os córpus Pares-PCSC e Pares-G1.

Característica	Pares-PCSC	Pares-G1
Número de sentenças	874	7024
Número de tokens	33108	182111
Tamanho mínimo e máximo das sentenças	8 – 146	7 – 124
Tamanho médio das sentenças originais	37.88	25.92
Tamanho médio das sentenças comprimidas	17.63	19.10
Taxa de compressão (CR) média	42%	57%

Na Figura 5, é exibido um histograma com a distribuição do tamanho das sentenças não comprimidas nos córpus. Uma vez que a quantidade de sentenças presentes nos dois conjuntos de dados é muito diferente (874 – 7024), os valores dispostos no eixo vertical do histograma correspondem à distribuição percentual da quantidade de sentenças. No eixo horizontal, são apresentados os respectivos intervalos de tamanhos considerados. Por exemplo, no córpus Pares-PCSC, há aproximadamente 35% de sentenças com tamanhos entre 20 e 40. Nesse gráfico, é possível observar que a maioria das sentenças possui tamanho entre 20 e 60 e que sentenças muito pequenas ou grandes são pouco frequentes nos dois córpus. Por exemplo, apenas 10% das sentenças possuem menos de 20 tokens no córpus Pares-PCSC.

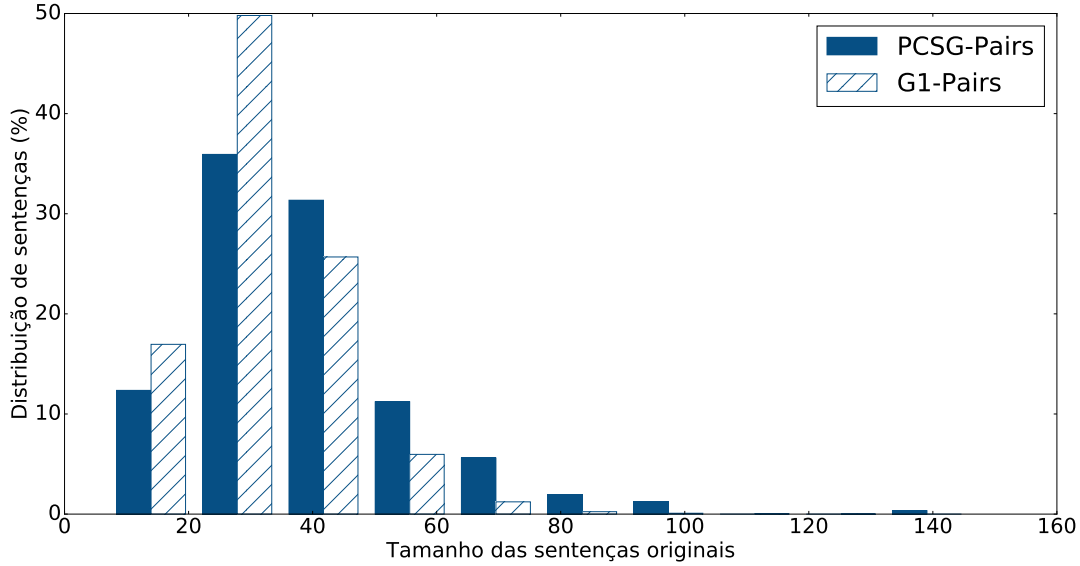

Figura 5 – Distribuição de sentenças por tamanho nos córpus Pares-PCSC e Pares-G1

Na Figura 6, é apresentado outro histograma com a Taxa de Compressão (CR) presente os córpus introduzidos neste Capítulo. De forma similar à Figura 5, a quantidade de sentenças apresentadas no histograma foi normalizado devido à discrepância de

tamanho dos córpus. Por exemplo, aproximadamente 15% das versões comprimidas no córpus Pares-PCSC apresentam valor CR de 40%. Nessa análise, é possível perceber que a distribuição de CR nos dois córpus é similar. Entretanto, há mais sentenças com alta CR (entre 60% e 80%) no córpus Pares-G1, enquanto no outro conjunto de dados o valor máximo ocorre em torno de 50%.

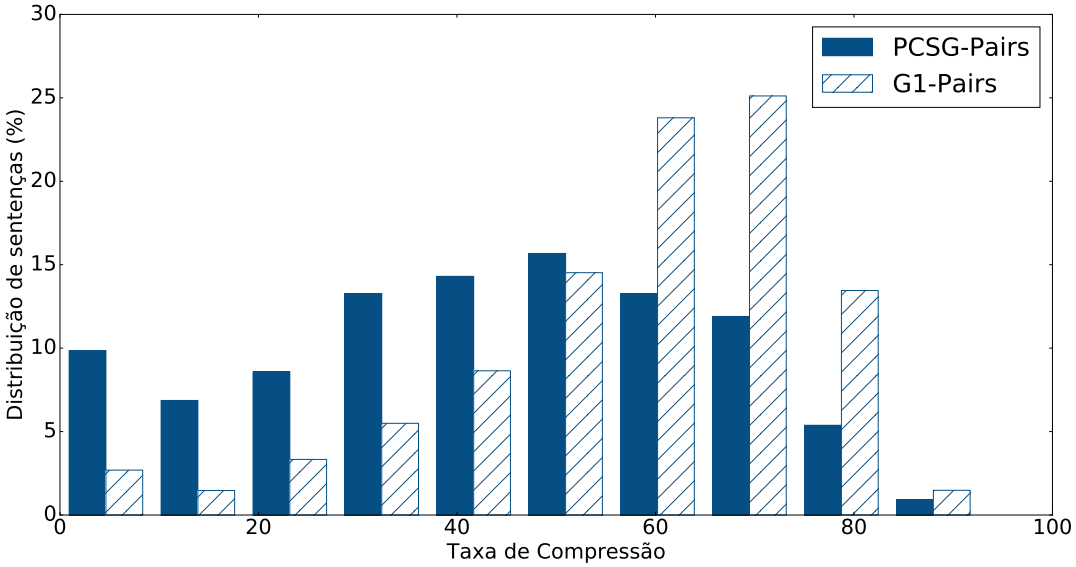

Figura 6 – Histograma com a taxa de compressão presente nos córpus Pares-PCSC e Pares-G1.

Por fim, na Figura 7 é apresentado um gráfico com a correlação entre os tamanhos (número de tokens) entre as sentenças originais e respectivas taxa de compressão das respectivas sentenças comprimidas nos dois córpus apresentados. Primeiramente, as sentenças de cada córpus foram organizadas em 10 grupos com relação à respectiva quantidade de tokens. Assim, para cada grupo, é apresentado o valor de CR médio. Por exemplo, a taxa de compressão presente nas sentenças entre 30 e 45 tokens é de aproximadamente 70% no córpus Pares-G1. Nesse gráfico, é possível observar que o valor de CR aumenta conforme o tamanho das sentenças. Contudo, no córpus Pares-PCSC, há uma variação nos últimos grupos. Além disso, como observado na Figura 6, também é possível verificar nesse gráfico que a taxa de compressão é maior no córpus Pares-G1.

4.2 Investigação inicial para a construção de métodos de Compressão Sentencial desenvolvidos

Os métodos de Compressão Sentencial propostos neste trabalho baseiam-se na Abordagem por Deleção. Uma vez que foram investigados diferentes níveis de conhecimento linguístico para a construção desses métodos, optou-se por empregar apenas deleção de itens lexicais, visando uma comparação mais adequada entre as diferentes propostas.

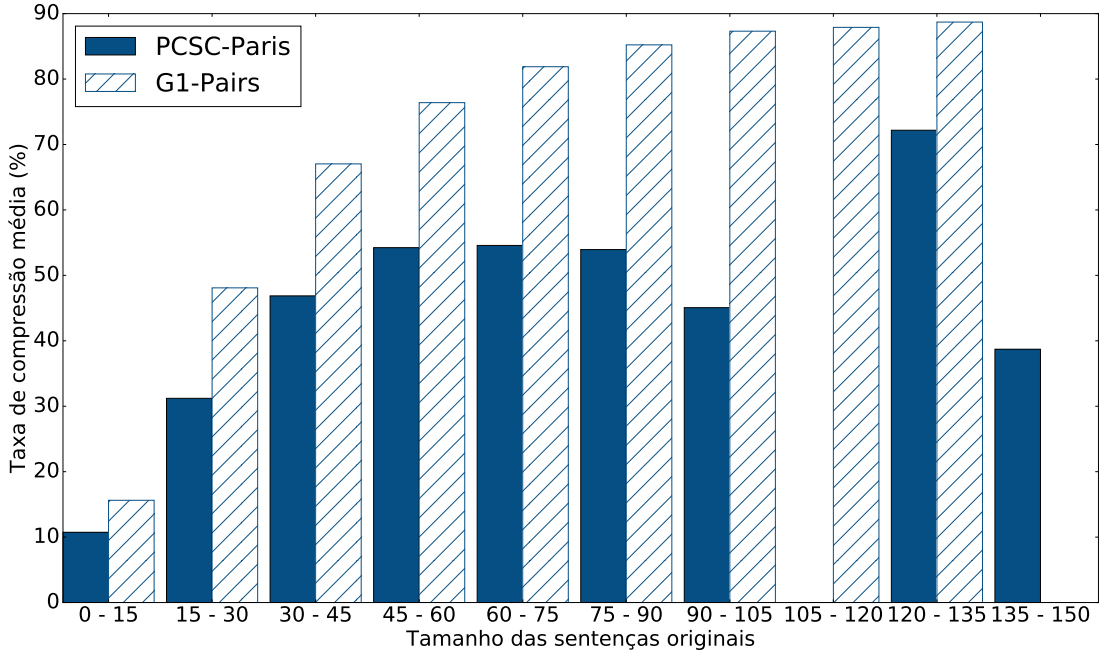

Figura 7 – Variação da taxa de compressão com relação ao tamanho das sentenças originais nos córpus Pares-PCSC e Pares-G1.

É importante ressaltar que métodos recentes do estado da arte em CS, como o sistema descrito em (FILIPPOVA *et al.*, 2015), seguem também essa metodologia.

Neste trabalho, a tarefa de Compressão Sentencial foi tratada por meio de técnicas distintas de Aprendizagem de Máquina para o problema tradicional de Classificação. Dessa forma, dada uma sentença de entrada s constituída por n itens lexicais (incluindo sinais de pontuação), a compressão ocorre por meio de uma sequência de n classificações, ou decisões, sobre s. Em cada decisão i, o método determina se o item lexical da posição i deve ser removido ou não da sentença sendo computada. Formalmente, dado que Γ_i indica a respectiva classificação para o item lexical na i-ésima posição em uma sentença s, e que essa classificação pode ser **Sim** ou **Não** para a seguinte pergunta "Esse item deve ser removido?" Uma sentença comprimida s_c de s será constituída por $\{t_i \in s \mid \Gamma_i = \text{Não}\}$. Por exemplo, no Quadro 16, é apresentado uma sentença original do córpus Pares-G1 e os respectivos rótulos de cada item lexical. Os rótulos estão demarcados entre < e >, tal que SIM indica que o item lexical deve ser removido da sentença, e NAO, o caso contrário.

Quadro 16 – Exemplo de uma sentença e os respectivos rótulos de cada item lexical.

s original:	Estão abertas as inscrições para um passeio ciclístico em Divinópolis .
s com rótulos:	Estão<SIM> abertas<NAO> as<SIM> inscrições<NAO> para<NAO> um<SIM> passeio<NAO> ciclístico<NAO> em<NAO> Divinópolis<NAO> . <NAO>
s_c final:	abertas inscrições para passeio ciclístico em Divinópolis.

Essa investigação inicial foi direcionada pelos seguintes questionamentos: 1) Qual

técnica de Aprendizado de Máquina é mais adequada para a tarefa de CS por meio da Abordagem de Deleção em que serão utilizados algoritmos de Classificação?; e 2) uma vez que a tarefa de Compressão Sentencial pode ser útil em diferentes aplicações, diferentes níveis de extração de atributos podem apresentar diferentes desempenho?. Dessa forma, somente o córpus Pares-PCSC foi utilizado nesse primeiro momento. O córpus Pares-G1 foi desconsiderado, pois representa apenas uma aplicação da CS, a produção automática de títulos. Por outro lado, tendo em vista que o córpus Pares-PCSC foi compilado a partir de outro conjunto de dados para a tarefa de Sumarização Compressiva, puderam ser investigados atributos em três diferentes contextos, ao nível sentencial, do documento e do sumário.

As questões que direcionaram esse experimento inicial, sobretudo a primeira, partem do princípio de que a tarefa de Compressão Sentencial pode auxiliar distintas aplicações em vários cenários, nos quais podem haver diferentes tipos de dados disponíveis que viabilizam a extração de atributos úteis. Por exemplo, para a construção de métodos de Sumarização Automática por meio da Síntese Compressiva, os atributos para um método de CS podem ser identificados a partir das sentenças e respectivos texto-fonte e sumários. Porém, em outras aplicações, há eventualmente apenas a sentença que deve ser comprimida.

Uma vez definido o córpus Pares-PCSC como ambiente de experimentação. Ressalta-se que esse recurso foi compilado a partir do córpus PCSC, que foi construído para a tarefa de SA, o que viabilizou a utilização de três níveis ou **conjuntos de atributos**: **Sentencial, Documento e Sumário**. Além disso, foi determinado que cada conjunto de atributos também inclui seus anteriores. Por exemplo, para a produção de sumários, usualmente, se tem acesso aos textos-fonte e as sentenças.

Independente do grupo de atributo utilizado, tendo em vista que o objetivo foi à construção de métodos de CS baseados na Deleção de itens lexicais, os atributos foram identificados para cada item lexical de uma sentença a ser processada. Em outras palavras, uma instância no conjunto de dados para treinamento e avaliação dos métodos propostos indica apenas um item lexical de alguma sentença no córpus. Assim, para dissertar sobre cada atributo proposto ao decorrer desta monografia, será utilizado *t* para identificar o item lexical alvo, ou seja, o item lexical da sentença de entrada para o qual os atributos estão sendo extraídos.

Os atributos ao nível da **Sentencial** foram extraídas por meio de informações diretamente relacionadas à sentença de entrada após a execução de um Parser sintático automático. Para tanto, utilizou-se do Parser PALAVRAS (BICK, 2000) para processar as sentenças de entrada e, posteriormente, extrair atributos organizados em 3 categorias, sendo elas: superficial, sintático e semântico.

Os atributos sentenciais considerados superficiais são aqueles que requerem apenas

uma análise simples da disposição dos itens lexicais na sentença de entrada, tal como se segue:

- inside-parentheses: um valor lógico que é verdadeiro caso *t* ocorra entre símbolos de parênteses "(" e ")", e falso caso contrário;
- token-position: que indica a posição em que *t* ocorre na sentença. Foram determinados três possíveis valores, "beggining", "ending" e "middle" para os casos respectivos em que *t* ocorre entre os 20% primeiros da sentença; após 80% dos itens da sentença ou entre os demais;
- is-stopword: um valor lógico que é verdadeiro quando *t*, necessariamente uma palavra, seja uma *stopword*;
- token-window: que corresponde à quarto atributos (*word-1, word-2, word+1, word+2*) que indicam a janela de palavras de tamanho 2 com tokens que circundam *t*;
- token-itself: o próprio item lexical sendo analisado.

Os atributos sentenciais das categorias sintática e semântica foram identificados por meio da análise da Árvore de Dependências Sintáticas obtida após a execução do Parser PALAVRAS (BICK, 2000), conforme descrito a seguir:

- token-POS: o rótulo de Part-of-Speech (POS) identificado para *t*;
- POS-window: as etiquetas de POS dos dois itens lexicais anteriores (pos-1, pos-2) e dos dois posteriores (pos+1, pos+2) em relação*t*;
- token-syntactic-function: a função sintática *t* na Árvore de Dependências Sintáticas;
- token-semantic: informação semântica *t* que é disponível pelo Parser empregado, tais como indicações de Entidade Nomeada e rótulos de classes semânticas.

Aqui, é importante ressaltar que a ferramenta PALAVRAS expande contrações presentes na língua Portuguesa, tais como: do = de + o, dele = de + ele, no = em + o). Dessa forma, visando obter as sentenças comprimidas como os mesmos tokens das sentenças originais (exceto àqueles descartados pelos métodos de compressão), foi utilizado um conjunto de regras para normalizar a saída da ferramenta e desconstruir essa etapa de normalização do Parser empregado neste trabalho. Além disso, o Parser PALAVRAS segmenta as sentenças de entrada por meio de alguns sinais de pontuação, que, não necessariamente delimitam quebras sentenciais, tais como os símbolos de ":" e ";". Dessa forma, para esses casos, optou-se por empregar a segmentação sugerida pelo Parser e, posteriormente à etapa de compressão, retornar à segmentação presente na sentença de entrada.

Como apresentado previamente, adotou-se uma abordagem de compressão baseada em uma sequência de deleções de itens lexicais sobre a sentença de entrada. Assim, assume-se que um decisão i do método é influenciada por decisões prévias e, que essa decisão, também influenciará as decisões posteriores. Dessa forma, além os atributos sentenciais supracitados, foram definidos também mais dois **atributos de Contexto**, que são valores lógicos, token-1-removed e token-2-removed, que indicam se o item lexical imediatamente anterior à t ou o anterior desse foram removidos ou não, respectivamente.

Todos os atributos supracitados que permitem três ou mais valores (*token-position, token-window, token-itself, token-POS, POS-window, token-syntactic-function, and token-semantic*) foram convertidos para um formato binário. Por exemplo, para o atributo *token-position*, que permite os valores *"beginning", "middle" or "ending"*, foram criadas três atributos binários, uma para cada possível valor.

Para os atributos que utilizam o próprio item lexical, que são as *token-window e token-itself*, foram utilizados somente os valores radicalizados, que foram identificados por meio da ferramenta RSLPStemmer[2] no pacote NLTK da linguagem Python. Além disso, para todos os valores numéricos desses atributos, foi utilizado o valor NUM. Dessa forma, a dimensionalidade desses atributos foi reduzida de forma significativa e, provavelmente, apresentou valores mais representativos. Por exemplo, provavelmente, para a tarefa de CS, é mais importante identificar que há algum número anteriormente a um item lexical alvo do que saber qual é esse número propriamente dito.

Como pode ser observado, alguns atributos propostos são referências a outros itens lexicais, tais como *token-window*, *POS-window* e os atributos de **Contexto**. Assim, em alguns cenários, os itens lexicais para esses atributos podem ser inválidos. Por exemplo, para o primeiro item lexical de uma sentença, não há itens anteriores, de forma que as referências para os dois itens anteriores não existe, o que faria com que a referência para os atributos de Contexto sejam inexistentes. Assim, para esses casos, definiu-se o valor padrão de $.

A seguir, será introduzidas os atributos relacionados ao nível do Documento e Sumário. Assim, é importante destacar que para um determinado item lexical t, esses atributos foram identificados somente a partir do respectivo Documento e Sumário relacionado à sentença em que t ocorre. Vale lembrar que o córpus Pares-PCSC é constituído de pares de sentença s e s_c, sendo que a primeira ocorre em algum Documento D do córpus, s_c está presente em algum sumário produzido a partir de D e é uma versão comprimida de s.

Por meio da análise do respectivo texto, ou *Documento*, da sentença a ser processada, foram propostos alguns atributos que visam identificar alguma informação de

[2] <http://www.nltk.org/_modules/nltk/stem/rslp.html>

relevância dos itens lexicais. Esses atributos baseiam-se no princípio de que a estrutura do documento pode sugerir a remoção de alguns itens lexicais. Por exemplo, sequências de itens lexicais muito frequentes podem ser simplificadas por meio da remoção de alguns itens da sequência, tal como a remoção do sobrenome de pessoas que foram citadas previamente no texto. Para tanto, foram propostas os seguintes atributos:

- sentence-position: um valor numérico que indica a posição da sentença no texto;
- in-gist-sent: um valor lógico, que é verdadeiro somente no caso em que o item lexical ocorre na sentença mais relevante do texto, que foi definida pela sentença que possui o maior valor na seguinte equação $\frac{1}{|s|} * \sum_{t_i \in s} freq_D(t_i)$, onde s é a sentença, $|s|$ seu respectivo tamanho, e $freq_D(t_i)$, é a frequência do token t_i no Documento. Em outras palavras, a sentença mais relevante é aquela que possui mais itens lexicais frequentes no Documento; e
- freq: um valor numérico que indica a frequência do item lexical t no respectivo documento.

Os **atributos Sumário** foram propostos baseando-se na ideia de que durante o procedimento de escrita de um sumário, as sentenças já produzidas podem interferir o processo de compressão. Por exemplo: se alguma sigla já foi previamente explicada, sua expansão pode ser desconsiderada; provavelmente, a primeira sentença do sumário deve ser menos comprimida do que as demais sentenças; etc. Os atributos para esse nível são apresentadas a seguir:

- in-previous: um valor lógico que é Verdadeiro caso o item lexical ocorre em alguma sentença previamente adicionada no sumário; e
- sent-sum-position: um valor numérico que indica a posição da sentença no sumário.

No Quadro 17, é apresentada uma sentença e os respectivos valores dos atributos para o item lexical “versões”, em negrito. Pode-se observar todas os valores (não transformados para formato binário) e respectivas referências a outros itens lexicais.

Para os conjuntos de atributos supracitados, foram investigados cinco (5) algoritmos de Aprendizagem de Máquina[3], sendo eles: Regressão Logística, *Suport Vector Machine* (SVM); Naïve Bayes, Árvore de Decisão e uma *Multi-Layer Perceptron* (MLP). Esses algoritmos foram selecionados porque se baseiam em diferentes abordagens de AM e são amplamente utilizados. Além desses algoritmos, foi também investigado um método Ensemble, que combina todos os algoritmos prévios por meio de votação.

[3] Foram utilizadas as implementações disponíveis no pacote Scikit-learn para a linguagem Python, que está disponível na seguinte url: <http://scikit-learn.org/stable/>.

Quadro 17 – Exemplo de item lexical com respectivos valores para cada atributo apresentado.

O iPad chega a Portugal em seis **versões** com o preço a variar entre 499 e 799 euros.	
inside-parentheses:	Não
token-position:	*Middle* (meio)
is-stopword:	Não
token-window:	“em”, “seis”, “com”, “o”
token-itself:	versões
token-POS:	Substantivo
POS-window:	Preposição, Substantivo, Preposição, Artigo
token-syntactic-function:	“P<”
token-semantic:	<vazio>
token-1-removed:	Sim
token-2-removed:	Sim

Nesses experimentos iniciais para a construções de sistemas de CS, os modelos propostos (algoritmo de Aprendizagem + atributos) foram avaliados conforme o ponto de vista da tarefa de Classificação. Em outras palavras, foi avaliada a qualidade de classificação de cada método com relação aos itens lexicais do córpus Pares-PCSC, mas não as respectivas sentenças comprimidas produzidas. Essa abordagem foi considerada suficiente para identificar qual o algoritmo de AM mais adequado para a tarefa de Compressão Sentencial baseada na Deleção de itens lexicais. Como métrica de avaliação, utilizaram os valores de Precisão, Cobertura e Medida F-1, que são comumente empregados na literatura. Ressalta-se que foi adotada a metodologia de *10-fold-cross-validation*, que é bastante usual no cenário da investigação de algoritmos de AM.

Na Tabela 5, são dispostos os resultados do experimento. Os diferentes níveis de atributos (Sentencial, Documento e Sumário) foram agrupados e são indicados na primeira coluna da tabela. Aqui, é importante ressaltar que os níveis são aglutinativos. Por exemplo, para os atributos do nível do Sumário, também são utilizados aqueles oriundos ao nível do Documento e da Sentença. Além disso, em todos os níveis, foram empregados os atributos de Contexto. Uma vez que se utilizou *10-fold-cross-validation*, os valores calculados correspondem à média dos dez *folds*. Por exemplo, o algoritmo Regressão Logística apresenta valor de F-1 igual a 0,88 por meio do uso dos atributos ao nível da Sentença.

Um primeiro achado relevante nessa primeira investigação, que pode ser observado na Tabela 5, é que não há diferença significante entre o uso de atributos de diferentes cenários. Isso não era esperado, pois o córpus utilizado como base para compilar os dados para experimentação foi desenvolvidos para a tarefa de Sumarização Automática. Contudo, esses resultados indicam que os atributos ao níveis sentencial, como informações sintáticas, superficiais e de contexto, são mais importantes e suficientes para a tarefa de Compressão Sentencial.

Além do fato supracitado, pode ser observado que o algoritmo de Regressão Logís-

Tabela 5 – Avaliação dos primeiros métodos de CS investigados usando *10-fold-cross-validation* no córpus Pares-PCSC.

Features	**Método**	**Precisão**	**Cobertura**	**F-1**
Sentença + Contexto	Árvore de Decisão	0.870	0.885	0.878
	Ensemble	0.736	0.783	0.759
	Regressão Logística	0.882	0.892	**0.887**
	Naïve Bayes	0.658	0.457	0.558
	MLP	0.729	0.764	0.746
	SVM	0.869	0.872	0.870
Documento + Sentença + Contexto	Árvore de Decisão	0.857	0.875	0.866
	Ensemble	0.855	0.880	0.867
	Regressão Logística	0.881	0.892	**0.887**
	Naïve Bayes	0.693	0.471	0.582
	MLP	0.774	0.793	0.784
	SVM	0.869	0.872	0.870
Sumário + Documento + Sentença + Contexto	Árvore de Decisão	0.859	0.855	0.857
	Ensemble	0.881	0.891	0.886
	Regressão Logística	0.882	0.892	**0.887**
	Naïve Bayes	0.693	0.471	0.582
	MLP	0.776	0.808	0.792
	SVM	0.869	0.872	0.870

tica apresentou os melhores resultados de F-1, praticamente, em todos as configurações de atributos adotados.

4.3 Aprimoramento dos modelos de Compressão Sentencial

A partir dos resultados dos experimentos apresentados na Seção anterior, optou-se por investigar novos conjuntos de atributos que são extraídos somente a partir da sentença de entrada, uma vez que a utilização de informações advindas do respectivo texto e sumário da sentença não apresentaram melhorias significantes. Essa restrição foi também de certa forma relevante, pois viabilizou o treinamento e avaliação dos novos modelos propostos sobre o córpus Pares-G1, que não possui sumários disponíveis. Além disso, somente o algoritmo de Regressão Logística, que obteve os melhores resultados nos experimentos prévios, será utilizado.

Para esta nova investigação, adotou-se a mesma arquitetura de compressão empregada nos experimentos anteriores. Em outras palavras, os novos métodos investigados também comprimem uma sentença s constituída por n itens lexicais por meio de decisões individuais sobre cada item lexical (portanto, n decisões) em s, que são determinadas por um Classificador.

Esses novos modelos foram desenvolvidos visando à redução de compressões incorretas produzidos na primeira investigação por meio de diferentes abordagens. Na primeira abordagem, assumiu-se que essa deficiência dos métodos foi ocasionada por ausência de mais informações sintáticas. Para tanto, fez-se também o uso do Parser PALAVRAS para extrair mais atributos a partir da estrutura da Árvore de Dependências Sintáticas. Além disso, com essa informação sintática disponível, foram investigadas duas formas de enumerar a ordem em que os itens lexicais da sentença alvo serão classificados, sendo uma sequencial (situação natural da leitura de uma sentença) e outra baseada nas relações de dependência, em que se inicia a classificação pelo item lexical identificado como raiz da árvore sintática e procede conforme as relações de dependência. Esses dois novos modelos serão apresentados na Seção 4.3.1.

Ressalta-se que os dois modelos supracitados fazem uso de uma ferramenta automática para identificar as relações sintáticas das sentenças de entrada, e que tal ferramenta pode também ser suscetível a erros. Assim, outra proposta de aprimoramento dos modelos baseou-se na possibilidade de que esses erros diminuíram a qualidade dos modelos desenvolvidos. Dessa forma, foi proposto um modelo Simplificado de Compressão, no qual são utilizados atributos advindos de um pré-processamento mais simples das sentenças. Nesse processamento, foi utilizado um Etiquetador Morfossintático, que por ser considerada uma análise mais simples (porém, mais veloz) em relação à identificação da Árvore Sintática, pode ser menos suscetível a erros e disponibilizar informações mais corretas para extração de atributos. Esse modelo **Simplificado de Compressão** será descrito na Seção 4.3.2.

Uma vez que a tarefa CS foi tratada como uma sequência de decisões individuais, que apresentam alta dependência com as decisões anteriores, e que uma dos atributos mais relevantes para a remoção de um determinado item lexical é a indicação se o item anterior foi ou não removido da sentença, uma resposta incorreta do classificador pode gerar erros em cascada. Aqui, é importante ressaltar que esse tipo de atributo é corretamente capturado durante a etapa de treinamento dos algoritmos. Porém, durante a compressão de novas sentenças, essa informação é oriunda de respostas prévias do classificador. Assim, com objetivo de reduzir a existência de erros em cascata dos classificadores propostos, foram também investigados dois algoritmos que tratam esse problema, o DAGGER (ROSS; GORDON; BAGNELL, 2011) e o *Conditional Random Fields* (CRF). Esses experimentos serão descritos na Seção 4.3.3 e 4.3.4.

Por fim, na Seção 4.3.5, serão apresentados os resultados de avaliação desses novos métodos de Compressão Sentencial propostos, bem como uma comparação com outros trabalhos da literatura.

4.3.1 Novos métodos de compressão baseados em árvore de dependência sintática

Nesta Seção, serão apresentados os dois modelos de Compressão propostos que fazem uso de novos atributos baseados na estrutura de uma Árvore de Dependências Sintáticas. Esse novo conjunto de atributos partem do princípio de que a análise da estrutura hierárquica presente nas relações de dependência sintática permite identificar meios para manter uma estrutura similar nas sentenças comprimidas. Por exemplo, se um determinado item lexical t_a foi removido da sentença, outro item t_b que possui dependência sintática para t_a provavelmente deve ser também removido.

Para identificação das Árvores de Dependências Sintáticas das sentenças, utilizou-se também o Parser Sintático PALAVRAS (BICK, 2000). Dessa forma, os mesmos procedimentos para normalização das sentenças utilizados no experimento descrito na Seção 4.2 foram também aplicados, como a contração das expansões apresentadas pela ferramenta (ex.: do = de + o) e correções na segmentação sentencial gerada pelo Parser PALAVRAS.

Partindo das ideias supracitadas, foram propostas os seguintes atributos baseados na **Estrutura Sintática**:

- token-is-root: um valor lógico que recebe o valor Verdadeiro caso o item lexical t seja identificado como raiz da árvore sintática, e Falso caso contrário;
- token-is-on-left: um valor lógico que recebe o valor Verdadeiro quando o item lexical t ocorre à esquerda de sua respectiva dependência sintática na árvore;
- token-father: que corresponde ao valor do item lexical que é a dependência sintática do item lexical alvo t;
- father-removed: um valor lógico que receber o valor Verdadeiro para a situação em que o item lexical que é a dependência sintática de t foi removido da sentença;

Além desses atributos, também foram utilizados todos aqueles propostos ao nível *Sentencial*, que foram apresentados na Seção 4.2. Ressalta-se que, de forma similar aos experimentos anteriores, foram também utilizadas algumas normalizações para os valores dos atributos. O caractere $ foi usado para indicar itens lexicais referenciados inválidos. Por exemplo, para um item lexical t_a indicado como raiz da árvore sintática, o atributo *token-father* não terá um valor válido, pois não há algum outro item com maior hierarquia. Dessa forma, para esse caso, esse atributo foi preenchida como o valor $. Além disso, o atributo *token-father* foi normalizado para uma atributo binária e todos os valores dos itens lexicais foram pré-processados por um Radicalizador, para diminuir a dimensionalidade do modelo.

Como introduzido anteriormente, tendo a Árvore de Dependência Sintática à disposição para extração desses atributos, foram também investigados os dois seguintes procedimentos para enumerar a ordem em que os itens lexicais devem ser classificados (removidos ou não): sequencial, conforme o processo tradicional de leitura de uma sentença; e baseado na respectiva Árvore Sintática de Dependência, no qual a classificação parte inicialmente do item lexical identificado como raiz da árvore e segue sucessivamente conforme as relações de dependência. Por exemplo, na Figura 18, é apresentada uma sentença e a respectiva ordem de classificação sequencial e baseada na respectiva árvore sintática. Como pode ser observado, na segunda ordem classificação, o primeiro item lexical lido (a ser classificado pelo método) é o verbo principal da sentença. Aqui, é importante ressaltar que o Parser PALAVRAS identifica qualquer sinal de pontuação como a raiz da árvore sintática. Assim, assumiu-se que esses itens lexicais são mais dependentes das palavras da sentença, de forma que se decidiu classificá-los somente após a classificação de todos os demais itens lexicais.

Quadro 18 – Exemplo de uma sentença extraída do córpus Pares-PCSC com os respectivos procedimentos de classificação sequencial e baseado na árvore de dependências sintáticas.

Leitura/classificação sequencial
O confronto de hoje será contra Cuba, às 22h. Os cubanos são considerados por Bernardinho como os rivais mais perigosos da fase classificatória.
Leitura/classificação baseada na árvore de dependências sintáticas
será confronto O de hoje contra Cuba às 22h às são cubanos Os considerados por Bernardinho como rivais os perigosos mais da fase da classificatória , . .

Uma vez que os modelos propostos possuem alguns atributos que dependem de respostas prévias do classificador, como os atributos *previous-word-removed, previous-word-2-removed e father-removed*, os dois procedimentos de enumeração dos itens lexicais podem apresentar situações em que alguma dessas informações estará indisponível. Por exemplo, na leitura sequencial, caso um item lexical ocorra previamente sua dependência sintática, o valor do atributo *father-removed* estará indisponível. Por outro lado, por meio da enumeração baseada na árvore sintática, um determinado item lexical pode ocorrer bem acima na hierarquia sintática em relação ao item lexical imediatamente anterior, de forma que o valor do atributo *previous-token-removed*. Assim, definiram-se os três possíveis valores para esses atributos: FALSO, que indica que o item referenciado já foi classificado como "Não deve ser removido"; VERDADEIRO, que é similar ao anterior, porém com o item referenciado classificado como "Deve ser removido"; INDEFINIDO, que indica que o item lexical é válido, porém não foi ainda classificado; e INVALIDO, para indicar que o item lexical não é válido (ex.: não há um item lexical anterior ao primeiro da sentença).

4.3.2 Um modelo de Compressão Sentencial simplificado

Esse novo modelo de Compressão Sentencial foi proposto tomando como princípio de que erros gerados por ferramentas automáticas de pré-processamento podem produzir erros que implicam em erros nos métodos propostos. Dessa forma, foi definido um conjunto de atributos que podem ser extraídos após a execução de somente um Etiquetador Morfossintático sobre as sentenças de entrada. Dessa forma, desconsiderou-se o uso do Parser PALAVRAS Bick (2000) para a construção desse modelo.

Previamente à extração dos atributos, foi executado o Etiquetar MXPOST (RATNAPARKHI, 1986), que foi treinado sob o córpus Mac-Morpho (AIRES, 2000; FONSECA; ROSA, 2013). Posteriormente, foram definidas as seguintes **Features Simplificadas**:

- token-position: que indica a posição em que *t* ocorre na sentença. Foram determinados três possíveis valores, “beggining”, “ending” e ”middle” para os casos respectivos em que *t* ocorre entre os 20% primeiros da sentença; após 80% dos itens da sentença ou entre os demais;
- is-stopword: um valor lógico que é verdadeiro quando *t*, necessariamente uma palavra, seja uma *stopword*;
- token-window: que corresponde à quarto atributos (*word-1, word-2, word+1, word+2*) que indicam a janela de palavras de tamanho 2 com tokens que circundam *t*;
- token-itself: o próprio item lexical sendo analisado.
- token-POS: que corresponde à etiqueta morfossintática identificada para o item lexical;
- POS-window: as etiquetas de morfossintáticas dos dois itens lexicais anteriores (pos-1, pos-2) e dos dois posteriores (pos+1, pos+2) em relação a *t*; e
- inside-parentheses: um valor lógico que é VERDADEIRO caso o item lexical ocorra entre os símbolos de parênteses;

De forma similar aos experimentos anteriores, os atributos apresentados foram normalizados para um formato binário. Para aqueles atributos que fazem uso dos próprios itens lexicais, utilizaram-se os valores radicalizados e NUM para indicar os valores numéricos. Além disso, as referências inválidas foram convertidas para o valor $.

4.3.3 Algoritmo DAGGER

O Algoritmo DAGGER, proposto em (ROSS; GORDON; BAGNELL, 2011), é um esquema iterativo que, de certa forma, visa “calibrar” classificadores treinados em cenários

de predição/classificação estruturada. Esse cenário ocorre quando a resposta desejada para alguma tarefa é produzida somente após uma sequência de classificações individuais e que, geralmente, possuem dependência entre si. Por exemplo, dada uma sentença de entrada, os métodos propostos para CS neste trabalho a comprimem após decidir individualmente a remoção ou não de cada item lexical.

No cenário supracitado, eventualmente, alguns atributos podem requere respostas prévias do próprio classificador. No caso específico dos métodos de CS propostos, foram definidas alguns atributos que indicam se algum item lexical prévio foi removido ou não da sentença sendo processada. Como apresentado anteriormente, esse tipo de atributo pode ser corretamente coletado durante a etapa de treinamento do classificador por meio do uso de informações advindas do conjunto de dados. Porém, após o treinamento, o classificador deverá usar as próprias respostas como valores para esses atributos. Assim, uma decisão errada do método, pode gerar mais erros em sequência.

Tendo em vista que o algoritmo DAGGER assume que o classificador será empregada em uma predição estruturada, o córpus para treinamento deve ser modelado dessa forma. Ressalta-se que as classificações continuarão ocorrendo conforme decisões individuais, porém, deve-se identificar a qual estrutura cada instância pertence. No caso da CS, deve-se identificar as respectivas sentenças em que cada item lexical ocorre.

Partindo da definição supracitada, o algoritmo DAGGER baseia-se no princípio de que as respostas do próprio classificador podem ser utilizadas durante o estágio de treinamento de forma iterativa. Dessa forma, espera-se que a possibilidade do classificador produzir erros devido a respostas prévias e incorretas utilizados com atributos possa ser reduzida. A seguir, é apresentado o pseudocódigo do algoritmo DAGGER.

Basicamente, dado um classificador inicial c, durante a execução do algoritmo DAGGER, novas instâncias para o conjunto de treinamento são produzidas por meio das saídas do próprio classificador. Porém, é importante observar que essas novas instâncias são geradas conforme um cenário de aplicação. Além disso, os erros produzidos pelo classificador também serão introduzidos no conjunto dados, o que podem ser interpretados como exemplos negativos pelo algoritmo de AM.

Para utilização do algoritmo DAGGER para calibrar os métodos de CS propostos (Seção 4.3.1 e 4.3.2), foi definido um número de iterações igual a 20. Ao final do processo, utilizou-se o modelo treinado na última iteração como método final.

4.3.4 Método baseados em um modelo de Conditional Random Fields

O Algoritmo *Conditional Random Fields* (CRF) é uma técnica de Aprendizagem de Máquina com foco em tarefas de Predição Estrutura. Ressalta-se que, como apresentado

Algoritmo 1 – Algoritmo DAGGER.

```
1: procedimento DAGGER(S,N)  ▷ Um conjunto de sentenças e o número total de iterações
2:     D ← Extrair atributos para cada sentença em S
3:     Treinar um classificador h_0 em D
4:     para i faça1N
5:         D_i ← ∅
6:         para todo sentença s ∈ S faça
7:             para todo item lexical t_j ∈ s faça
8:                 se i ≥ 2 então
9:                     c ← um número aleatório entre 0 e 1.0
10:                    se c ≤ 0.7 então
11:                        h_c ← h_{i−1}
12:                    senão
13:                        h_c ← h_{i−2}
14:                    fim se
15:                senão
16:                    h_c ← h_0
17:                fim se
18:                Classificar t_j usando h_c
19:                D_i ← D_i∪ Extrair atributos a partir de t_j
20:            fim para
21:        fim para
22:        D ← D∪D_i
23:    fim para
24:    retorne h_N
25: fim procedimento
```

anteriormente, a tarefa de CS pode ser tratada pela perspectiva da Predição Estrutura, em que a sentença comprimida é produzida após classificações/decisões individuais para cada item lexical dessa sentença.

Dado o cenário de aplicação do algoritmo CRF, o conjunto de dados para treinamento (e posteriormente novas instâncias que serão processadas) devem ser representadas em uma estrutura de linear (lista ou vetor), de forma que em cada posição é encontrada uma instância individual do problema. Dessa forma, o algoritmo CRF automaticamente assume que as classificações prévias influenciam nas decisões posteriores do classificado, além de também empregar os demais atributos explicitamente indicados.

Dada às características supracitadas, optou-se por desconsiderar o modelo de atributos introduzido da Seção 4.3.1 que enumera os itens lexicais por meio da hierarquia das relações de Dependências Sintáticas. Assim, foram treinados dois somente modelos de CRF, que, respectivamente, utilizam os atributos extraídos por meio da Árvore Sintática (modelo sequencial apresentado em 4.3.1) e por um pré-processamento simplificado da sentença de entrada (4.3.2). Dessa forma, ressalta-se que para o treinamento desses mo-

delos CRF, foram desconsiderados os atributos baseados na análise de algum item lexical que foi removido previamente, tais como *previous-token-removed* e *father-removed*).

4.3.5 Avaliação dos métodos de Compressão Sentencial

Nesta Seção, será apresentada a avaliação de todos os métodos que foram descritos na Seção 4.3, que correspondem às propostas de aprimoramentos dos métodos de CS inicialmente desenvolvidos.

Foram utilizados os dois córpus para CS introduzidos neste Capítulo, o Pares-PCSC e Pares-G1. Contudo, é importante ressaltar que esses córpus possuem sentenças comprimidas que foram geradas de formas distintas. O Pares-PCSC foi compilado a partir de outro recurso desenvolvido para investigações em Sumarização Automática. Por outro lado, no córpus Pares-G1, as centenas comprimidas são títulos de textos jornalísticos. Além dessas características distintas, a diferença entre a quantidade de pares de sentenças nesses dois recursos é muito discrepante, 874 e 7024, respectivamente.

Tendo em vista as características inerentes de cada córpus, as avaliações foram direcionadas por meio de quatro diferentes configurações desses recursos. Nas duas primeiras, cada córpus foi utilizado individualmente para treinamento e teste. Na terceira, foram utilizados os córpus em conjunto. Por fim, os modelos foram treinados no córpus Pares-G1, que possui maior número de pares de sentenças, e avaliados no Pares-PCSC. Essa última configuração foi importante para viabilizar a comparação dos modelos propostos o sistema baseado em *Deep Learning* introduzido em (FILIPPOVA *et al.*, 2015), que requer uma maior quantidade de dados para treinamento.

Exceto para a última configuração de córpus supracitada, em que se faz um uso de um córpus para treino e outro para teste, os demais experimentos foram direcionados por meio da tradicional metodologia *10-fold-cross-validation*. Dessa forma, para as três primeiras configurações (Pares-PCSC, Pares-G1, e Pares-PCSC + Pares-G1), os resultados reportados correspondem ao valor médio entre todos os 10 *folds*. Aqui, é importante ressaltar que, embora uma instância isolada de treinamento para os métodos propostos corresponde a um único item lexical, os *folds* foram construídos por meio das sentenças. Dessa forma, para toda sentença nos córpus, os respectivos itens lexicais são dispostos em um mesmo *fold*. Em outras palavras, se a instância de algum item lexical de alguma sentença s está em um *fold* de treinamento, todos os demais itens lexicais de s também estão dispostos nesse mesmo *fold* de treinamento.

Independentemente da configuração utilizadas, os resultados foram computados sob duas perspectivas diferentes. Na primeira, seguindo o procedimento comum na tarefa de Classificação em Aprendizagem de Máquina, avaliou-se a qualidade dos métodos na classificação de cada item lexical, desconsiderando-se a estrutura da sentença nesse mo-

mento. Como métricas para essa perspectiva, foram utilizadas as métricas tradicionais de **Precisão**, **Cobertura** e Medida-F (**F-1**), que são amplamente empregadas na literatura.

Na segunda perspectiva de avaliação, foi comparado a sentença comprimida por cada método com a respectiva sentença comprimida presente no córpus (*gold-standart*), considerando assim, a estrutura da sentença gerada pelos métodos. Como métrica, primeiramente foi computado o **Erro** produzido por cada método em cada sentença, que indica o percentual de itens lexicais erroneamente classificados (removidos ou não) por sentença. Por fim, calculou-se o **Erro Médio** de cada método. Dessa forma, é possível estimar a qualidade das sentenças produzidas pelos métodos de CS propostos.

Para organizar os resultados, serão utilizados alguns padrões para identificar cada método proposto e avaliado, como se seguem:

- **Mod. Sintático**: indica os métodos apresentados na Seção 4.3.1, em que se faz a análise da Árvore de Dependência Sintática para extrair *atributos* mais informativas;
- **+TREE**: é adicionado à etiqueta Mod. Sintático, para indicar quando ordem de classificação dos itens lexicais de cada sentença foi executado conforme a estrutura sintática, iniciando o processo pelo item lexical demarcado como a raiz (um exemplo é apresentado na Figura 18). Caso essa etiqueta não seja utilizada, significa que os itens lexicais foram classificados conforme a ordem sequencial natural;
- **Mod. Simples**: indica o método proposto que faz uso de um pré-processamento mais simples das sentenças de entrada (Seção 4.3.2);
- **+DAGGER**: será adicionado à etiqueta do modelo para indicar a utilização do algoritmo DAGGER para reduzir a possibilidade de erros em cascata; e
- **+CRF**: análogo ao anterior, porém indica a aplicação do algoritmo CRF.

Na Tabela 6, são apresentados os resultados para todas as configurações em que se usou a metodologia de *10-fold-cross-validation*. Nas linhas, são dispostos cada método avaliado e, nas colunas, as métricas de avaliação utilizadas. Por exemplo, o método Mod. Simples +CRF possui valores de Precisão e Erro Médio iguais a X e Y, respectivamente, no córpus Pares-PCSC. Para uma melhor organização dos resultados, os métodos foram ordenados conforme o valor de F-1, do maior para o menor, e para as demais métricas fez-se uso de marcações em negrito para destacar valores relevantes.

Como esses resultados, pode-se observar que, do ponto de vista do Aprendizado de Máquina, o método que faz uso dos atributos baseados na Árvore de Dependências Sintáticas por meio da enumeração dos itens lexicais baseadas nessa estrutura sintática apresentou os melhores resultados. Isso demonstra que o uso atributos mais informados são muito relevantes para a tarefa de Compressão Sentencial. Além disso, tem-se também

Tabela 6 – Resultados para as avaliações que utilizaram **10-fold-cross-validation**

Método	Precisão	Cobertura	F-1	Erro Médio
Pares-PCSC				
Mod. Sintático	0.92	0.92	0.92	0.08
Mod. Sintático+CRF	0.87	0.87	0.87	0.12
Mod. Sintático+DAGGER	0.91	0.91	0.91	0.08
Mod. Sintático+Tree	0.93	0.93	0.93	0.07
Mod. Sintático+ Tree+ DAGGER	0.88	0.88	0.88	0.11
Mod. Simples	0.90	0.90	0.90	0.10
Mod. Simples + CRF	0.94	0.94	0.94	0.06
Mod. Simples + DAGGER	0.87	0.87	0.87	0.13
Pares-G1				
Mod. Sintático	0.90	0.90	0.90	0.09
Mod. Sintático + CRF	0.91	0.91	0.91	0.09
Mod. Sintático + DAGGER	0.76	0.76	0.76	0.25
Mod. Sintático + Tree	0.90	0.90	0.90	0.10
Mod. Sintático + Tree + DAGGER	0.81	0.83	0.82	0.21
Mod. Simples	0.89	0.89	0.89	0.11
Mod. Simples + CRF	0.81	0.81	0.81	0.19
Mod. Simples + DAGGER	0.69	0.69	0.69	0.32
Pares-PCSC + Pares-G1				
Mod. Sintático	0.89	0.89	0.89	0.10
Mod. Sintático+ CRF	0.90	0.89	0.90	0.10
Mod. Sintático+ DAGGER	0.76	0.79	0.78	0.21
Mod. Sintático+Tree	0.89	0.89	0.89	0.11
Mod. Sintático+Tree+ DAGGER	0.81	0.82	0.81	0.15
Mod. Simples	0.88	0.88	0.88	0.11
Mod. Simples + CRF	0.92	0.90	0.91	0.12
Mod. Simples + DAGGER	0.69	0.69	0.69	0.32

que a extração de atributos e classificação dos itens lexicais por meio da ordem indicada pela Árvore Sintática é promissora.

O método Mod. Simples, que requer um pré-processamento simples da sentenças, apresenta uma taxa de Erro Médio ligeiramente maior do que os demais métodos, além de também dispor valores similares para as demais métricas (Precisão, Cobertura e F-1). Esse achado sugere que informações simples do item lexical alvo e de seu contexto (ex.: etiqueta morfossintática dos demais itens lexicais próximos e a indicação se os itens lexicais anteriores foram removidos) é muito satisfatório para a classificação de um item lexical individualmente. Entretanto, provavelmente, esse tipo de abordagem pode produzir compressões com baixa gramaticalidade, pois a estrutura sintática pode não ser preservada. Resultados similares foram demonstrados em Filippova *et al.* (2015), em que o respectivo método proposto mais simples, no qual não há análise de alguma estrutura sintática, apresentou resultados muito. Porém, o melhor modelo proposto por tais autores incorpora atributos extraídos a partir da Árvore de Dependências Sintáticas.

Os algoritmos DAGGER e CRF não demonstraram melhorias significantes em relação aos demais métodos. Por exemplo, o modelo CRF treinado com os atributos empregados no Mod. Simples, que foi referenciado como Mod. Simples+CRF, só apresenta um menor Erro Médio no córpus Pares-PCSC. Esse comportamento era, de certa forma, esperado nesse momento, pois se extraiu os valores para os atributos a partir do próprio córpus e essas algoritmos visam diminuir a possibilidade de erros em cascadas ocasionados pelo uso de atributos cujo valores advém de classificações prévias do classificador. Além disso, é possível também observar que o algoritmo CRF foi ligeiramente melhor do que o uso do DAGGER.

Outro fato relevante é que não há muita diferença entre os resultados para um mesmo método nos três diferentes cenários de experimento, Pares-PCSC, Pares-G1 e Pares-PCSC + Pares-G1. Acredita-se que isso ocorreu pois os atributos propostos podem representar diferentes aplicações da CS (SA Compressiva e Produção automática de títulos). Além disso, ressalta-se que na primeira investigação de CS aplicada neste trabalho, cujo objetivo foi avaliar diferentes níveis de atributos (Sentença, Documento e Sumário), não foi encontrada diferenças significantes. Dessa forma, muito provavelmente, independentemente da aplicação do método de CS, os atributos mais importantes encontram-se na estrutura da sentença e não em informações de *background*, como o texto-fonte.

Na Tabela 7, é apresentado a configuração de experimentos em que os métodos foram treinados por meio do córpus Pares-G1 e testados no córpus Pares-PCSC. Essa configuração possibilitou a comparação dos métodos propostos com o trabalho apresentado em Filippova *et al.* (2015), que requer um volume maior de dados para treinamento. A disposição dessa Tabela é a mesma empregada nos resultados anteriores. Aqui, é importante informar que essa avaliação ocorreu conforme a aplicação dos métodos propostos no cenário real, em que os atributos que requerem classificações prévias do método, foram efetivamente preenchidas com as respectivas respostas do classificador.

Tabela 7 – Avaliação dos métodos por meio do treinamento no córpus Pares-PCSC e teste no córpus Pares-PCSC.

Método	**Precisão**	**Cobertura**	**F-1**	**Erro Médio**
Mod. Sintático+Tree	0.66	0.48	0.33	0.56
Mod. Sintático+Tree+ DAGGER	0.65	0.48	0.32	0.56
Mod. Sintático	0.86	0.85	0.85	0.15
Mod. Sintático+ DAGGER	0.85	0.84	0.84	0.16
Mod. Sintático+ CRF	0.85	0.84	0.84	0.16
Mod. Simples	0.62	0.57	0.55	0.43
Mod. Simples + DAGGER	0.63	0.57	0.54	0.44
Mod. Simples + CRF	0.86	0.85	0.85	0.15
FILIPPOVA *et al.*	0.84	0.84	0.84	0.16

Nesse cenário, é evidente que os resultados são relativamente inferiores do que os

valores reportados na Tabela 6. Como dissertado anteriormente, isso ocorre devido aos erros em cascadas, que eventualmente ocorrem após uma classificação incorreta. Além disso, pode-se observar que os modelos treinados por meio do CRF e os modelos que foram calibrados pelo DAGGER são ligeiramente superiores aos demais.

De forma distinta ao reportado na primeira avaliação, o Mod. Sintático+Tree não superou os demais sistemas avaliados. Esse cenário indica que, embora esses atributos sejam relevantes, a enumeração e classificação dos itens lexicais das sentença conforme a ordem indicada pela Árvore de Dependência Sintáticas é mais suscetível aos erros em cascada. De certa forma, esse fenômeno pode ser considerado intuitivo, pois uma decisão errônea para um determinado item lexical localizado em posições superiores da árvore sintática, pode corromper toda a estrutura sintática abaixo.

O método que requer um pré-processamento simples das sentenças, o Mod. Simples, apresentou uma taxa de Erro Médio ligeiramente melhor do que os demais modelos que não fazem uso do algoritmo DAGGER ou CRF. Isso demonstra que, mesmo aplicando atributos que requerem decisões prévias do classificar e que tal cenário pode induzir erros em cascada, erros advindos do pré-processamento das sentenças também influenciam diretamente nos erros produzidos pelos modelos propostos, visto que os métodos Mod. Sintático e Mod. Sintático+TREE, requerem um pré-processamento mais complexo das sentenças.

Além das discussões supracitadas, é também importante observar que nossos modelos baseados em informações sintáticas apresentaram resultados muito próximos aos atingidos pelo métodos proposto em (FILIPPOVA *et al.*, 2015), que faz uso de técnicas de *Deep Learning*. Em algumas ocasiões, inclusive, os métodos propostos apresentam resultados superiores. Contudo, é importante também ressaltar que técnicas de *Deep Learning* são mais sensíveis à quantidade de dados para treinamento e que, nesse trabalho, esse quantidade foi praticamente a metade do volume de dados utilizados para treinamento do modelo original para a língua Inglesa, que foi reportado pelos autores FILIPPOVA *et al.*.

No Quadro 19, são dispostas alguns sentenças do córpus Pares-PCSC que foram comprimidas pelos métodos após o treinamento no córpus Pares-G1. Para comparação, também são apresentadas as respectivas sentenças originais e compressões feitas por humanos.

Como pode ser observado pelos exemplos no Quadro 19, os modelos sintáticos (Mod. Sintático) com ou sem o algoritmo DAGGER apresentam sentenças comprimidas muito distantes das sentenças geradas manualmente. Em geral, esses modelos identificaram e mantiveram sintagmas relevantes na sentença, porém removeram alguns itens lexicais centrais para a composição do significado principal da sentença. Por exemplo, para o primeiro caso, o Mod. Sintático, com e sem o DAGGER, produziu uma sentença

Quadro 19 – Exemplo de sentenças do córpus Pares-PCSC e respectivas versões comprimidas por humanos e pelos métodos propostos.

Original:	O plano estabelece um empréstimo no valor total de 78 mil milhões de euros até Junho de 2013 .
Humano:	O plano estabelece um empréstimo de 78 mil milhões de euros até 2013 .
Mod. Sintático:	de até Junho de 2013
Mod. Sintático DAGGER:	de até Junho de 2013
Mod. Sintático +CRF:	plano estabelece empréstimo no valor total de até 2013 .
Mod. Simplificado+ DAGGER:	plano estabelece .
Mod. Simplificado:	plano estabelece empréstimo no valor 78 mil milhões de euros .
Filippova *et al.* (2015):	plano estabelece um empréstimo no 78 mil milhões de euros até .
Mod. Simplificado +CRF:	plano estabelece 78 mil milhões .
Original:	Os Estados Unidos vão começar a armar os rebeldes sírios , marcando assim um crescendo do envolvimento norte-americano no conflito .
Humano:	Os Estados Unidos vão começar a armar rebeldes sírios .
Mod. Sintático:	vão começar a armar rebeldes sírios .
Mod. Sintático DAGGER:	vão começar a armar rebeldes sírios , marcando crescendo .
Mod. Sintático +CRF:	Estados Unidos vão começar rebeldes sírios .
Mod. Simplificado+ DAGGER:	Estados Unidos vão começar a armar rebeldes sírios .
Mod. Simplificado:	Estados Unidos vão começar a armar rebeldes sírios crescendo do envolvimento norte-americano no conflito .
Filippova *et al.* (2015):	Estados Unidos vão começar a armar os rebeldes sírios ,
Mod. Simplificado +CRF:	Estados Unidos vão começar a armar rebeldes crescendo .

apenas com a data do acontecimento e desconsiderou o evento principal da sentença original. Ressalta-se que um efeito similar também pode ser observado no Mod. Simplificado +DAGGER.

Os métodos baseados em modelos de CRF apresentam boas sentenças (usualmente) de forma similar ao trabalho de Filippova *et al.* (2015). Contudo, pode-se verificar alguns erros, sobreduto, no final da sentenças. Esses erros podem ter sido gerados pois, no processo de compressão, foram consideras as sentenças de forma integral, de forma que para uma dada sentença de entradas, os métodos deveriam classificar todos os itens lexicais

em uma única sequência. Assim, sentenças com dois ou mais trechos (sintagmas) significativos poderiam apresentar regiões de transição de conteúdo que dificulta a classificação adequada por parte dos métodos. Por exemplo, no segundo caso apresentado, pode-se considerar que a sentença original possui uma informação (trecho) principal e um secundário, após a demarcação de vírgula. Dado isso, observa-se que os métodos Mod. Simplificado +CRF e Filippova *et al.* (2015) produziram sentenças comprimidas muito satisfatórias. Entretanto, o Mod. Simplificado +CRF gera uma compressão com vestígios do trecho de conteúdo secundário. Uma possibilidade para contornar esse problema seria treinar/aplicar os métodos de compressão sobre segmentos previamente delimitados das sentenças de entradas.

No Quadro 20, são apresentadas mais três sentenças e as respectivas versões comprimidas que foram produzidos pelo métodos Mod. Sintático +CRF. Nesses casos, pode-se perceber três situações bem distintas. Na primeira, a sentença produzida pelo método possui uma boa gramaticalidade e mantém o significado principal da sentenças original. No segundo caso, entretanto, embora a versão comprimida seja bem formada, o sentido principal da sentença original está vago ou incorreto. Por fim, o último exemplo é constituído por uma sentença com pouca qualidade gramatical.

Quadro 20 – Exemplos de sentenças comprimidas com diferentes níveis de qualidade que foram geradas pelo método Mod. Sintático +CRF.

Original:	A decisão da Mesa Diretora do Senado sairá hoje, às 11h
Compressão:	A decisão Mesa Diretora do Senado sairá às 11h.
Original:	Kaká fez excelente jogada na direita e virou o jogo para Robinho na esquerda.
Compressão:	Kaká fez jogada na direita e o Robinho na esquerda .
Original:	Ainda aos quatro minutos, Júlio Baptista dominou dentro da área, driblou Ayala e chutou forte, marcando um golaço.
Compressão:	Ainda aos, Júlio Baptista dominou da área, Ayala, um golaço .

CAPÍTULO

5

MÉTODOS DE SUMARIZAÇÃO DE ATUALIZAÇÃO INVESTIGADOS

Neste capítulo, serão apresentados os métodos de Sumarização Automática de Atualização (SAA) investigados e propostos neste trabalho, bem como a metodologia e resultados de avaliação por meio da abordagem de síntese Extrativa e Compressiva.

Para avaliar as hipóteses e questões de pesquisa deste trabalho, foram selecionados os métodos mais representativos de diferentes abordagens para a tarefa de SAA. Dessa forma, pôde-se analisar o impacto do uso de conhecimento semântico e discursivo em diferentes perspectivas para tratar os desafios inerentes à SAA, tais como métodos baseado em Ranqueamento Sentencial, Algoritmos de grafos e Modelos Gerativos. A descrição detalhada de cada método e as respectivas propostas para inserção de conhecimento linguístico serão descritas na Seção 5.1. Posteriormente, na Seção 5.2, será apresentada a Arquitetura para Sumarização Compressiva proposta neste trabalho, bem como as possíveis configurações investigadas.

Para realização de experimentos e avaliação dos métodos de SAA para a língua Portuguesa, foi preparado o córpus CSTNews-Update, que é constituído por uma diferente estruturação do córpus CSTNews (ALEIXO; PARDO, 2008; CARDOSO *et al.*, 2011). Esse recurso foi necessário, pois não se encontrou algum córpus destinado à tarefa de SAA para esse idioma. Na Seção 5.3, será apresentada a metodologia empregada para preparar o CSTNews-Update e os detalhes mais relevantes de suas características.

Na seção 5.4, serão descritos a metodologia e os resultados de avaliação dos métodos de SAA Extrativos e Compressivos que foram investigados. Além disso, também serão apresentados alguns experimentos para a Língua Inglesa, em que os córpus empregados são mais consolidados e amplamente empregados para avaliação de métodos de SAA.

5.1 Métodos de SAA investigados

Nesta Seção, serão descritas todas as abordagens de SAA investigadas neste trabalho. Os métodos que serão apresentados correspondem aos mais representativos de suas respectivas abordagens de SAA. Além disso, para cada método, também será introduzida a proposta para incorporar conhecimento linguístico, visando à produção de sumários mais informativos.

Todos os métodos investigados foram submetidos à tarefa de SAA, em que, para cada coleção textual C, há dois grupos de textos, referenciadas como A_C e B_C, de forma que um sumário de atualização deve ser produzido a partir dos textos da coleção B_C admitindo-se que o leitor possui conhecimento prévio dos textos da coleção A_C. Dessa forma, distintamente das competições da TAC 2008 e 2009, não foram consideradas restrições extra-textuais, como definição de uma pergunta do usuário/leitor, ou categorização de aspectos textuais. Na Figura 8, é ilustrado o cenário em que os métodos de SAA foram submetidos.

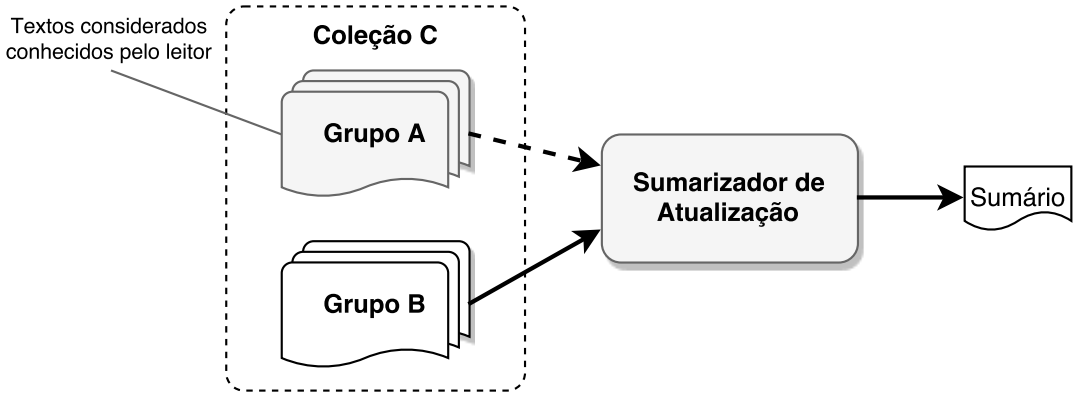

Figura 8 – Cenário de experimentação dos métodos de SAA.

Para uma melhor organização do texto, os métodos serão ordenados conforme a complexidade de processamento computacional exigido. Assim, primeiramente, serão apresentados dois métodos que fazem uso de somente informações posicionais para ranquear as sentenças relevantes nas Seções 5.1.1 e 5.1.2. Na Seção 5.1.3, serão descritos os métodos baseados na métrica de Fator de Novidade (NF). Na Seção 5.1.4, serão apresentados algoritmos baseados na estrutura de grafos que são empregados na SAA. Por fim, nas seções 5.1.5 e 5.1.6, serão descritos os métodos baseados em modelos gerativos de n-gramas e distribuição de tópicos, respectivamente.

Algumas abordagens de SAA investigadas neste trabalho fazem uso de uma análise de vocabulário presente nos textos-fonte. Comumente, esses vocabulários são identificados por meio de procedimentos superficiais de delimitação dos itens lexicais (tokenização) com posterior filtragem (ex.: remoção de *stopwords* e sinais de pontuação) e normalização dos itens remanescentes (radicalização ou lematização). Dessa forma, algumas informações

relevantes presentes nessa estrutura superficial das sentenças, como a presença de cadeias que referenciam Entidades Nomeadas (EN), são desconsideradas. Dessa forma, visando a aplicação de técnicas mais linguisticamente informadas de SAA, investigou-se também a delimitação de entidades nomeadas durante o procedimento de análise de vocabulário frequentemente empregada na tarefa de sumarização. Essa abordagem será descrita na Seção 5.1.8.

Como visto anteriormente, foram investigadas técnicas distintas de representação textual e seleção de conteúdo no contexto da SAA. Comumente, em aplicações de Aprendizado de Máquina, tem-se algumas técnicas que partem do princípio de que a variabilidade de algoritmos ou modelos podem ser combinadas por meio de um método Ensemble visando a melhoria dos resultados. Tendo em vista que, além das distintas abordagens de representação textual, os métodos de SAA investigados neste trabalho requerem diferentes níveis de processamento computacional, foi proposto um método Ensemble que agrega os métodos que, do ponto de vista computacional, são relativamente menos complexos. Essa decisão pautou-se na hipótese de combinar métodos mais simples pode produzir sumários mais informativos em relação aos métodos mais complexos, como o DualSum. Esse método Ensemble será descrito na Seção 5.1.7.

Alguns métodos de SAA que foram investigados neste trabalho não possuem procedimentos para identificação e remoção de conteúdo redundante nos sumários produzidos. Assim, visando a criação de sumários mais informativos e sucintos, foi adotado o mesmo procedimento descrito por Ribaldo *et al.* (2012) para tratar o fenômeno da redundância nos sumários. Esse método será apresentado na Seção 5.1.9.

Ressalta-se que as diferentes técnicas de representação textual para SAA, para as quais foram propostas maneiras para incorporar algum conhecimento linguístico neste trabalho, serão descritas em maiores detalhes durante essa seção. Dessa forma, acredita-se que o entendimento por parte do leitor será facilitado. Além disso, é importante salientar que, tendo em vista que relações discursivas modeladas por meio da CST, comumente, são computacionalmente representadas por meio de grafos, optou-se por incorporar esse conhecimento linguístico em apenas métodos baseados nessa estrutura de dados.

5.1.1 Características posicionais

Ouyang *et al.* (2010) propuseram métricas de ranqueamento sentencial baseados somente em características posicionais das sentenças e/ou respectivos itens lexicais. Segundo os autores, essas métricas partem do princípio de que a primeira ocorrência de algum componente textual (parágrafo, sentença, item lexical, etc.) em um texto possui maior relevância dos que as demais.

Adotando a premissa anterior, Ouyang *et al.* (2010) introduziram quatro métricas

que computam distintamente a relevância de um elemento textual por meio de sua respectiva posição no texto. Essas métricas são apresentadas a seguir, em que n indica o número de elementos considerados no texto; i representa a respectiva posição do elemento sendo avaliado; e λ foi definido como 0, uma vez que os autores originais sugeriram apenas a utilização de um valor numérico positivo pequeno:

- **Proporção Direta**: $f(i) = (n-i+1)/n$;
- **Proporção Inversa**: $f(i) = 1/i$;
- **Proporção Geométrica**: $f(i) = (1/2)^{i-1}$;
- **Proporção Binária**: $f(i) = 1$ *se* $i = 1$ senão λ

Pode-se observar que as equações supracitadas assumem uma pontuação maior para a primeira ocorrência de um elemento textual e decrementam esse valor, de forma mais suave ou abrupta, para as demais ocorrências. Por exemplo, em um texto-fonte com 10 sentenças, os valores de Proporção Direta para a primeira e terceira sentenças desse texto serão respectivamente de 1,0 $((10-1+1)/10)$ e 0,8 $((10-3+1)/10)$.

Ouyang *et al.* (2010) afirmam que essas características posicionais podem ser definidas por meio de distintos elementos textuais, como sentenças e palavras. No escopo da SA, sobretudo baseando-se na síntese Extrativa, é intuitivo aplicar essas métricas para ranquear sentenças por meio de suas respectivas posições nos textos-fonte, assim como exemplificado anteriormente. Contudo, para estender essa abordagem para computar a relevância posicional das palavras que compõem um texto, assume-se que i é a respectiva ocorrência da palavra sendo avaliada e n é a quantidade de vezes em que essa palavra ocorreu no texto. Com essa definição, percebe-se que palavras com uma única ocorrência no texto apresentam uma pontuação alta. Por exemplo, por meio da Proporção Direta, tem-se computado o valor 1, pois $(n-i+1)/n = (1-1+1)/1$. Assim, para ponderar casos similares, pode-se considerar também a frequência de ocorrência de uma palavra, de forma que a primeira ocorrência de uma palavra muito frequente no texto-fonte terá uma pontuação superior em relação à primeira ocorrência de alguma outra palavra pouco frequente.

Tendo em vista que as características posicionais das palavras que compõem uma sentença s podem influenciar em sua relevância, Ouyang *et al.* (2010) propuseram experimentos para identificar a relevância r de um sentença s por meio da seguinte equação $r_{lexical}(s) = \frac{1}{|s|} * \sum_{w^i \in s} f(w_s^i)$, em que w^i é uma palavra em s. Dessa forma, o valor calculado para a sentença s é definido por meio da média das pontuações de suas palavras. Assim, mesmo que uma sentença seja a primeira em algum texto, seu valor posicional de relevância pode não ser o maior, pois alguma outra sentença pode conter palavras mais

frequentes e com características posicionais mais relevantes. Além disso, Ouyang *et al.* (2010) consideraram que as características posicionais de uma sentença e de suas palavras são complementares, de forma que a relevância de uma sentença s pode ser calculada por meio dessas duas informações, com que $r(s) = f(s) + r_{lexical}(s)$.

Como descrito anteriormente, por meio da abordagem de SAA de (OUYANG *et al.*, 2010), é possível computar a relevância de uma sentença por meio das características posicionais de diferentes elementos textuais. Dessa forma, dado os objetivos desta tese, propôs-se a análise da relevância posicional dos subtópicos presentes em um texto com objetivo de aprimorar o cálculo da relevância posicional de uma sentença. Para tanto, definiu-se que a relevância posicional de um subtópico ***Sub*** pode ser computada por meio das características posicionais dos itens lexicais presentes em suas sentenças. Essa decisão foi tomada porque se assumiu que a utilização direta da posição do ***Sub*** pode desconsiderar um conhecimento importante sobre a distribuição das ideias presentes em um texto. Por exemplo, um subtópico que ocorre em uma posição intermediária do texto pode introduzir outro conteúdo relevante, que não está presente nos respectivos primeiros subtópicos. Dessa forma, considerar as características posicionais dos itens lexicais em um subtópico pode modelar de forma mais eficiente esse cenário. Com essa intuição, pode-se computador a relevância r_{sub} de um ***Sub*** de forma muito similar ao cálculo da relevância $r_{lexical}(s)$ para uma sentença, que foi definida anteriormente, por meio da equação $r_{sub}(s) = \frac{1}{|Sub|} * \sum_{w^i \in Sub} f(w^i_{Sub})$, em que $|Sub|$ indica a quantidade de itens lexicais presentes em ***Sub***, e $w^i \in Sub$ são todos os itens lexicais presentes no subtópico sendo avaliado.

Com as equações supracitadas, foram realizados experimentos em que a relevância de uma sentença é dada pela soma ponderada $r(s) = \alpha\ r_{sub}(Sub_s) + (1-\alpha) r(s)$, em que $r_{sub}(Sub_s)$ é a relevância posicional computada para o subtópico em que s ocorre, α é um parâmetro do algoritmo e $r(s)$ é o valor de importância definido para a sentença por meio das características posicionais dos itens lexicais. Nos experimentos realizados, o melhor valor de α encontrado foi de 0,6. Isso indica que as características posicionais dos subtópicos contribuem para essa abordagem de SAA.

No decorrer deste texto, esse método será referenciado como **Posição F**, em que F será substituído por Direta, Inversa, Geométrica ou Binária conforme a função de ranqueamento posicional utilizado. Para indicar a utilização do enriquecimento dessa abordagem por meio dos subtópicos, será adotada a marca **+SUB**.

5.1.2 Ranqueamento Posicional Ótimo (OPP)

Katragadda, Pingali e Varma (2009) propuseram um método de SA em que a relevância de uma determinada sentença é definida por meio de um modelo de Ranqueamento Posicional previamente treinado, que os autores referenciam como OPP (do inglês, *Optimal Position Policy*).

Para treinar um modelo de OPP, os autores fizeram uso de um córpus com coleções textuais e respectivos sumários para os quais foram manualmente identificados EDUs definidas pela métrica da Pirâmide (NENKOVA; PASSONNEAU, 2004) (mais detalhes dessa métrica foram apresentados no Capítulo 2). Com esse cenário, Katragadda, Pingali e Varma (2009) definiriam que a quantidade de EDUs presentes em uma sentença como sua respectiva relevância. Posteriormente, computou-se a importância de cada posição sentencial i por meio da média dos valores de relevância de todas as sentenças que ocorrem na posição i em algum texto do córpus. É importante ressaltar que, dada a discrepância no tamanho dos textos, os autores classificaram-nos entre pequenos ou grandes e normalizaram o resultado de forma distinta para cada categoria.

Segundo esses autores, embora simples, tal abordagem é um sólido *baseline* para a tarefa de SAA, que produz sumários mais informativos do que apenas selecionar as 100 primeiras palavras do texto mais recente, que foi definido como método *baseline* na DUC 2007 e TAC 2008.

Neste trabalho, foram treinados dois modelos OPP por meio do córpus CSTNews, o **OPP-Freq** e **OPP-Alinhamento**. A diferença desses modelos para a proposta original de Katragadda, Pingali e Varma (2009) é a forma com que as sentenças são inicialmente valoradas para, posteriormente, identificar o valor de OPP para cada posição sentencial.

Por meio do OPP-Freq, cada sentença s de algum texto em uma coleção C foi ponderada por meio da média da frequência de suas respectivas palavras na coleção C. Formalmente, tem-se que o valor considerado para uma sentença é dado por $\frac{1}{|s|}\sum_{w \in s} freq(w,C)$, em que $|s|$ é a quantidade de palavras na sentença s, w é uma palavra presente em s, e $freq(w,C)$ indica a quantidade de ocorrência da palavra w na coleção textual C.

No OPP-Alinhamento, foram analisados alinhamentos semânticos entre sentenças dos textos-fonte e respectivos sumários nos córpus CSTNews. Esses alinhamentos foram manualmente identificados no córpus, cuja metodologia e detalhes do processo são descritos em (AGOSTINI; CONDORI; PARDO, 2014). Duas sentenças s_t e s_{sum} foram alinhadas se: (i) o conteúdo de s_t está, pelo menos parcialmente, semanticamente presente em s_{sum}; e (ii) a sentença s_{sum} está presente em um sumário que foi produzido a partir da coleção textual em que o texto da sentença s_t ocorre. Por exemplo, no Quadro 21, são dispostos trechos de textos-fonte e sumários com os respectivos alinhamentos identificados. É interessante observar que mesmo com a presença de diferentes itens lexicais, o alinhamento foi identificado por meio dos significados das sentenças. Pode-se afirmar, então, que tal procedimento viabiliza um tipo de conhecimento linguístico muito refinado, permitindo um cálculo mais informado de ranqueamento sentencial.

Dessa forma, para treinamento do modelo OPP-Alinhamento, admitiu-se que os alinhamentos supracitados descrevem que o conteúdo de alguma sentença está presente no sumário e que, portanto, essa sentença é relevante. Assim, computou-se a relevância das

Quadro 21 – Exemplo de alinhamentos sentenciais identificados entre sentenças dos textos-fonte e sumários no córpus CSTNews.

Sentença do sumário	Sentença do documento
Vários moradores e turistas nas regiões, inclusive brasileiros, foram retirados dos locais, enquanto outros estão se preparando para a passagem do furacão.	Na Jamaica, muitos estocaram alimentos, água, lanternas e velas.
O Brasil não fará parte do trajeto de 20 países do revezamento da tocha	A tocha passará por vinte países, mas o Brasil não estará no percurso olímpico.

sentenças por meio da quantidade dos respectivos alinhamentos identificados no córpus.

Após calcular o valor de relevância para cada sentença no córpus, tanto para o OPP-Freq quanto para o OPP-Alinhamento, computou-se a relevância de cada posição sentencial pelo mesmo procedimento adotado pelo OPP.

5.1.3 Fator de Novidade (NF)

A métrica de Fator de Novidade (NF, do inglês *Novelty-Factor*) (VARMA *et al.*, 2009) é uma abordagem de ranqueamento de sentenças que visa identificar conteúdo atualizado baseada na diferença de vocabulário presente nos textos-fonte de um grupo A_C (mais antigos) e B_C (mais recentes) de uma mesma coleção C. Por meio do NF, a pontuação de uma sentença s é definida pela equação a seguir, em que w é uma palavra que ocorre em s, e $|w \in D_X|$ indica o número de textos no grupo textual X (A_C ou B_C) em que a palavra w ocorre.

$$NF(s) = \frac{1}{|s|} \sum_{w \in s} \frac{|w \in D_B|}{|w \in D_A| + |D_B|} \tag{5.1}$$

Como pode ser observado na formulação supracitada, uma sentença com alto valor de NF é aquela que possui proporcionalmente (pois o valor de NF computado para uma sentença é normalizado pelo seu respectivo tamanho) mais palavras frequentes nos textos do grupo B_C em relação ao grupo A_C.

A abordagem definida pelo NF é uma maneira simples e veloz de identificar o nível de atualização de uma sentença, baseando-se na ideia de que conteúdo atualizado é aquele que não ocorreu previamente. Entretanto, o NF considera apenas informações presentes nos textos-fonte e não verifica se palavras que ocorrem em um mesmo documento possuem diferentes níveis de atualização. Por exemplo, mesmo que duas palavras distintas ocorram em um mesmo conjunto de textos-fonte, sejam eles do grupo A_C ou B_C, elas podem indicar diferentes níveis de relevância ou atualização considerando o assunto presente nos textos.

Dadas as características supracitadas, foi proposta a utilização de subtópicos para enriquecer a abordagem de NF, conforme indicado na equação a seguir, em que Sub_X

indica o conjunto de subtópicos presentes em um grupo textual X (A_C ou B_C) da coleção C. Dessa forma, o NF é alterado de forma a considerar a diferença de vocabulário existente entre os subtópicos que compõem o assunto principal dos textos-fonte dos grupos A e B. Dessa forma, mesmo que duas palavras distintas ocorram em um mesmo conjunto de textos-fonte, podem-se identificar diferentes níveis de atualização.

$$NF_{sub}(s) = \frac{1}{|s|} \sum_{w \in s} \frac{|w \in Sub_B|}{|w \in Sub_A| + |Sub_B|} \tag{5.2}$$

5.1.4 Métodos baseados em grafos

Algoritmos para a tarefa de SA que fazem uso de alguma estrutura de grafo são amplamente utilizados na literatura. Pode-se citar os trabalhos de Erkan e Radev (2004), Leskovec, Milic-Frayling e Grobelnik (2005), Lin e Kan (2007), Li, Du e Shen (2011), Ribaldo *et al.* (2012), Li, Du e Shen (2013). Com relação ao conhecimento dos autores deste trabalho, o resultado mais expressivo no contexto da SAA por meio de algoritmos baseados em grafo foi alcançado pela investigação de Wenjie *et al.* (2008), na qual foi proposto o método *Positive and Negative Reinforcement* (PNR^2).

No método PNR^2, inicialmente, um grafo $G(V,E)$ é construído para modelar uma coleção de documentos, sendo que cada vértice $v \in V$ representa uma sentença dessa coleção; e cada aresta $a \in E$ entre dois vértices (duas sentenças) é valorada com a respectiva similaridade sentencial calculada por meio da métrica do Cosseno (SALTON; WONG; YANG, 1975). Além disso, defini-se G como um grafo completo, em que cada vértice $v_i \in V$ possui uma aresta para todos os demais vértices $v_j \in V$ com $i \neq j$. Com a composição de um grafo que modela os textos-fonte, são definidas pontuações aleatórias, com valores entre 0 e 1, para cada sentença. Posteriormente, por meio de um estágio de otimização, essas pontuações são atualizadas por meio de reforços positivos e negativos entre as sentenças, tal que um reforço positivo ocorre somente entre duas sentenças que estejam presentes em um mesmo grupo de textos A_C ou B_C de uma coleção C. Por outro lado, um reforço negativo ocorre entre sentenças que estão presentes em grupos diferentes.

Esses reforços, ou relações sentenciais, indicam o mecanismo que o algoritmo emprega para atualizar a pontuação entre duas sentenças, compartilhando suas pontuações conforme a respectiva similaridade entre elas e um parâmetro positivo β ou negativo α, conforme o tipo de relação sentencial. Dessa forma, uma sentença com alta pontuação indica que essa é mais similar a sentenças que ocorrem em textos do grupo B_C e dissimilar com aquelas do grupo A_C.

Formalmente, PNR^2 define uma matriz de adjacências M, em que $M_{i,j}$ indica a similaridade entre as sentenças s_i e s_j, em que $m_{i,i} = 0$. Posteriormente, cada célula de M é multiplicada por α ou β conforme as restrições positivas e negativas apresentadas

anteriormente. Dessa forma, a matriz M armazena as relações positivas e negativas entre todas as sentenças presentes nos grupos A_C e B_C. Cada peso/valor sentencial é ajustado por meio de um procedimento de otimização iterativo, que é definido pela equação a seguir, em que R_i^k indica a pontuação da sentença s_i previamente calculada na iteração de número k, $R_i^{(k+1)}$ representa a pontuação da sentença s_i sendo calculada, $\overrightarrow{p_i}$ é a relevância da sentença s_i para uma determinada consulta do usuário (quando não disponível, pode-se usar o valor 1), e $a_{i,j}$ é o peso (positivo, ou negativo) entre as sentenças s_i e s_j.

$$R_i^{(k+1)} = \frac{1}{a_{ij}}(\overrightarrow{p_i} - \sum_{j<i} a_{ij} R_j^{(k+1)} - \sum_{j>i} a_{ij} R_j^{(k)}) \tag{5.3}$$

A atualização dos pesos das sentenças ocorre até a convergência do modelo ou até que o número de iterações máximas definido tenha sido alcançado. A convergência pode ser verificada pela diferença δ entre a alteração dos pesos em R^{k+1} e R^k, de forma que o algoritmo termina quando o valor de δ for menor que um determinado parâmetro previamente estipulado. Portanto, o algoritmo pode finalizar sua execução quando não há uma alteração significativa entre os pesos das sentenças. Neste trabalho, foram definidas condições de parada quando $\delta < 0,001$ e número máximo de iterações igual a 1000. Ressalta-se que no artigo original dos autores do PNR^2, esses parâmetros não são apresentados.

Como pode ser observado pela descrição do método PNR^2, o algoritmo considera apenas relações entre pares sentenciais, que podem ser positivas ou negativas conforme os textos em que essas sentenças ocorrem. Dessa forma, visando à utilização de informação discursiva na modelagem textual proposta pelo método PNR^2, foi proposta inicialmente a utilização de subtópicos, que será posteriormente referenciada como **PNR^2+SUB**, por meio da ideia de que se duas sentenças possuem uma relação positiva por ocorrerem em um mesmo tipo de grupo textual (A_C ou B_C), as relações entre sentenças presentes em um mesmo subtópico devem ser maiores, pois essas pertencem a um mesmo assunto que compõe a ideia principal de um texto. Para tanto, o preenchimento da Matriz M é dado pelas seguintes restrições, em que $sub(s_k)$ indica o subtópico em que s_k está presente:

$$M[i][j] = \begin{cases} 0, & \text{se i = j} \\ 1, & \text{se } sub(s_i) = sub(s_j) \\ cosseno(s_i, s_j), & \text{caso contrário} \end{cases} \tag{5.4}$$

Ressalta-se que não foi utilizado um valor maior do que 1 para representar o reforço positivo entre sentenças presentes em um mesmo subtópico. Essa restrição foi necessária para garantir a convergência do modelo de otimização.

Durante a etapa de otimização para utilização dos subtópicos, foram empregados os valores dos parâmetros $\alpha = -0,5$ e $\beta = 1$, que também foram utilizados nos experimentos reportados em (WENJIE *et al.*, 2008). Assim, de forma similar ao algoritmo PNR^2

original, a aplicação proposta de subtópicos também assume que uma boa sentença para o sumário é aquela mais similar com sentenças dos textos do conjunto B_C e dissimilar com sentenças do conjunto A_C. Além disso, admitindo que o peso da aresta entre sentenças de um mesmo subtópico é 1, sentenças com pontuações altas contribuem para incrementar o peso das demais sentenças em seu respectivo subtópico. Dessa forma, espera-se que, após a convergência do modelo, as sentenças dos subtópicos mais relevantes para a coleção de textos-fonte também sejam identificadas.

Tendo em vista que as relações discursivas multidocumento do modelo CST podem ser modelas por meio de um grafo, foi proposta uma adaptação do algoritmo PNR^2, referenciada como **PNR^2+CST**, para utilização de relações CST em vez da similaridade sentencial computada por meio da métrica do Cosseno. Para tanto, inicialmente foram identificadas as relações discursivas CST nos textos-fonte por meio do CSTParser[1] (MAZIERO; PARDO, 2012), que é um parser discursivo do estado da arte para a língua Portuguesa.

Após a identificação das relações CST entre os pares sentenciais dos textos-fonte de cada coleção C, para utilização do algoritmo de otimização para atualização dos pesos das sentenças empregado no PNR^2, cada relação discursiva identificada é alterada por um respectivo valor numérico, que indica o peso da respectiva relação entre duas sentenças. Esses pesos foram definidos empiricamente e são dispostos no Quadro 22. Como pode ser observado, foram definidos valore positivos e negativos para serem utilizados quando as sentenças ocorrem em coleções textuais iguais ou distintas, respectivamente. Por exemplo, a relação *Equivalence*, que indica que duas sentenças possuem conteúdo similar, foi definida com os valores de 0,8 (reforço positivo) e -1 (reforço negativo). Aqui, é importante também notar que esse tipo de informação discursiva é muito mais refinado do que a utilização de apenas a métrica do Cosseno. Por exemplo, uma vez que a métrica do Cosseno identifica o nível de similaridade sentencial por meio dos itens lexicais em comum, duas ou mais sentenças podem apresentar conteúdo muito similar e serem compostas por itens lexicais distintos. Além disso, algumas informações relevantes para a SAA, como fluxo temporal ou nível de contradição entre pares sentenciais, não são identificadas. Por outro lado, por meio da CST, tem-se a relação *Historical-background* e *Follow-Up*, que identificam um tipo de conhecimento muito pertinente no cenário da SAA.

Para utilização das relações CST em conjunto com o método PNR^2, tendo em vista que essa informação discursiva modela um tipo de conhecimento muito refinado e que foram definidos pesos numéricos para cada relação CST, omitiram-se os parâmetros α e β (considerados com valor igual a 1) para multiplicar esses pesos para compor a matriz M, que é usada no algoritmo de otimização definido pelo PNR^2.

[1] Neste trabalho, foi utilizado a versão desktop dessa ferramenta, que está disponível para download na seguinte URL: <http://www.icmc.usp.br/~taspardo/sucinto/files/CSTParser-20standalone.rar>

Quadro 22 – Pesos empiricamente definidos para relações CST visando à utilização dessa informação discursiva no método PNR^2.

Relações CST	**Ocorre em Coleção**	
	Igual	Diferente
Identity	1	-1
Equivalence	0.8	-1
Summary	0.9	-1
Subsumption	0.7	-0.9
Overlap	0.6	-0.5
Historical-background	0.4	-1
Follow-up	0.6	-0.2
Elaboration	0.5	-0.1
Contradiction	0.5	-0.1
Citation	0.3	-0.8
Attribution	0.3	-0.8
Modality	0.4	-0.8
Indirect-speech	0.2	-1
Translation	0.2	-1

Além das propostas de aplicação de subtópicos e relações CST ao PNR^2, admitindo-se que essas informações podem ser complementares, investigou-se também o uso desses dois conhecimentos linguísticos em conjunto, que será referenciada com **PNR^2+SUB+CST**. Para tanto, foram utilizadas as relações CST e os respectivos pesos como arestas no grafo que modela os textos-fonte. Porém, as arestas entre sentenças que pertencem a um mesmo subtópico foram ponderadas com o valor 1.

Além das investigações supracitadas que almejam incorporar diferentes conhecimentos linguísticos ao método PNR^2, foram também realizados experimentos com o algoritmo PageRank (BRIN; PAGE, 1998), de forma similar como apresentado no trabalho de Wenjie *et al.* (2008). Para tanto, foram utilizados os mesmos procedimentos anteriores para criação de um grafo $G(V,A)$ a partir de uma coleção de textos-fonte, bem como as restrições para incorporar os conhecimentos de subtópicos e CST.

De forma semelhante aos experimentos de Wenjie *et al.* (2008), foram investigadas duas configurações do algoritmo PageRank, o **PageRank(B)** e **PageRank(A+B)**. Na primeira, um grafo para representação textual é construído a partir de todas as sentenças presentes nos textos-fonte do conjunto B_C, somente. Por outro lado, na segunda configuração, todas as sentenças dos textos dos grupos A_C e B_C são utilizadas para compor o grafo. Contudo, ressalta-se que no contexto da SAA, o sumário produzido pelos métodos é constituído de sentenças advindas somente de textos do conjunto B_C.

5.1.5 KLSum

KLSum é uma abordagem de SA que faz uso da divergência de Kullback-Leibler (KL), que tem sido amplamente investigada em trabalhos de SA, tais como Haghighi e Vanderwende (2009), Wang *et al.* (2009), Castro Jorge e Pardo (2011), Delort e Alfonseca (2012). Primeiramente, métodos de SA baseados no KLSum aprendem uma distribuição de elementos alvo T a partir de uma coleção de textos-fonte e, posteriormente, definem que um bom sumário é aquele cuja distribuição S seja a mais próxima (menos divergente) à T.

Comumente, os elementos computados por meio dessa abordagem são n-gramas, de forma que T e S representam a distribuição de probabilidades de cada n-grama w presente nos textos-fonte e sumário, respectivamente. Inicialmente, para se calcular a distribuição T a partir de uma coleção de textos-fonte C, define-se $pT(w)$ para cada w presente em um vocabulário fixo V como $\frac{f(w,C)}{|V|}$, em que $f(w,C)$ indica a frequência de w na coleção C; e $|V|$ é o tamanho do vocabulário. Posteriormente, por meio dessa modelagem, pode-se também representar o conteúdo dos sumários sendo produzidos. Para tanto, basta utilizar o mesmo vocabulário V e computar a probabilidade de cada palavra sob o conteúdo do sumário.

Uma vez definida a distribuição T, um sumário pode ser produzido a partir de um subconjunto de sentenças $S*$ dos textos-fonte que satisfaçam as restrições de tamanho do sumário de forma que a respectiva distribuição S seja a mais próxima a T por meio da equação a seguir, em que τ é um fator de suavização frequentemente empregado para impedir divisão indefinida por zero.

$$S* = argmin_S \; KL(T,S) = \sum_{w \in V} pT(w) \; log \frac{pT(w)}{pS(w) + \tau} \tag{5.5}$$

Tendo em vista que a formulação supracitada implica na análise de todos os subconjuntos válidos (dada as restrições de tamanho do sumário) de sentenças dos textos-fonte para compor o sumário, e que tal procedimento exige muito tempo computacional, métodos de SA por meio dessa abordagem frequentemente empregam um algoritmo guloso. Dessa forma, a equação anterior é utilizada iterativamente, de forma que apenas uma sentença é selecionada para o sumário em cada passo da execução. Em outras palavras, a cada momento, o algoritmo identifica a sentença que mais aproxima a distribuição S do sumário à distribuição alvo T.

Neste trabalho, visando incorporar conhecimento de subtópicos ao KLSum, foram empregadas duas propostas diferentes, o **KLSum-Sub** e **KLSum-E**, que computam a distribuição alvo a partir dos subtópicos presentes nos textos-fonte.

Para o KLSum-Sub, assumiu-se que cada subtópico de um texto indica uma porção

da ideia principal descrita em toda a coleção de textos-fonte. Dessa forma, foi proposta a produção de sumários em que as respectivas sentenças melhor representam a distribuição dessas ideias. Para tanto, a formulação anterior do KLSum foi alterada, de forma que a distribuição de probabilidades dos n-gramas de um vocabulário V foi computada por meio dos subtópicos e não mais a partir dos textos. Essa formulação pode ser observada a seguir, em que: Sub corresponde ao conjunto de todos os subtópicos que ocorrem em uma determinada coleção de textos-fonte e c_j é um subtópico desse conjunto.

$$pT(w) = \frac{1}{|Sub|} \sum_{c_j inSub} 1 \; if \; w \in c_j \; else \; 0 \tag{5.6}$$

Na segunda proposta para incorporar subtópicos ao método KLSum, referenciada como KLSum-E, assumiu-se que: (i) a ideia principal M de uma coleção de textos é modelada por um conjunto sub-ideias $m \in M$, que, juntos, a compõe; cada subtópico $c \in Sub$ indica a existência de alguma ideia m; e que $|Sub| \leq |M|$, consequentemente, assume-se que uma mesma ideia m pode ocorrer em um ou mais subtópicos. Além disso, foi definido que cada subtópico, de forma similar a uma coleção de textos, pode ser também modelado como uma distribuição de probabilidades a partir de um vocabulário V definido.

Dadas as definições supracitadas, pode-se computar a probabilidade para cada subtópico $c \in Sub$ como uma estimativa de cada sub-ideia que compõe o tema principal da coleção de textos. Dessa forma, o método KLSum-E, por meio da equação a seguir, computa a distribuição alvo T por meio da estimativa dessas ideias em vez de palavras, usando a probabilidade de cada subtópico como uma aproximação das ideias presentes nos textos $pT(c)$.

$$pT(c) = \frac{1}{|V|} \sum_{w_i inV} f(w_i, c) \tag{5.7}$$

Adotando a formulação anterior, deve-se também modelar a distribuição pS do sumário como uma aproximação da distribuição de ideias. Assim, se $Sub \approx M$, a distribuição do sumário pode ser computada por meio da probabilidade de suas respectivas sentenças que ocorrem em uma determinada ideia, conforme a equação a seguir:

$$pS(c) = \sum_{w \in V} \frac{f(w,S)}{|S|} * \frac{f(w,c)}{|c|} \tag{5.8}$$

Dessa forma, pode-se utilizar a equação a seguir para calcular a divergência KL por meio de um algoritmo guloso, visando aproximar o sumário à distribuição alvo, que, nesse momento, estima a quantidade de sub-ideias que compõem o tema principal dos

textos-fonte:

$$KL(T,S) = \sum_{c \in Sub} pT(c)\ log \frac{pT(c)}{pS(c)+\tau} \tag{5.9}$$

5.1.6 Modelo baseado em distribuição de tópicos

Modelos de representação de tópicos por meio de diferentes abordagens, como LSA (*Latente Semantic Analyis*) e LDA (*Latent Dirichlet Allocation*), foram investigados em diferentes tarefas da SA, em que se podem citar os trabalhos de (REEVE; HAN, 2007; STEINBERGER; JEžEK, 2009; HUANG; HE, 2010; LI *et al.*, 2012; DELORT; ALFONSECA, 2012). No âmbito da SAA, o trabalho com resultados mais pertinentes foram alcançados por meio do sistema DualSum que, foi proposto por Delort e Alfonseca (2012). O DualSum é uma extensão introduzida no modelo de SA multidocumento baseado em tópicos do TopicSum (REEVE; HAN, 2007). Segundo os autores, essa ampliação do modelo permite o tratamento mais eficiente dos desafios existentes na tarefa de SAA.

No DualSum, a partir de uma coleção textual C com dois grupos de textos, A_C (textos antigos) e B_C (textos mais recentes), o algoritmo aprende um modelo de distribuição de tópicos latentes similares à formulação de LDA. Posteriormente, de forma similar ao algoritmo KLSum apresentado na seção anterior, um sumário é produzido de forma que sua distribuição de tópicos seja similar à distribuição aprendida a partir da coleção textual.

Inicialmente, cada texto é representado como uma *bag-of-words* e cada palavra é associada a um tópico latente. Na proposta do DualSum, são definidos os quatro seguintes tipos possíveis de tópicos: ϕ^G, que representa um tópico geral, para identificar palavras de uso comum; ϕ^{Cd}, que representa um tópico relevante para um determinado texto d de uma coleção textual C; por fim, ϕ^{Ac} e ϕ^{Bc}, que são os tópicos considerados importantes para os grupos textuais A_C e B_C, respectivamente. Aqui, é importante ressaltar que a partir de um texto no grupo B_C, pode-se identificar todos esses possíveis tópicos, uma vez que o conteúdo nos textos do grupo B_C são relacionados às informações existentes no grupo A_C. Porém, a partir de um texto da coleção A_C, não é possível aprender um tópico do tipo ϕ^{Bc}. Dessa forma, visando a produção de bons sumários de atualização, o DualSum almeja produzir sumários cuja distribuição de tópicos é mais direcionada a ϕ^{Bc} do que ϕ^{Ac}.

Por meio da modelagem de SAA empregada pelo método DualSum, foram propostos dois métodos de SAA baseados em informações de subtópico. Na primeira abordagem, referenciada como **DualSum+SUB**, admitindo-se que um subtópico encapsula uma informação discursiva mais pontual em relação ao assunto presente nos textos-fonte, foi alterado o tópico ϕ^{cd} para ϕ^{sub-cd}, em que *sub* é um subtópico de um determinado do-

cumento d em uma coleção c. Em outras palavras, alterou-se o DualSum para aprender uma distribuição de tópicos por meio dos subtópicos em vez de aprender a partir dos respectivos textos. Ressalta-se que foram mantidos todos os demais tópicos ϕ^G, ϕ^{Bc} e ϕ^{Ac}.

5.1.7 Método Ensemble

Como apresentado nas seções anteriores, investigaram-se diferentes técnicas de SAA, que requerem diferentes níveis de processamento computacional e baseiam-se na análise de distintas características textuais para identificar as sentenças mais relevantes para o sumário. Nesse cenário, tendo em vista essa variabilidade, pode-se combinar algumas dessas abordagens por meio de um método Ensemble, de forma que a identificação das sentenças para o sumário ocorra após a análise de características distintas dos textos-fonte.

Para composição do Ensemble, utilizou-se os métodos baseados em Caraterísticas Posicionais (5.1.1), Fator de Novidade (NF, 5.1.3) e os métodos baseados em grafos (5.1.4). Esses métodos foram adotados por serem de abordagens variadas e por requererem baixo custo computacional, em relação aos métodos KLSum e DualSum. Além disso, tais métodos admitem pontuações de relevância para cada sentença candidata para o sumário. Por exemplo, o método PNR^{2}computa valores para cada sentença após um procedimento de otimização, Por outro lado, no método NF, cada sentença recebe uma pontuação por meio de um equacionamento baseado na diferença de vocabulário entre os textos mais recentes e antigos. Além disso, para esses dois métodos, assim como os demais selecionados para o Ensemble, definem-se que valores de pontuação mais elevados indicam sentenças mais relevantes.

Dados os métodos supracitados, bem como suas características para seleção de sentenças, optou-se por desenvolver um método Ensemble baseado em soma de valores. Assim, cada sentença s é ranqueada pela soma das respectivas pontuações computadas por cada método no Ensemble. Para tanto, utilizou-se a seguinte equação $rank_{ensemble}(s) = \sum m_i(s)$, em que m_i indica o i-ésimo método que compõe o Ensemble.

A cada iteração do algoritmo, todas as sentenças dos textos-fonte, que ainda não foram selecionadas para o sumário, são ranqueadas por meio da equação anterior e, posteriormente, a sentença com maior pontuação (maior valor de $rank_{ensemble}(s)$) é direcionada para o sumário. Contudo, é importante salientar que os métodos selecionados possuem escalas de valores variadas para ranquear as sentenças. Por exemplo, os métodos baseados em características posicionais assumem um valor de relevância igual a 1 para a primeira ocorrência de uma sentença ou item lexical em um texto-fonte, porém, decrementam esse valor para as demais ocorrências de forma mais suave ou abrupta conforme a proporção utilizada (vide as proporções Direta, Inversa, Geométrica e Binária na Seção 5.1.1). Por outro lado, por meio do método NF, a sentença melhor ranqueada não necessariamente

terá pontuação igual a 1. Assim, tendo em vista que essa variação de intervalos poderia privilegiar alguns procedimentos de ranqueamento, optou-se por normalizar os valores de pontuação computados por cada método m_i individualmente entre valores de 0 a 1. Assim, garante-se que em cada passo de execução, a sentença melhor ranqueada por cada método, terá a maior pontuação respectiva e que tal pontuação será equivalente aos demais métodos.

5.1.8 Uso de entidades nomeadas

Alguns métodos apresentados anteriormente, como o Fator de Novidade (NF), KLSum e DualSum, empregam algum tipo de processamento de vocabulário presente nos textos-fonte, visando identificar as sentenças mais relevantes para o sumário. Por exemplo, no método NF, analisa-se a diferença de vocabulário nos textos do grupo A_C e B_C de uma coleção textual C. Por outro lado, o DualSum associa tópicos latentes a cada n-grama de um vocabulário pré-definido.

Em geral, o procedimento para identificação de um vocabulário a partir de uma coleção textual se resume na tokenização, normalização e filtro dos itens lexicais de cada sentença. Por exemplo, comumente, os sinais de pontuações e *stopwords* são desconsiderados e os itens lexicais remanescentes são utilizados com ou sem normalização, como radicalização ou lematização. Dessa forma, algumas informações semânticas importantes podem ser desconsideradas, como a presença de entidades nomeadas, que são cadeias lexicais que referenciam entidades como pessoas, lugares, organizações, etc.

Dado o cenário supracitado, foi proposta a utilização de Entidades Nomeadas (EN) durante o procedimento de identificação do vocabulário de uma coleção textual. Para tanto, inicialmente, os textos-fonte foram processados por meio um identificador automático de EN e, posteriormente, as cadeias de entidades foram considerados itens lexicais únicos. Por exemplo, no trecho “As Olimpíadas do Rio de Janeiro...’ ’, por meio da análise de n-gramas com $n = 1$, haveriam 6 itens lexicais (considerando-se as *stopwords*), de forma que a cadeia “Rio de Janeiro”, que referencia uma entidade do mundo real (a cidade Rio de Janeiro), seria considerada uma sequência de três itens lexicais isolados. Assim, por meio da abordagem proposta, essa cadeia é considerada uma entrada única no vocabulário sendo identificado.

Ressalta-se que, no decorrer deste texto, será utilizada a marcação **+EN** para identificar os métodos investigados para os quais foi adicionada essa abordagem para incorporar Entidades Nomeadas no procedimento de criação/análise do vocabulário presente em um texto-fonte.

Neste trabalho, foram utilizadas duas ferramentas para extração de entidades nomeadas, sendo uma para a língua Portuguesa e outra para o idioma Inglês. Para o pri-

meiro idioma, foi utilizado o extrator de EN desenvolvido no grupo de pesquisa LIAAD[2], que é vinculado à Universidade do Porto. Para a língua inglesa, foi utilizado o modelo de etiquetagem morfo-sintática do Parser da Universidade de Stanford estendido para a identificação de Entidades Nomeadas, que está disponível no pacote de desenvolvimento NLTK para a linguagem Python.

5.1.9 Identificação e remoção de conteúdo redundante

Um bom método de SA deve produzir sumários sucintos que, preferencialmente, sejam constituídos por informação não redundante. Entretanto, com exceção dos métodos baseados em modelos gerativos, como o KLSum e DualSum, as abordagens de sumarização apresentadas anteriormente não eliminam o conteúdo redundante intrinsecamente. Assim, para esses métodos, foi empregado o mesmo procedimento para remoção de redundância que adotado por Ribaldo *et al.* (2012), que será descrito a seguir.

Pelo método supracitado, somente sentenças consideradas não redundantes em relação ao conteúdo do sumário são utilizadas. Para tanto, descartam-se as sentenças candidatas cuja similaridade para com alguma sentença já inserida no sumário seja maior que um limiar de redundância previamente definido. Como métrica de similaridade sentencial, foi adotada a métrica do Cosseno (SALTON; WONG; YANG, 1975), que é amplamente utilizada na literatura. Por meio dessa métrica, pode-se obter um valor entre 0 e 1 que indica a similaridade entre duas sentenças, de forma que essa similaridade é diretamente proporcional ao valor computado.

Comumente, o limiar de redundância é definido empiricamente. Contudo, segundo a abordagem de Ribaldo *et al.* (2012), defini-se esse valor dinamicamente para cada coleção textual por meio da média entre o valor do Cosseno entre todos os respectivos pares sentenciais. Dessa forma, tem-se que o limiar mais flexível a diferentes contextos. Por exemplo, o limiar calculado para uma coleção com textos muito similares será maior do que em conjuntos textuais muito distintos.

É importante ressaltar que para a produção de sumários de atualização, definiu-se que uma coleção textual C é constituída por dois conjuntos de documentos, referenciados como A_C e B_C. Além disso, uma vez que se almeja produzir um sumário a partir do conjunto B_C admitindo-se que o leitor conhece o conteúdo em A_C, somente as sentenças presentes em B_C foram consideradas para computar o limiar de redundância pelo procedimento descrito anteriormente.

[2] *Laboratory of Artificial Intelligence and Decision Support*, cujo site está disponível em <http://www.liaad.up.pt/>

5.2 Sumarização compressiva

Tendo em vista que foram investigados diversos métodos de SAA, foi proposta uma Arquitetura de Síntese Compressiva, em que a etapa de produção do conteúdo para o sumário é desacoplada das demais fases da SA. Dessa forma, pode-se empregar tal arquitetura em diversos métodos de SA baseados na Síntese Extrativa.

Na Figura 9, é apresentada a arquitetura proposta, que é composta pelos seguintes três módulos principais: Seleção, Compressão e Síntese. No primeiro, pode-se adotar qualquer método de SA cuja execução seja iterativa, ou seja, um método que seleciona apenas uma sentença para o sumário em cada etapa de execução. Após essa seleção, no Módulo de Compressão, geram-se algumas versões comprimidas das sentenças candidatas. Posteriormente, no módulo de Síntese, identifica-se a sentença mais adequada para o sumário, que pode ser a sentença original ou alguma respectiva versão comprimida.

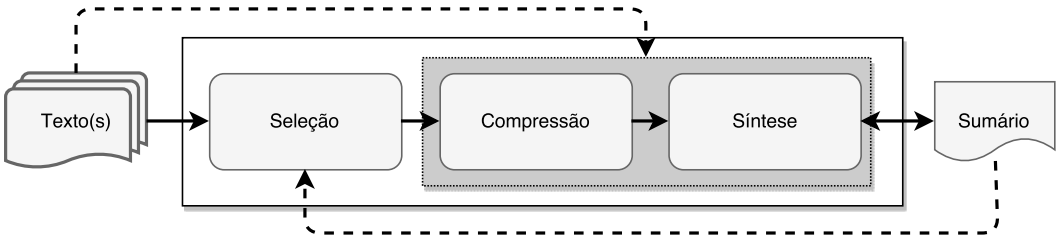

Figura 9 – Arquitetura de Síntese Compressiva proposta.

Pode-se observar que, após a adição de uma sentença ao sumário, seja ela comprimida ou não, o conteúdo do sumário é direcionado ao primeiro modulo. Isso é necessário, pois alguns métodos de SA, como o KLSum e DualSum, analisam as informações do sumário sendo produzido para selecionar as próximas sentenças que o comporão.

Para o módulo de Compressão, adotou-se o melhor método de Compressão Sentencial (CS) apresentado no Capítulo 4, que faz uso do algoritmo CRF. Ressalta-se que os métodos de CS investigados neste trabalho comprimem uma sentença constituída por n itens lexicais após n decisões, de forma que, em cada decisão i, o método determinada se o i-ésimo item lexical deve ser removido ou não. Formalmente, para uma sentença de n itens lexicais, o método produz uma resposta Γ com n valores, em que cada Γ_i corresponde à decisão do algoritmo para o i-ésimo item lexical. Além disso, por simplificação, pode-se considerar que uma resposta Γ_i pode ser valorada como 1 ou 0, para indicar que um item lexical deve ser removido ou não, respectivamente.

Com as características supracitadas, tem-se que após a identificação e remoção de todos os itens lexicais demarcados como "1" (ou seja, os itens lexicais que forma classificados como "pode ser removido" pelo modelo compressivo) na sentença sendo processada, será produzida a menor versão comprimida identificada pelo método de CS. Contudo, em alguns cenários pode ser interessante que duas ou mais versões comprimidas diferentes

possam ser analisadas. Assim, a partir de uma resposta Γ de um método de CS, definiu-se que a sentença pode ser interpretada como um conjunto de Blocos de Permanência ou de Remoção. Cada bloco é constituído por uma sequência de itens lexicais que foram identificados como sendo da mesma classe (0 ou 1). Por exemplo, um Bloco de Remoção é composto por itens lexicais que ocorrem sequencialmente na sentença de entrada e que foram todos demarcados como "1". Dessa forma, definiu-se que se pode comprimir uma sentença por meio da remoção iterativa de cada bloco. Assim, tem-se uma nova versão comprimida após a remoção de cada bloco.

No Quadro 23, é exemplificado o procedimento supracitado para produzir diferentes versões comprimidas para uma mesma sentença por meio dos métodos de CS investigados neste trabalho. Pode-se observar que o método de CS identificou 6 itens lexicais que podem ser removidos, que estão demarcados em negrito. Com essa saída do método de compressão, podem-se agrupar 3 Blocos de Remoção para a sentença de entrada. Assim, a primeira versão comprimida é resultado da remoção do primeiro bloco somente. Posteriormente, remove-se o segundo bloco, e assim sucessivamente, para a geração das demais versões comprimidas.

Quadro 23 – Exemplo de composição de diversas versões comprimidas adotada neste trabalho.

Original:	O iPad chega a Portugal **em seis versões** com **o** preço **a variar** entre 499 e 799 euros.
Versão 1:	O iPad chega a Portugal com **o** preço **a variar** entre 499 e 799 euros.
Versão 2:	O iPad chega a Portugal com preço **a variar** entre 499 e 799 euros.
Versão 3:	O iPad chega a Portugal com preço entre 499 e 799 euros.

Após a execução do Módulo de Compressão, tem-se um conjunto de sentenças candidatas para o sumário, que é constituído pela sentença original extraída do texto-fonte e todas as versões comprimidas dessa sentença que foram geradas pelo procedimento supracitado. Contudo, para gerar o sumário, apenas uma dessas sentenças deve ser selecionada. Essa tarefa é executada pelo Módulo de Síntese, em que todas as sentenças candidatas são analisadas em conjunto com o sumário visando identificar aquela mais adequada.

Como algoritmo para o módulo de Seleção, foi investigado o método de seleção de sentenças adotado pelo KLSum (descrito na Seção 5.1.5). Esse algoritmo identifica uma distribuição de conteúdo $\boldsymbol{T}$ a partir dos textos-fonte (computada por meio da probabilidade de n-gramas nos documentos) e almeja produzir sumários com uma distribuição $\boldsymbol{S}$ que seja a mais próxima ao conteúdo dos textos-fonte. Dessa forma, acredita-se que a aplicação desse algoritmo possa verificar qual a sentença mais adequada para o sumário, em relação ao seu conteúdo, e desconsiderar sentenças agramaticais, pois o modelo gerativo gerado a partir dessas sentenças pode se afastar da distribuição alvo.

Para aplicação do KLSum em nossa Arquitetura Compressiva, inicialmente, todas as sentenças candidatas, que são geradas no módulo de Compressão, são analisadas pelo

algoritmo adotado. O método verifica qual sentença, que ao ser inserida no sumário, irá aproximar sua respectiva distribuição S à distribuição de conteúdo T, que foi previamente calculada. Uma vez identificada a melhor candidata, o método de SA adotado pode inseri-la no sumário sendo produzido.

Visando a investigação de abordagens mais ou menos restritivas com relação ao uso das sentenças automaticamente comprimidas, foram propostas também algumas configurações diferentes para a arquitetura proposta, como descrito a seguir:

- **1Comp**: na etapa de Síntese, consideram-se apenas a sentença original (extraída dos textos-fonte) e a menor versão comprimida;
- **+Comp**: na etapa de Síntese, consideram-se a sentença original e todas as versões comprimidas geradas por meio do procedimento descrito anteriormente;
- **OnlyComp**: semelhante à configuração anterior, contudo a sentença original do texto-fonte não é considerada uma candidata para o sumário.

Pode-se observar que as duas primeiras configurações apresentadas podem, eventualmente, conforme o resultado do módulo de Síntese, adicionar alguma sentença extraída do texto-fonte no sumário. Contudo, isso não ocorre na configuração OnlyComp.

5.3 O córpus CSTNews-Upate

O córpus CSTNews-Update é uma configuração diferente do córpus CSTNews (ALEIXO; PARDO, 2008; CARDOSO *et al.*, 2011), que foi proposta neste trabalho e viabiliza a investigação de métodos de SAA para a língua Portuguesa. Ressalta-se que o córpus CSTNews foi amplamente empregado em investigações para SA mono e multidocumento, sobretudo, por meio de abordagens linguisticamente guiadas, como os trabalhos de (RINO *et al.*, 2004; MARGARIDO *et al.*, 2008; Castro Jorge, 2010; RIBALDO *et al.*, 2012; CARDOSO, 2014; RIBALDO; CARDOSO; PARDO, 2016).

Como apresentado no Capítulo 2, o córpus CSTNews possui 50 coleções textuais com 2 ou 3 textos jornalísticos relacionados por assunto. Partindo dessa estrutura, foram definidos cenários adequados às restrições presentes nos córpus das conferências DUC e TAC, em que cada coleção textual C possui dois grupos, referenciados como A_C e B_C, de forma que um sumário de atualização deve ser produzido a partir dos textos em B_C, admitindo-se que o usuário/leitor conhece os textos do outro grupo. Para tanto, foram definidas 59 coleções textuais no córpus CSTNews-Update por meio de duas propostas, uma intracoleção e outra intercoleção, de forma que sempre existam dois ou mais textos nos grupos B_C de cada coleção, com objetivo de manter o contexto multidocumento du-

rante o processo de sumarização, que se faz presente nas definições das tarefas da DUC 2007, TAC 2008 e posteriores.

As abordagens supracitadas, intracoleção e intercoleção, que serão detalhadas a seguir, foram as maneiras identificadas pelas quais foi possível estender a estrutura do córpus CSTNews para o cenário de aplicação da SAA. Dessa forma, além de viabilizar a investigação da SAA por meio desse recurso, algumas anotações e conhecimentos linguísticos explicitados no córpus podem, eventualmente, serem empregados em experimentações nesse tarefa tarefa. Ressalta-se também que, o CSTNews é um córpus de referência para a área de Sumarização Automática para a língua Portuguesa.

Na primeira proposta, referenciada como intracoleção, foram selecionadas todas as coleções textuais do CSTNews com três textos. Posteriormente, para cada coleção, ordenaram-se os textos-fonte pela respectiva data de publicação ou, quando essa informação não estava disponível, por evidências temporais presentes nos textos. Assim, definiram-se os dois textos mais recentes como pertencentes ao grupo B_C e o restante ao grupo A_C (conhecido pelo leitor). É importante ressaltar que definir apenas um texto como conhecido pelo leitor não desconfigura a definição da tarefa de SAA e que tal procedimento é necessário para manter o cenário multidocumento de sumarização, tendo em vista as limitações de número de textos do córpus.

No Quadro 24, é disposta a organização das coleções com três textos do CSTNews. Os textos que foram classificados como do grupo A_C e B_C são listados, respectivamente, na segunda e terceira colunas do quadro, e o respectivo número da coleção na primeira coluna. A diferença temporal, expressa em horas e minutos, entre os textos é disposta na quarta coluna. Por exemplo, na coleção C1, os dois textos mais recentes foram publicados 10 minutos e 1 hora e 58 minutos, respectivamente, após a publicação do texto menos recente. É importante ressaltar que a data e hora de publicação de alguns textos do córpus não estão disponíveis, o que inviabilizou a análise da diferença temporal entre alguns textos. Para esses casos, foi utilizado o símbolo –. Por exemplo, na coleção C10, a data e hora não estavam disponíveis para um texto. Para esses casos, os textos foram manualmente ordenados por meio da análise de referências textuais presentes nos textos.

Na segunda proposta para criação de sumários de atualização por meio do CSTNews, referenciadas como intercoleção, coleções textuais diferentes que abordam assuntos similares foram agrupadas em pares, de forma que um sumário de atualização deverá ser produzido a partir da coleção de textos mais recentes, admitindo que o leitor conheça a coleção anterior. Para definir essa organização, somente os pares de coleções com pelo menos um evento ou fato em comum e de uma mesma categoria do córpus (Ciência, Cotidiano, Dinheiro, Esportes, Mundo ou Política) foram agrupados.

No Quadro 25, listam-se as coleções do CSTNews que foram agrupados manualmente por meio da análise do conteúdo dos textos-fonte. Nas primeira e segunda colunas,

Quadro 24 – Ordenação dos textos por coleções do CSTNews para produção de sumários de atualização.

Coleção	**Textos**		**Diferença temporal**
	Conhecido	**Não conhecidos**	
C1	D3	D1,D2	1:58,0:10
C2	D1	D2,D3	4:31,1:6
C3	D1	D2,D3	9:37, –
C4	D2	D1,D3	0:24, –
C6	D3	D1,D2	–,–
C8	D3	D1,D2	0:0,0:17
C9	D2	D1,D3	3:27,4:16
C10	D4	D3,D5	–,3:29
C11	D4	D3,D5	–,2:3
C12	D2	D1,D3	6:1,5:59
C13	D2	D1,D3	6:36,4:29
C14	D3	D2,D4	0:0,2:35
C15	D3	D2,D4	0:34, –
C16	D2	D1,D3	–,–
C18	D2	D1,D3	2:2, –
C20	D5	D1,D4	0:55, –
C21	D3	D1,D2	–,–
C22	D5	D2,D4	0:7, –
C24	D2	D3,D4	–,0:22
C25	D4	D1,D5	–,–
C26	D4	D1,D5	2:19,2:48
C27	D1	D2,D4	23:49, –
C28	D1	D2,D3	0:12,0:6
C29	D2	D1,D3	2:54,7:49
C30	D2	D1,D3	1:11,1:20
C32	D4	D2,D3	–,–
C33	D2	D3,D4	9:8, –
C34	D1	D2,D3	4:0,1:0
C35	D5	D1,D4	9:4,3:45
C36	D1	D3,D4	–,–
C38	D2	D1,D4	0:12, –
C39	D3	D2,D4	0:47, –
C40	D2	D3,D4	4:36, –
C41	D5	D2,D4	1:27, –
C43	D4	D1,D2	4:47,3:23
C45	D2	D1,D3	0:35,0:5
C46	D2	D3,D4	–,–
C47	D4	D3,D5	–, 0:0
C49	D1	D3,D4	–,5:32
C50	D1	D3,D4	–,1:15

são listadas as coleções que serão definidas como, respectivamente, conhecidas e não conhecidas pelo leitor. Na terceira coluna, é disposta a diferença temporal entre o texto mais recente da primeira coluna e mais antigo da segunda coluna. Por fim, na última coluna é sugerido o assunto (título) que relaciona as coleções. Por exemplo, entre as coleções C2 e C17, a diferença temporal é de 16 dias e um pouco mais de 5 horas, e, além disso, ambas as coleções são constituídas por informações relacionadas às eleições presidenciais de 2007 no Brasil. É importante ressaltar que, de forma análoga ao Quadro 24, houve alguns casos, demarcados com o símbolo –, para os quais a data de publicação dos textos não estava disponível.

Quadro 25 – Agrupamento de coleções do CSTNews para produção de sumários de atualização.

Coleção		**Diferença temporal**	**Título**
Conhecida	**Não conhecida**		
C2	C17	6d 05h32	Eleições presidenciais
C3	C5	61d 22h33	Acidente da TAM e decisões da ANAC
C3	C21	17d 04h12	Acidente da TAM e reforma em Cumbica
C3	C22	4d 21h23	Acidente da TAM e cancelamentos de voo em congonhas
C17	C16	1d 12h33	Escândalo do Sanguessugas
C20	C50	48d 23h18	PEC
C21	C5	44d 17h38	Pronunciamento do Ministro da Defesa
C25	C27	93d 16h03	Trajetória da seleção brasileira de futebol
C28	C8	20d 20h49	Liga Mundial de Vôlei
C38	C41	3d 22h41	Natação brasileira no PAN
C39	C5	43d 19h43	Medidas do Ministério da Defesa sobre a aviação
C40	C44	35d 16h43	Escândalo do Renan Canelheiro
C43	C40	20d 02h40	Escândalo do Renan Canelheiro
C44	C42	5d 23h41	Escândalo do Renan Canelheiro
C46	C32	–	Terremoto no Japão
C49	C24	6d 10h51	Abertura do PAN e resultados do atletismo
C49	C38	03h38	Abertura do PAN e resultados da natação
C49	C41	3d 02h31	Abertura do PAN e resultados da natação
C49	C48	7d 01h31	Abertura do PAN e resultados da vôlei

Com as configurações supracitadas, têm-se a possibilidade de 59 cenários para produção de sumários de atualização, sendo 40 intracoleções e 19 intercoleções. Em todos os cenários apresentados, notam-se as seguintes características: (i) os textos são ordenados cronologicamente por data de publicação e, quando não disponível, por evidências temporais presentes nos textos; (ii) o conhecimento do leitor é representado por um ou

mais textos menos recentes; (iii) o tema principal é mantido; e (iv) o sumário deverá ser produzido a partir de dois ou mais textos.

Como apresentado nos Quadros 24 e 25, a diferença temporal entre os textos conhecidos e não conhecidos nas duas abordagens, intra e intercoleção, é bem distinta. No primeiro caso, a diferença temporal é bem menor, variando entre alguns minutos (zero, em alguns casos) e aproximadamente 24h. Já por meio da segunda proposta, essa diferença alcança até vários dias. Assim, acredita-se que a produção de sumários de atualização, de forma manual ou automática, terá comportamentos distintos nesses cenários. Por exemplo, no cenário intracoleção, acredita-se que identificar conteúdo com atualização será uma tarefa mais complexa, pois esse, provavelmente, estará inserido em meio a conteúdo também conhecido pelo leitor (presente nos textos definidos como lidos). Por outro lado, estima-se que para a maioria dos casos no cenário intercoleção, a identificação de conteúdo novo será mais fácil, dada a distância temporal do texto.

5.4 Avaliação

Nesta Seção, será apresentada a avaliação dos métodos investigados para a língua Portuguesa e Inglesa. Para tanto, inicialmente na Seção 5.4.1, disserta-se sobre a metodologia e métricas de avaliação adotadas neste trabalho. Posteriormente, serão apresentados os resultados dos métodos de SAA para a língua Portuguesa por meio da síntese Extrativa e Abstrativa nas Seções 5.4.2 e 5.4.3, respectivamente. Por fim, na seção 5.4.4, será disposta a avaliação dos métodos desenvolvidos para a língua Inglesa.

5.4.1 Metodologia de avaliação para a tarefa de SAA

Neste trabalho, foram utilizados os *frameworks* de avaliação automática de informatividade ROUGE (LIN, 2004) e Noveau-ROUGE (CONROY; SCHLESINGER; O'LEARY, 2011), pois esses são as formas de avaliação mais presentes em investigações de SAA na literatura. De forma similar, esses *frameworks* contrastam um sumário produzido ***sum*** com um conjunto de textos de referência $\boldsymbol{R}$, comumente constituído com sumários gerados por humanos, com objetivo de computar um valor de informatividade, de forma que um bom sumário automático possui um maior valor de informatividade.

Como apresentado no Capítulo 2, A ROUGE possui diversas configurações, que representam a quantidade ou forma com que um sumário e os textos de referência serão modelados e comparados por meio de n-gramas. Neste trabalho, foram adotadas as configurações de ROUGE-1 (1-grama) e ROUGE-2 (bi-gramas) com os respectivos valores de Precisão, Cobertura e F-1, que são frequentemente reportados na literatura.

Além das configurações da ROUGE, há diversos parâmetros alteráveis que influenciam no valor de informatividade calculado para um sumário. Por exemplo, pode-se

definir o procedimento de tokenização, se serão considerados sinônimos, etc. Neste trabalho, foram utilizados os mesmos parâmetros empregados na competição da *Text Analysis Conference* (TAC) de 2008[3].

A Noveau-ROUGE é uma disposição diferente da ROUGE com foco na tarefa de SAA, em que se enfatiza a necessidade de informação atualizada e/ou nova no sumário produzido. Para tanto, a Noveau-ROUGE computa a informatividade de um sumário por meio de dois conjuntos de referência, R_A e R_B, que correspondem aos textos de referência do grupo A_C (definidos como conhecidos pelo leitor) e B_C (textos não conhecidos pelo leitor) de uma coleção C, respectivamente. Dessa forma, um bom sumário de atualização é aquele que possui avaliação ROUGE maior em R_B em relação à R_A.

A Noveau-ROUGE, como proposta por Conroy, Schlesinger e O'Leary (2011), disponibiliza duas configurações principais que correlacionam dois valores ROUGE, um para cada conjunto de referência usado, por meio de multiplicações que aproximam o valor de informatividade calculado as duas métricas manuais, a métrica da Pirâmide (NENKOVA; PASSONNEAU, 2004) e a Responsividade, que foram descritas no Capítulo 2. Além disso, de forma similar à ROUGE, pode-se também obter valores de Precisão, Cobertura e F-1.

Para a língua Portuguesa, os métodos de SAA investigados foram avaliados por meio do córpus CSTNews-Update. Como apresentado na Seção 5.3, não há sumários de atualização que possam ser adotados como textos de referência nesse córpus. Dessa forma, utilizou-se da abordagem de avaliação proposta em (LOUIS; NENKOVA, 2009), em que os próprios textos-fonte são definidos como referência.

Dada a abordagem de avaliação supracitada, para calcular os valores ROUGE para um determinado sumário produzido a partir de uma coleção de textos C, em que há um grupo A_C de textos antigos e outro B_C de textos mais recentes, definiu-se todos os textos do grupo B_C como textos de referência. Por outro lado, para a Noveau-ROUGE, em que é necessário dois grupos de referência, foram utilizados A_C e B_C.

Para a língua Inglesa, foram também utilizadas as métricas ROUGE e Noveau-ROUGE. Como córpus para investigação, utilizaram-se os conjuntos de dados disponibilizados pelas competições de SAA da DUC 2007 e TAC de 2008 e 2009. Dessa forma, tendo em vista que há sumários produzidos por humanos nesses córpus, adotaram-se esses respectivos sumários para cada córpus como textos de referência em vez dos textos-fonte, como foi adotado para a avaliação em língua Portuguesa.

Em todos os experimentos, de forma similar ao definido nas competições da DUC e TAC, foram produzidos sumários com não mais de 100 palavras. Aqui, ressalta-se que caso alguma sentença faça com que o tamanho do sumário seja maior do que o estipulado, o sumário é truncado.

[3] Parâmetros ROUGE usados: -n 4 -w 1.2 -2 4 -u -c 95 -r 1000 -f A -p 0.5 -t 0 -a.

5.4.2 Resultados dos métodos extrativos

Por meio da metodologia de avaliação descrita na seção anterior, todos os métodos de SAA investigados neste trabalho foram avaliados com o córpus CSTNews-Update, que foi descrito na Seção 5.3.

Na Tabela 8, são dispostos todos os valores aferidos. As métricas de avaliação são alocadas nas colunas e cada método em uma linha. Com relação à ROUGE, são apresentados os valores de Precisão, Cobertura e F-1 respectivamente para as configurações ROUGE-1 e ROUGE-2, que são as mais empregadas na literatura. Para a Noveau-ROUGE, adotaram-se os valores de F-1 para a correlação para a métrica da Pirâmide e de Responsividade. Por exemplo, o método NF (Fator de Novidade) apresenta F-1 para ROUGE-1 e ROUGE-2 iguais a 0,414 e 0,319 respectivamente. Para uma melhor organização dos métodos, eles foram ordenados de forma decrescente por meio do valor F-1 da ROUGE-2. Além disso, alguns itens relevantes para as demais métricas são destacados em negrito. Os métodos demarcados com "*" indicam a utilização de bi-gramas para análise do vocabulário. Ressalta-se que todos os valores foram truncados em três casas decimais.

A diferença média de todos os valores de avaliação computados entre os métodos baseados em algum conhecimento linguístico e as respectivas versões tradicionais é de 0,02. Aqui, ressalta-se que foram consideradas diferenças positivas (quando o método mais linguisticamente informado apresentou resultados superiores) e negativas (que corresponde à situação em que o método tradicional obteve melhores resultados de avaliação).

De modo geral, pode-se observar que as abordagens mais simples para produção de sumários de atualização, como o Fator de Novidade e Métricas Posicionais, possuem resultados ligeiramente melhores em termos de ROUGE quando algum conhecimento linguístico é incorporado. Por exemplo, o valor F-1 para ROUGE-2 do método NF com Entidades Nomeadas (**Fator de Novidade+EN**) é ligeiramente superior à versão tradicional (Fator de Novidade). Além disso, todos os métodos baseados em características posicionais para os quais foi adicionado conhecimento de subtópicos (+SUB) são superiores aos respectivos métodos sem essa informação linguística. Entretanto, há duas exceções, o método **Posição Direta+SUB** e **Fator de Novidade+SUB**, para os quais os valores ROUGE aferidos não são melhores do que os respectivos métodos.

Para os métodos baseados em algoritmos de grafo, PNR^{2}e PageRank (com variações), pode-se observar que a adição de conhecimento de subtópicos foi mais positivamente efetiva para os resultados. Por exemplo, o método **PageRank(B)+SUB** apresenta os melhores valores ROUGE em relação aos demais experimentos baseados em grafos. Isso demonstra que tal conhecimento linguístico possui uma carga de informação discursiva alta que, efetivamente, contribui para a produção de melhores sumários de atualização. Por outro lado, tal resultado não foi alcançado quando se utilizou as relações discursivas

Tabela 8 – Avaliação dos métodos de SAA extrativos investigados neste trabalho.

	ROUGE-1			ROUGE-2				
Método	**Pre**	**Cob**	**F-1**	**Pre**	**Cob**	**F-1**	**NR-Pir**	**NR-Res**
klsum	0,813	0,302	**0,438**	0,636	0,233	0,339	7,132	2,241
Ensemble	0,843	0,296	0,426	**0,648**	0,230	0,329	7,224	1,264
KLSumSub*	0,825	0,297	0,421	0,640	**0,233**	0,329	6,744	2,271
KLSumSub+EN	0,835	0,299	0,425	0,641	0,232	0,328	6,680	2,261
klsum+EN	0,814	0,293	0,416	0,633	0,231	0,326	6,642	2,229
DualSum*	0,823	0,294	0,418	0,636	0,230	0,325	**7,694**	**2,438**
DualSum+Sub*	0,822	0,293	0,417	0,633	0,228	0,323	6,418	2,141
Fator de Novidade+EN	0,833	0,289	0,418	0,634	0,223	0,320	6,658	2,261
Posição Direta	0,831	0,289	0,418	0,632	0,222	0,319	6,639	2,240
Fator de Novidade*	0,837	0,285	0,414	0,645	0,220	0,319	6,942	2,424
Posição Binária+Sub	0,814	0,284	0,409	0,630	0,221	0,318	6,731	2,309
PageRank(B)+Sub	0,815	0,284	0,410	0,629	0,221	0,318	6,651	2,244
Posição Geométrica+Sub	0,811	0,282	0,407	0,629	0,221	0,317	6,692	2,288
PageRank	0,817	0,288	0,413	0,620	0,221	0,317	6,410	2,118
Posição Invertida+Sub	0,807	0,281	0,406	0,627	0,220	0,316	6,672	2,279
PageRank+Sub	0,810	0,286	0,410	0,619	0,221	0,316	6,484	2,144
PageRank(B)	0,822	0,284	0,411	0,627	0,218	0,314	6,642	2,260
PageRank+CST	0,812	0,280	0,406	0,621	0,216	0,312	6,461	2,195
Fator de Novidade+Sub	0,809	0,280	0,404	0,612	0,213	0,307	6,524	2,232
PageRank+CST+Sub	0,791	0,276	0,397	0,606	0,213	0,306	6,517	2,202
PNR^2+Sub	0,798	0,278	0,402	0,606	0,213	0,306	6,298	2,113
Posição Direta+Sub	0,790	0,274	0,396	0,605	0,212	0,305	6,413	2,153
Posição Binária	0,797	0,277	0,400	0,603	0,211	0,304	6,370	2,136
Posição Geométrica	0,796	0,277	0,400	0,602	0,211	0,304	6,389	2,145
Posição Inversa	0,795	0,277	0,400	0,601	0,211	0,304	6,383	2,141
PageRank(B)+CST	0,800	0,276	0,399	0,606	0,211	0,304	6,437	2,199
PNR^2+CST	0,799	0,276	0,399	0,601	0,210	0,302	6,492	2,225
PageRank(B)+CST+Sub	0,788	0,274	0,395	0,598	0,209	0,301	6,368	2,148
PNR^2	0,785	0,271	0,392	0,595	0,207	0,299	6,239	2,087
klsumE+EN	0,779	0,279	0,397	0,575	0,209	0,295	6,133	2,051
klsumE	0,783	0,279	0,397	0,578	0,208	0,295	6,155	2,073
OPP-Freq	0,788	0,272	0,394	0,579	0,202	0,291	6,284	2,092
PNR^2+Sub+CST	0,759	0,263	0,379	0,558	0,195	0,281	6,063	2,025

CST para modelar as arestas entre as sentenças das coleções textuais. Isso pode ser observado pelos resultados dos métodos PageRank(B)+CST, PNR^2+CST e PNR^2+SUB+CST, que não apresentaram valores ROUGE mais elevados em relação aos métodos tradicionais. É importante informar que, em geral, um grafo modelado por meio da CST possui menos arestas entre as sentenças, pois somente sentenças com alguma relação discursiva são conectadas. Além disso, o algoritmo PNR^2, bem como as derivações do algoritmo PageRank investigadas neste trabalho, assumem um grafo completamente conectado, em que uma sentença (vértice) possui uma aresta para todas as demais sentenças na coleção textual. Assim, possivelmente, a diminuição do número de arestas no grafo de representação textual afetou a qualidade dos sumários de atualização produzidos.

O método DualSum apresenta um dos melhores valores para Noveau-ROUGE. Isso já era esperado, pois tal abordagem modela de forma mais robusta o cenário da SAA por meio da análise dos tópicos textuais presentes nos textos já conhecidos pelo leitor e os textos mais recentes. Entretanto, o DualSum foi apenas o sexto melhor método com relação aos valores ROUGE. Provavelmente, isso ocorreu devido ao volume de dados disponíveis no córpus CSTNews-UPDATE para treinamento do modelo de representação dos tópicos textuais, pois tal resultado, como será demonstrado na Seção 5.4.4, não ocorreu nos experimentos para a língua inglesa, que possui córpus maiores. Nesse contexto, nota-se também que o método KLSum, cujo algoritmo principal para seleção das sentenças também está presente no DualSum, apresenta resultados melhores. Isso pode ser explicado pelo fato de que o KLSum possui uma modelagem mais simples, relativamente ao DualSum, que não é tão interferida pela ausência de textos-fontes.

Em termos de Noveau-ROUGE, pode-se observar que os métodos que fazem uso de subtópicos em conjunto com o algoritmo KLSum também apresentam resultados ligeiramente melhores em relação ao método original. Contudo, tal situação não se reflete nos valores ROUGE, para os quais o KLSum tradicional obteve as melhores pontuações.

O método *Ensemble* apresentou o segundo melhor resultado de ROUGE e também o mais elevado valor de Precisão para ROUGE-2. Dessa forma, tal abordagem foi superior à todos os métodos que foram utilizados para compor o Ensemble. Esses resultados corroboram a intuição de que, no cenário da SAA, a variabilidade de métodos pode contribuir para a geração de sumários de atualização mais informativos.

É importante ressaltar que a diferença entre os valores ROUGE e Noveau-ROUGE não foi muito significativa. Entretanto, tal diferença pode ser oriunda da metodologia de avaliação utilizado, que não faz uso de sumários humanos de referência, que foi proposta em (LOUIS; NENKOVA, 2009).

5.4.3 Resultados dos métodos compressivos

Para avaliação da Arquitetura de Síntese Compressiva neste trabalho, foram utilizados os dois melhores métodos de SAA de diferentes abordagens conforme a avaliação descrita na Seção anterior, o métodos DualSum e KLSum. Esses dois métodos foram adotados individualmente como o módulo de Seleção da arquitetura proposta, que foi descrita visualmente na Figura 9.

Na Tabela 9, são dispostos os resultados da avaliação. Para organização dos resultados, utilizou-se a mesma formatação adotada na Tabela 8, na qual as métricas de avaliação e os métodos investigados estão organizados nas colunas e linhas, respectivamente. Por exemplo, o método KLSum 1Comp apresenta valor F-1 para ROUGE-1 igual a 0,430. Para fins de comparação, também é disposto o método KLSum extrativo.

Tabela 9 – Avaliação dos métodos de SAA compressivos investigados neste trabalho.

	ROUGE-1			**ROUGE-2**				
Método	**Pre**	**Cob**	**F-1**	**Pre**	**Cob**	**F-1**	**NR-Pir**	**NR-Res**
Klsum	0,813	0,302	0,438	0,636	0,233	0,339	7,132	2,241
KLSum 1Comp	**0,814**	**0,309**	**0,445**	0,616	0,232	0,335	7,095	1,228
KLSum +Comp	0,782	0,297	0,428	0,585	0,220	0,318	6,796	1,168
KLSum OnlyComp	0,769	0,290	0,419	0,542	0,203	0,294	6,340	1,076

Como pode ser observado, a produção de sumários por meio de somente versões comprimidas das sentenças dos textos-fonte, referenciada como OnlyComp, apresentaram resultados inferiores em relação às demais. Além disso, o melhor modelo de SA Compressiva foi o 1Comp, no qual se utiliza a sentença original ou a respectiva menor versão comprimida produzida pelo método de compressão. Com esses resultados, pode-se concluir que a utilização de inúmeras versões comprimidas insere mais ruído no modelo de seleção de sentenças, tornando mais complexa a tarefa de gerar um bom sumário.

É importante ressaltar que todos os sumários, sejam os extrativos ou compressivos, foram avaliados sobre a mesma perspectiva da Seção 5.4.1, na qual se utilizaram os textos-fonte como modelos para ROUGE. Assim, espera-se que os valores ROUGE computados para os sumários compressivos sejam ligeiramente afetados. Contudo, pode-se perceber que não há uma grande diferença entre as performances do método KLSum 1Comp e Klsum, que é puramente extrativo.

No Quadro 26, 27 e 28 são apresentados alguns exemplos de sumários extrativos e os respectivos sumários compressivos produzidos por algum método supracitado.

5.4.4 Resultados para a língua Inglesa

Nesta seção, serão apresentados os resultados dos experimentos realizados para a língua Inglesa por meio de três córpus distintos, que foram respectivamente empregados

Quadro 26 – Exemplo 1 de sumários extractivos e respectivos compressivos por meio da coleção de ID C4632 do córpus CSTNews-Update.

KLSum
KASHIWAZAKI - Autoridades da maior usina de energia nuclear do mundo, no Japão, admitiram nesta terça-feira que houve mais vazamentos de radiação após um terremoto ter matado nove pessoas no país e deixado milhares de desabrigados. Niigata foi atingida em outubro de 2004 por um tremor também de magnitude 6,8, que deixou mais de 3 mil feridos. TÓQUIO - Os chefes da usina nuclear do Japão atingida por terremotos na última segunda-feira admitiram que ocorreram mais vazamentos radioativos no local. - Não sei quando poderei voltar para casa. Nesta segunda, porém, a magnitude foi a mesma que a do primeiro. A Companhia de Energia Elétrica de Tóquio afirmou que o terremoto foi mais forte do que a usina de Kashiwazaki, uma das maiores do mundo, foi planejada para aguentar. **Ao menos 9 pessoas morreram e cerca de 700 se feriram.** Um incêndio afetou a central nuclear de Kashiwazaki-Kariwa, situada próxima do epicentro do tremor, causando o vazamento de água com restos de material radioativo, segundo a companhia elétrica.
KLSum 1Comp
KASHIWAZAKI - Autoridades da maior usina de energia nuclear do mundo, no Japão, admitiram nesta terça-feira que houve mais vazamentos de radiação após um terremoto ter matado nove pessoas no país e deixado milhares de desabrigados. Niigata foi atingida em outubro de 2004 por um tremor também de magnitude 6,8, que deixou mais de 3 mil feridos. TÓQUIO - Os chefes da usina nuclear do Japão atingida por terremotos na última segunda-feira admitiram que ocorreram mais vazamentos radioativos no local. -Não sei quando poderei voltar para casa. Nesta segunda, porém, a magnitude foi a mesma que a do primeiro. A Companhia de Energia Elétrica de Tóquio afirmou que o terremoto foi mais forte do que a usina de Kashiwazaki, uma das maiores do mundo, foi planejada para agüentar. **9 pessoas morreram 700 se feriram.** Um incêndio afetou a central nuclear de Kashiwazaki-Kariwa, situada próxima do epicentro do tremor, causando o vazamento de água com restos de material radioativo, segundo a companhia elétrica.

nas competições de SAA organizadas na DUC 2007 e TAC de 2008 e 2009. Aqui, é importante ressaltar que somente os textos-fonte foram utilizados para os experimentos, assim como foi ilustrado na Figura 8. Portanto, restrições externas, como perguntadas definidas pelo leitor e/ou aspectos textuais, não foram empregados.

Os córpus supracitados são mais difundidos e amplamente empregados em investigações no âmbito da SAA. Além disso, para esses conjuntos de dados, foram disponibilizados sumários de atualização produzidos por humano. Portanto, a metodologia de avaliação proposta em (LOUIS; NENKOVA, 2009), que faz uso dos textos-fonte como textos modelos, não foi adotada. É importante ressaltar que, tendo em vista que o objetivo deste trabalho foi direcionado à tarefa de SAA, apenas sumários de atualização foram produzidos e avaliados por meio destes córpus. Assim, não foram produzidos/avaliados sumários tradicionais multidocumentos para os textos-fonte identificados com o rótulo A, que estão presentes em todos esses conjuntos de dados.

Quadro 27 – Exemplo 2 de sumários extractivos e respectivos compressivos por meio da coleção de ID C2 do córpus CSTNews-Update.

KLSum
No segundo turno, as intenções de voto do presidente Lula caíram de 53% em junho para 50% em julho, enquanto o candidato Alckmin subiu de 29% para 36%. A pesquisa CNI/Ibope realizada em julho e divulgada nesta sexta-feira mostra que o presidente Luiz Inácio da Silva teria 44% dos votos no primeiro turno, enquanto o candidato tucano Geraldo Alckmin teria 25% das intenções de voto. Já o índice da candidata do PSOL, Heloísa Helena, caiu de 36% para 32%. Brancos e nulos somam 12% e os que não sabem ou não opinaram, 5%. **O senador Cristovam Buarque, candidato pelo PDT, teve alta de 29% para 32%.** A CNI explica que a pesquisa não traz a comparação com pesquisas anteriores para primeiro turno porque é a primeira vez que o Ibope utiliza a lista oficial de candidatos a presidente da República.
KLSum 1Comp
No segundo turno, as intenções de voto do presidente Lula caíram de 53% em junho para 50% em julho, enquanto o candidato Alckmin subiu de 29% para 36%. A pesquisa CNI/Ibope realizada em julho e divulgada nesta sexta-feira mostra que o presidente Luiz Inácio da Silva teria 44% dos votos no primeiro turno, enquanto o candidato tucano Geraldo Alckmin teria 25% das intenções de voto. Já o índice da candidata do PSOL, Heloísa Helena, caiu de 36% para 32%. Brancos e nulos somam 12% e os que não sabem ou não opinaram, 5%. **senador Cristovam Buarque teve alta de 29% para 32%.** A CNI explica que a pesquisa não traz a comparação com pesquisas anteriores para primeiro turno porque é a primeira vez que o Ibope utiliza a lista oficial de candidatos a presidente da República.

Quadro 28 – Exemplo 3 de sumários extractivos e respectivos compressivos por meio da coleção de ID C28 do córpus CSTNews-Update.

KLSum
KATOWICE, Polônia - A seleção brasileira de vôlei mais uma vez mostrou o porquê de ser o melhor time do mundo ao vencer a Rússia por 3 sets a 1, com parciais de 18 a 25, 25 a 23, 28 a 26 e 25 a 22, na tarde deste domingo, em Katowice, Polônia, na final da Liga Mundial de vôlei, conquistando o heptacampeonato da competição. **E agora com mais esse título na bagagem, a seleção chega ainda com mais moral para disputar o ouro nos Jogos Pan-Americanos**.
KLSum 1Comp
KATOWICE, Polônia - A seleção brasileira de vôlei mais uma vez mostrou o porquê de ser o melhor time do mundo ao vencer a Rússia por 3 sets a 1, com parciais de 18 a 25, 25 a 23, 28 a 26 e 25 a 22, na tarde deste domingo, em Katowice, Polônia, na final da Liga Mundial de vôlei, conquistando o heptacampeonato da competição. **seleção chega ainda com mais moral para disputar ouro nos Jogos Pan-Americanos.**

Para realização desses experimentos, foram utilizadas ferramentas de processamento textual, como métodos para radicalização e identificação automática de subtópicos, bem como uma lista de *stopwords* disponíveis na biblioteca NLTK (*Natural Language*

ToolKit[4]) para a linguagem Python.

Na Tabela 10, são apresentados os resultados dos métodos avaliados no córpus da DUC 2007. Para organização dos resultados, foi adotada a mesma estrutura definida na Tabela 8. Assim, os métodos e as métricas de avaliação são dispostos nas linhas e colunas, respectivamente. Por exemplo, para o método DualSum, foi aferido os valores de F-1 para ROUGE-1 e ROUGE-2 de 0,335 e 0,083, respectivamente.

Tabela 10 – Avaliação dos métodos de SAA extrativos investigados neste trabalho no córpus da DUC 2007.

	ROUGE-1			**ROUGE-2**				
Método	**Pre**	**Cob**	**F-1**	**Pre**	**Cob**	**F-1**	**NR-Pir**	**NR-Res**
DualSum*+SUB	0,338	0,341	0,339	0,091	0,092	0,091	6,651	1,097
Fator de Novidade*	0,322	0,329	0,325	0,083	0,085	0,084	2,446	0,292
DualSum*	0,334	0,337	0,335	0,083	0,084	0,083	6,551	1,075
KLSum*	0,313	0,318	0,315	0,077	0,078	0,078	2,307	0,263
Fator de Novidade+SUB	0,341	0,344	0,342	0,076	0,077	0,076	6,816	1,126
Posição Binário	0,341	0,347	0,344	0,073	0,074	0,074	2,135	0,227
Posição Direta	0,341	0,347	0,344	0,073	0,074	0,074	2,135	0,227
Posição Geométrico	0,341	0,347	0,344	0,073	0,074	0,074	2,135	0,227
Posição Invertido	0,341	0,347	0,344	0,073	0,074	0,074	2,135	0,227
Posição Binário +SUB	0,341	0,347	0,344	0,073	0,074	0,074	2,135	0,227
Posição Direta +SUB	0,341	0,347	0,344	0,073	0,074	0,074	2,135	0,227
Posição Geométrico +SUB	0,341	0,347	0,344	0,073	0,074	0,074	2,135	0,227
Posição Invertido +SUB	0,341	0,347	0,344	0,073	0,074	0,074	2,135	0,227
KLSum*+EN	0,300	0,306	0,303	0,072	0,073	0,072	2,202	0,242
KLSum-E	0,282	0,288	0,284	0,069	0,070	0,069	6,834	1,130
PageRank(B)	0,299	0,305	0,301	0,063	0,065	0,064	2,118	0,225
Fator de Novidade+EN	0,303	0,306	0,304	0,064	0,064	0,064	2,036	0,208
KLSum-Sub	0,330	0,336	0,332	0,063	0,064	0,064	6,591	1,081
PageRank(B)+SUB	0,301	0,308	0,304	0,060	0,062	0,061	2,064	0,214
PageRank+SUB	0,303	0,310	0,306	0,059	0,060	0,059	1,983	0,196
PNR^2	0,302	0,307	0,304	0,057	0,058	0,058	1,419	0,086
PNR^2+SUB	0,302	0,307	0,304	0,057	0,058	0,058	1,417	0,085
PageRank	0,295	0,299	0,296	0,053	0,053	0,053	1,858	0,171

De forma diferente aos resultados para a língua Portuguesa com o córpus CSTNews-Update, uma versão do método DualSum obteve resultados mais significativos em relação aos demais. Isso se explica possivelmente, sobretudo, pelo maior volume de textos presentes no córpus da DUC (que varia entre 8 a 10 textos por grupo de texto em uma coleção). Dessa forma, o modelo de distribuição de tópicos pode ser mais efetivo. Entretanto, o melhor método em relação à Noveau-ROUGE foi o KLSum-E, que é uma aproximação do modelo de distribuição de tópicos do DualSum por meio da analogia dessa informação

[4] <http://www.nltk.org/>

com a distribuição de subtópicos, o que demonstra que tal abordagem pode, de forma eficiente, auxiliar a produção de melhores sumários de atualização.

Considerando a distinção dos métodos avaliados para a língua Inglesa e Portuguesa, pode-se também observar que, em geral, os métodos mais simples não apresentaram resultados melhores por meio da inserção de conhecimento linguístico em relação aos métodos tradicionais, como ocorreu de forma distinta durante a avaliação por meio do CSTNews-Update. Por exemplo, os métodos Fator de Novidade e Posição Binário, Direta, Geométrico e Invertido foram ligeiramente superiores às respectivas versões com subtópicos (+SUB). Isso pode ser interpretado com o fato de que conhecimento linguístico pode ser mais preciso, auxiliando a produção de sumários mais informativos por meio de abordagens mais simples em volume de dados menores.

É importante também observar que o métodos KLSum-E, que emprega a distribuição de subtópicos em uma coleção textual como um estimador da distribuição de tópicos textuais, apresentou valor de Noveau-ROUGE superior ao DualSum*+SUB (que corresponde ao melhor modelo investigado do DualSum). Isso demonstra, de forma análoga ao encontrado na avaliação para a língua Portuguesa, que a segmentação de subtópicos pode efetivamente contribuir para a produção de sumários mais informativos. Além disso, ressalta-se novamente que o método KLSum-E exige menor tempo e recursos computacionais, como memória, em relação ao DualSum.

De forma similar à disposição dos resultados dos experimentos com o córpus da DUC 2007, são apresentadas as avaliações por meio dos córpus da TAC 2008 e TAC 2009 nas Tabelas 11 e 12, respectivamente. Os métodos investigados também foram inversamente ordenados pelos valores de F-1 para ROUGE-2 e alguns valores para outras métricas foram formatados em negrito para enfatizar informações relevantes que serão discutidas posteriormente.

Uma diferença saliente em relação aos resultados computados por meio dos córpus da TAC (2008 e 2009) em relação à DUC é a diferença de desempenho do DualSum em frente às variações propostas para o KLSum, como o KLSum-Sub e KLSum-E, que apresentaram resultados superiores para todas as métricas.

Esses experimentos foram interessantes, sobretudo, para analisar a diferença de comportamento dos métodos investigados em cenários diferentes. Um característica interessante em relação às diferenças de performances dos métodos nos córpus para a língua Inglesa e no CSTNews-Update são os resultados do método DualSum e respectivas variações. Esses métodos apresentaram resultados melhores para a língua inglesa. Isso de certa forma era esperado, pois a quantidade de dados (textos-fonte) disponíveis nesses córpus é maior do que no CSTNews-Update, o que contribui diretamente na qualidade do modelo de distribuição de tópicos inferido pelo modelo. Ressalta-se que para o DualSum são definidos quatro tipos principais de tópicos, referenciados como ϕ^G, ϕ^{Cd}, ϕ^{Ac} e ϕ^{Bc}. Segundo

Tabela 11 – Avaliação dos métodos de SAA extrativos investigados neste trabalho no córpus da TAC 2008.

	ROUGE-1			ROUGE-2				
Método	**Pre**	**Cob**	**F-1**	**Pre**	**Cob**	**F-1**	**NR-Pir**	**NR-Res**
KLSumS*	0,322	0,334	0,328	0,076	0,079	0,077	2,113	0,219
KLSum+EN*	0,308	0,320	0,314	0,075	0,079	0,077	2,170	0,230
KLSum*	0,308	0,321	0,314	0,075	0,078	0,076	2,160	0,228
DualSum*	0,302	0,314	0,308	0,070	0,073	0,072	2,022	0,198
PageRank	0,309	0,326	0,317	0,067	0,072	0,070	2,040	0,204
Fator de Novidade	0,311	0,322	0,317	0,068	0,070	0,069	2,062	0,210
Fator de Novidade+EN	0,309	0,321	0,315	0,066	0,069	0,067	2,033	0,205
PageRank+SUB	0,304	0,316	0,310	0,061	0,063	0,062	1,932	0,184
PageRank(B)+SUB	0,301	0,313	0,307	0,060	0,062	0,061	1,911	0,180
Fator de Novidade+SUB	0,301	0,309	0,305	0,057	0,058	0,058	1,821	0,164
PNR^2+SUB	0,301	0,309	0,305	0,057	0,058	0,058	1,269	0,055
PNR^2	0,301	0,309	0,305	0,057	0,058	0,058	1,251	0,052
KLSum-E	0,276	0,287	0,281	0,051	0,053	0,052	1,737	0,147

Tabela 12 – Avaliação dos métodos de SAA extrativos investigados neste trabalho no córpus da TAC 2009.

	ROUGE-1			ROUGE-2				
Método	**Pre**	**Cob**	**F-1**	**Pre**	**Cob**	**F-1**	**NR-Pir**	**NR-Res**
Ensemble	0,344	0,356	0,349	0,079	0,082	0,080	2,174	0,232
Fator de Novidade	0,325	0,336	0,330	0,077	0,080	0,078	2,248	0,248
KLSum-Sub*	0,328	0,339	0,333	0,076	0,078	0,077	2,086	0,213
Fator de Novidade+EN	0,320	0,331	0,325	0,074	0,077	0,075	2,202	0,239
DualSum*	0,312	0,323	0,317	0,073	0,076	0,074	2,078	0,210
KLSum*	0,312	0,323	0,317	0,072	0,075	0,074	2,099	0,215
KLSum+EN*	0,312	0,323	0,317	0,072	0,075	0,073	2,101	0,216
KLSum+SUB*	0,311	0,321	0,316	0,060	0,062	0,060	1,834	0,165
KLSum-E*	0,273	0,283	0,277	0,049	0,051	0,050	1,725	0,145
Fator de Novidade+SUB	0,288	0,296	0,292	0,049	0,050	0,050	1,740	0,149
PNR^2+SUB	0,237	0,243	0,240	0,024	0,025	0,024	1,323	0,067
PNR^2	0,223	0,228	0,225	0,020	0,021	0,021	1,262	0,054

Delort e Alfonseca (2012), o primeiro tópico pode ser modelado previamente a partir de uma coleção textual diferente dos textos-fonte, visando diminuir o tempo computacional necessário para processamento. Entretanto, os três últimos tipos de tópicos são extraídos diretamente dos textos-fonte. Assim, tendo em vista que a quantidade de textos no córpus CSTNews-Update é inferior em relação aos córpus para a língua inglesa, provavelmente o modelo de distribuição de tópicos aprendido pelo método pode ser menos discriminativo.

5.5 Considerações finais

Neste capítulo, foram introduzidos os recursos e métodos desenvolvidos para a língua Portuguesa no âmbito da SAA e SAA Compressiva que foram desenvolvidos neste trabalho. Acredita-se que tais contribuições possam estimular a ampliar o escopo da investigação na área da SA para esse idioma, para o qual a maioria dos trabalhos nesse campo de investigação encontra-se na SA Mono e Multidocumento.

Os métodos de SAA investigados e propostos foram avaliados com quatro córpus, sendo um para a língua Portuguesa, o CSTNews-Update, que foi desenvolvido durante o andamento deste trabalho. Essa metodologia de avaliação permitiu analisar as diferenças de desempenho das distintas abordagens de representação textual para a SAA em distintas variáveis, como idioma e quantidade de textos-fonte. Por exemplo, o CSTNews-Update é constituído por textos em língua Portuguesa e cada grupo textual possui 2 ou 3 textos-fonte. Por outro lado, o córpus da TAC 2009, para a língua Inglesa, possui 10 textos-fonte em cada grupo textual.

Por meio dos resultados computados por meio da ROUGE e Noveau-ROUGE para os métodos investigados, pode-se concluir que, em geral, a segmentação e aplicação de subtópicos auxilia mais efetivamente diferentes abordagens de SAA, sobretudo àquelas mais simples e em cenários nos quais há um menor volume de textos-fonte, como no caso do CSTNews-Update. Além disso, é importante ressaltar que o método KLSum-E e KLSum-Sub, que são constituídos por uma análise da distribuição de subtópicos nos textos-fonte que visa simplificar o modelo de tópicos textuais empregado pelo DualSum, apresentaram resultados muito satisfatórios, que são superiores aos demais métodos investigados em alguns casos.

Por meio dos exemplos de sumários apresentados no Quadro 29, pode-se observar que apenas PNR^2+SUB+CST apresenta um sumário com maiores problemas de encadeamento lógico das sentenças. Ressalta-se que, dos métodos apresentados, esse foi o que obteve valores ROUGE e Noveau-ROUGE menos relevantes. Nos dois primeiros sumários, encontram-se informações mais gerais sobre o evento descrito nos textos-fonte e, posteriormente, assim como no terceiro sumários, mais detalhes sobre a partida de futebol reportada.

O métodos KLSum e DualSum* não adotam o procedimento para remoção de conteúdo redundante descrito na Seção 5.1.9. Entretanto, é importante salientar que o conteúdo de seus respectivos sumários não apresentam muito conteúdo redundante. Isso já era esperado, pois, usualmente, métodos baseados em modelos gerativos ou de distribuição de tópicos textuais lidam com esse tipo de fenômeno internamente.

Quadro 29 – Exemplo de sumários gerados pelos métodos desenvolvidos neste trabalho.

KLSum
Para a surpresa da maioria, a Seleção Brasileira apresentou um futebol de alto nível e com propriedade aplicou uma goleada de 3 a 0 na Argentina para conquistar o título da Copa América de 2007. Ainda na pressão, o Brasil chega com perigo na jogada de Maicon, que deu uma meia lua na marcação e cruzou para Robinho. O Brasil é a melhor seleção das américas pela oitava vez. Mas, para a alegria dos brasileiro e para apimentar ainda mais a rivalidade entre os países, foi a canarinho quem balançou as redes. Vagner Love fez lindo passe para Daniel Alves,
Ensemble
Para a surpresa da maioria, a Seleção Brasileira apresentou um futebol de alto nível e com propriedade aplicou uma goleada de 3 a 0 na Argentina para conquistar o título da Copa América de 2007. Na seqüência disparou uma bomba e Doni fez linda defesa, evitando o empate. Ainda na pressão, o Brasil chega com perigo na jogada de Maicon, que deu uma meia lua na marcação e cruzou para Robinho. O Brasil é a melhor seleção das américas pela oitava vez. Ayala deu o carrinho para fazer o corte e tocou para dentro do seu próprio gol. Dentro da
DualSum*
Aplicado taticamente, com uma marcação forte no meio e na defesa e um toque de bola rápido e preciso no ataque, o Brasil chegou ao segundo gol ainda no primeiro tempo, com uma mãozinha ou melhor, um pezinho - de Ayala, que marcou contra após cruzamento de Maicon, aos 39. Vágner Love puxou um contra-ataque rápido e tocou na medida, nas costas da zaga, para Daniel Alves, que chutou forte, colocado, sem chances para Abbondanzieri. Em 2004, também pela Copa América, os argentinos venciam por 2 a 1, quando Tevez resolveu fazer graça perto da bandeirinha de escanteio. As seleções de
PNR^2+SUB+CST
Ao que parece, contra a Argentina, um time B é mais do que o suficiente. Dessa vez, nem foi necessária, a presença dos astros de Barcelona e Milan. Impecável, a Seleção, comandada por Ronaldinho Gaúcho e Kaká aplicou uma goleada humilhante por 4 a 1 e trouxe a taça. O outro revés aconteceu em 2005, na Copa das Confederações, na Alemanha. No final, vitória brasileira. No ataque seguinte, Adriano, que terminou a competição como artilheiro, com sete gols, empatou e levou a decisão para os pênaltis. Em 2004, também pela Copa América, os argentinos venciam por 2 a 1, quando

CAPÍTULO

6

CONCLUSÃO

Neste capítulo, serão apresentadas as considerações finais deste trabalho de doutorado, as principais contribuições, limitações e sugestões de trabalhos futuros. Para uma melhor organização dos temas abordados, esse capítulo será organizado em quatro seções, que serão descritas a seguir. Nas Seções 6.1 e 6.2, dissertara-se sobre as conclusões relacionadas aos métodos de Compressão Sentencial (CS) e Sumarização Automática de Atualização (SAA), extrativos e compressivos, que foram desenvolvidos e avaliados neste trabalho, respectivamente. As principais contribuições acadêmicas e técnicas, sobretudo para língua Portuguesa, para as duas áreas de pesquisa abordadas neste trabalho serão descritas na Seção 6.3. Por fim, as limitações e trabalhos futuros serão apresentados na Seção 6.4.

6.1 Considerações sobre os métodos de Compressão Sentencial

Neste trabalho foram desenvolvidos métodos de CS para a língua Portuguesa por meio da abordagem de Deleção, na qual a sentença comprimida é produzida após a remoção de alguns itens lexicais. Essa abordagem foi adota porque é a mais difundida na literatura, como pode ser observado pelos trabalhos de Knight e Marcu (2000), Madnani *et al.* (2007), Berg-Kirkpatrick, Gillick e Klein (2011), Li *et al.* (2013), Qian e Liu (2013), Almeida e Martins (2013), Filippova *et al.* (2015).

Para construção desses métodos, optou-se por utilizar algoritmos para o problema de classificação em aprendizado de máquina. Dessa forma, a compressão de uma sentença s com n itens lexicais ocorre após o mesmo número de decisões (ou classificações) individuais do modelo compressor, de forma que, para cada iteração i, é decidido se o respectivo item lexical na i-ésima posição será removido ou não de s. Ressalta-se que tal modelagem

do problema de CS também vem sendo empregada em trabalhos mais recentes na literatura, como Filippova *et al.* (2015), que adotam técnicas de *Deep Learning* e reportagem resultados muito bons.

É importante salientar que os métodos de CS desenvolvidos neste trabalho objetivaram uma respectiva aplicação posterior em métodos de SAA visando estende-los para a abordagem de síntese Compressiva, que possibilita tratar algumas limitações dos métodos baseados na síntese Extrativa. Por exemplo, uma vez que não há alteração das sentenças selecionadas para o sumário por meio da síntese extrativa, alguns segmentos sentenciais redundantes ou menos relevantes podem ocorrer, o que inibe a adição de novas sentenças e reduz à respectiva informatividade do sumário sendo produzido. Assim, por meio de métodos de CS, podem-se remover esses segmentos das sentenças visando à produção de sumários mais informativos e coesos.

Para comprimir sentenças no âmbito da SA, além de informações advindas da própria sentença alvo com algum processamento, como a aplicação de um Parser Sintático, podem-se empregar outros níveis de informação para auxiliar as decisões do modelo compressivo, tais como o sumário sendo construído e/ou o texto-fonte de origem da sentença alvo. Nesse contexto, os primeiros experimentos de CS foram realizados visando identificar o impacto desses três níveis de atributos (Sentença, Documento e Sumário) para a qualidade dos modelos de compressão. Após o treinamento e avaliação dos modelos compressivos, não foram identificadas diferenças significativas entre esses diferentes níveis de informação para composição dos atributos para os algoritmos de aprendizado de máquina adotados. Assim, pôde concluir que os atributos advindos da própria sentença alvo são suficientes para a geração de boas compressões.

Como descrito anteriormente, adotou-se uma modelagem de CS na qual uma sentença é comprimida após decisões individuais do classificador para cada respectivo item lexical. Nesse cenário, é intuitivo perceber que a classificação (remoção ou não) do i-ésimo item lexical é influenciada por decisões prévias e que, além disso, tal classificação irá influenciar posteriores respostas do modelo compressivo. Isso também pôde ser verificado após a análise da importância dos atributos adotados pelos modelos de aprendizado de máquina, na qual foi observado que um dos atributos mais relevantes é o *previous-token-removed*, que é a indicação se o item lexical imediatamente anterior foi removido ou não da sentença. Ressalta-se que esse tipo de atributo pode ser corretamente identificado durante a etapa de treinamento, porém, durante o cenário de aplicação do modelo compressivo, tal informação será gerada pelo próprio modelo.

Dada às características supracitadas, pode-se observar que o modelo compressivo está suscetível a erros em cadeia, de forma que uma resposta errada, eventualmente, produzirá mais erros em sequência. Esse problema foi tratado por meio de duas perspectivas distintas. Na primeira, assumiu-se que os erros gerados pelo classificador poderiam ser

originados por valores inconsistentes dos atributos de origem sintática, uma vez o conjunto de atributos inicialmente proposto eram extraídas após a geração de uma Árvore de Dependências Sintáticas, que é gerada automaticamente para cada sentença de entrada. Assim, foi proposto um novo conjunto de atributos que requer apenas a identificação das etiquetas morfossintática de cada item lexical, cujo processo de identificação automático é menos complexo do que aplicação de um Parser Sintático. Já na segunda abordagem, visando à diminuição da propagação de erros de classificação, foram realizados experimentos por meio dos algoritmos CRF (Conditional Random Fields) e o DAGGER citepRoss2011, que são técnicas mais direcionadas à Classificação Estruturada, na qual a resposta final esperada pelo classificador depende possui uma cadeia (estrutura) de classificações.

Após a avaliação dos modelos desenvolvidos em um segundo experimento, foi constatado que os modelos estruturados apresentaram sentenças comprimidas com maior qualidade, sobretudo o modelo de CRF. Além disso, observou-se também que os atributos extraídos por meio de um processamento mais simples das sentenças também contribuíram efetivamente para os modelos treinados. Após a comparação desses modelos compressivos com trabalhos da literatura, identificou-se que os resultados do modelo CRF foram superiores aos do método de Filippova *et al.* (2015), que emprega uma abordagem de *Deep Learning*.

6.2 Considerações sobre os métodos de Sumarização Automática

Tendo em vista a hipótese principal deste trabalho, de que alguns conhecimentos linguísticos em conjunto com distintas abordagens de representação textual podem auxiliar a produção de sumários mais informativos, foram investigadas variadas técnicas de modelagem textual e seleção de conteúdo adotadas em métodos SAA. Entre as técnicas selecionadas, encontram-se métodos baseados em características posicionais, ranqueamento superficial de sentenças, algoritmos de grafo, modelos gerativos de n-gramas e distribuição de tópicos, bem como um modelo *ensemble*. Dessa forma, foi possível verificar o impacto do conhecimento linguístico em métodos com níveis de processamento computacional e análise linguística variados. Por exemplo, o método Fator de Novidade requer pouco tempo computacional de processamento e baseia-se em uma análise superficial do vocabulário presente nos textos-fonte. Por outro lado, o método DualSum faz uso de uma técnica mais refinada, que visa analisar a distribuição de tópicos textuais de categorias distintas, que, consequentemente, requer maior tempo de processamento computacional.

Para cada abordagem de SAA investigada, foram propostas maneiras para incorporar um ou mais conhecimentos linguísticos. Por exemplo, para o método KLSum, foram desenvolvidos duas versões com a análise de segmentos de subtópicos, referenciadas como

KLSum-SUB e KLSum-E, e uma versão com incorporação de entidades nomeadas, o KLSum+EN. Foram adotadas três informações semântico-discursivas, que correspondem aos segmentos de Subtópicos, relações discursivas baseadas na CST e Entidades Nomeadas. Esses conhecimentos linguísticos foram selecionados baseando-se em trabalhos prévios para a língua Portuguesa no âmbito da SA tradicional multidocumento, como Castro Jorge (2010), Cardoso (2014), Ribaldo, Cardoso e Pardo (2016), que evidenciaram a aplicabilidade e efetividade desse tipo de conhecimento.

Todos os métodos foram avaliados por meio da ROUGE (LIN, 2004) e Noveau-ROUGE (CONROY; SCHLESINGER; O'LEARY, 2011) por meio da abordagem de avaliação proposta por Louis e Nenkova (2009), na qual os textos-fonte são adotados como textos de referência para os sumários sendo avaliados. Para a língua Portuguesa, foi utilizado o córpus CSTNews-Update, que foi organizado durante este trabalho. Após a análise dos resultados, contatou-se que a segmentação dos subtópicos auxilia mais efetivamente os métodos investigados, sobretudo àqueles que apresentam procedimentos menos complexos, como métricas posicionais e o método de Fator de Novidade. Além disso, por meio desse conhecimento linguístico, foram também propostas duas variações para o método KLSum, o KLSum-Sub e KLSum-E, que, em alguns casos, apresentou resultados superiores ao método DualSum, que requer um procedimento de análise de tópicos textuais mais complexo, que é similar a um modelo de LDA.

Um resultado inesperado foi a relativa baixa qualidade dos sumários produzidos pelos métodos baseados em grafo modelados por meio de relações discursivas CST em relação aos métodos de grafo tradicionais. Por exemplo, o método PNR^2+CST, que emprega CST, e PNR^2+SUB+CST, que faz uso de subtópicos e CST, apresentaram resultados inferiores à versão original do PNR^{2}e PNR^2+SUB. Contudo, ressalta-se que os métodos de SAA dessa abordagem que foram investigados neste trabalho assumem um grafo completo, em que cada vértice (sentença) possui arestas para todos os demais. Em um grafo modelado por meio da CST, uma sentença possui aresta apenas com sentenças para as quais exista alguma relação discursiva. Dessa forma, o número menor de arestas no grafo pode ter afetado negativamente para a produção de sumários mais informativos.

Investigou-se também um método *Ensemble*, que emprega as abordagens de ranqueamento sentenciais menos complexos que foram investigados neste trabalho. Esse método obteve os melhores resultados, mesmo sem aplicação de algum conhecimento linguístico. Tal resultado foi esperado, pois não havia muita diferença entre os valores ROUGE computados entre os métodos, com ou sem conhecimento linguístico, e essa abordagem agrega as vantagens de todos os métodos investigados.

Os resultados para a língua Inglesa por meio dos córpus da DUC 2007 e TAC de 2008 e de 2009 foram similares aos obtidos por meio do CSTNews-UPDATE. Contudo, foi observada uma exceção para o método DualSum, que apresentou resultados

muito mais satisfatórios. Esse método baseia-se em um modelo LDA para distribuição de tópicos textuais que, comumente, apresenta resultados melhores em volume de dados maiores. Assim, tendo em vista que os córpus para a língua inglesa são maiores do que o CSTNews-UPDATE, possivelmente essas características dos conjuntos de dados influenciaram diretamente a qualidade do DualSum.

Além dos métodos de SAA baseados na síntese Extrativa, foi proposta uma arquitetura de síntese Compressiva, que pode ser facilmente acoplada em métodos extrativos. Essa arquitetura é composta por três principais módulos, que correspondem às fases de Seleção, Compressão e Síntese. No primeiro módulo, pode-se adotar algum método de SA extrativo. Na etapa de compressão, que corresponde à produção de versões comprimidas das sentenças selecionadas previamente, foi utilizado o melhor modelo compressivo desenvolvido neste trabalho. Por fim, na etapa de Síntese, em que se identifica a sentença mais adequada para o sumário, que pode ser a sentença original ou alguma versão comprimida, foram investigadas diferentes abordagens, que se constituem na análise de uma ou mais versões comprimidas e a consideração ou não da sentença original. Na avaliação desses métodos, foi constatado que o melhor modelo de sumarização compressiva foi aquele que considera somente a sentença original e a menor versão comprimida. Em outras palavras, esse modelo pode, eventualmente, utilizar uma sentença original (sem compressão) para produzir o sumário.

Os resultados avaliados para os métodos Compressivos, em termos de ROUGE e Noveau-ROUGE, não foram superiores aos respectivos métodos extrativos. Entretanto, é importante salientar que foi utilizada uma metodologia de avaliação baseada na comparação dos sumários com os respectivos textos-fonte, que foi proposta por Louis (2014). Embora essa metodologia seja muito factível para a análise de métodos de SA extrativos, pode penalizar sumários automáticos produzidos por meio da Síntese Compressiva. Porém, ressalta-se que as diferenças de valores ROUGE não foram muito significativa. Além disso, a arquitetura de SA Compressiva pode ser investigada em outros métodos de SA.

6.3 Contribuições

As principais contribuições desse trabalho, sobretudo para língua Portuguesa, pautam-se nos recursos, ferramentas e métodos adaptados e desenvolvidos que foram disponibilizadas para desenvolver futuras pesquisas no âmbito da SAA para esse idioma.

Para a área de Compressão de Sentenças, foram compilados dois córpus com pares de sentenças originais e respectivas versões comprimidas, o córpus Pares-PCSC e Pares-G1. Esses recursos podem ser utilizados para pesquisas futuras em CS e Sumarização Compressiva para a língua Portuguesa.

Por meio dos córpus supracitados, foram desenvolvidos diversos modelos compres-

sivos baseados em técnicas de aprendizado de máquina, que podem ser aplicados em inúmeras aplicações, como Sumarização Automática Compressiva, Análise Automática de Legendas, entre outras. Ressalta-se que um dos modelos compressivos propostos neste trabalho apresentaram resultados superiores ao trabalho de Filippova *et al.* (2015).

No contexto da SAA, foi organizado um novo córpus, o CSTNews-UPDATE, que é contituído por uma diferente disposição do córpus CSTNews, que vêm sendo investigado em diversas investigações de SA mono e multidocumento. Além desse recurso, foi também proposta e analisada uma metodologia de avaliação para a tarefa de SAA por meio desse córpus.

Foram investigados diversos métodos de SAA e três tipos de conhecimento linguísticos, a segmentação de Subtópicos, teorias discursivas baseadas na CST e o uso de Entidades Nomeadas para incorporar técnicas baseadas em análise de vocabulário. Por meio da avaliação desses métodos, para a língua Portuguesa e Inglesa, constatou-se que a distribuição de Subtópicos é mais efetiva para diversas abordagens de SAA.

6.4 Limitações e trabalhos futuros

Os métodos de CS desenvolvidos neste trabalho baseiam-se em uma abordagem muito direcionada a técnicas de Aprendizado de Máquina, que vem de encontro a trabalhos mais recentes, como o método proposto por Filippova *et al.* (2015). Dessa forma, por exemplo, foi desconsiderada a aplicação de um Modelo de Língua, que foi adotada por alguns trabalhos prévios, como Madnani *et al.* (2007), Berg-Kirkpatrick, Gillick e Klein (2011), Li *et al.* (2013), visando à redução produção de sentenças agramaticais geradas pelos métodos. Assim, tendo em vista que, embora o modelo CRF proposto tenha apresentado resultados muito satisfatórios, algumas sentenças produzidas pelos métodos desenvolvidos ainda possuem alguns erros, pretende-se acoplar um modelo de Modelo de Língua nos métodos de CS propostos visando à produção de sentenças comprimidas com maior qualidade gramatical.

Além das características supracitadas, os modelos de CS investigados não foram submetidos a taxas de compressão. Ou seja, dada uma sentença de entrada, os modelos compressivos produzem uma respectiva versão comprimida a partir da análise sentencial e sem restrições externas, como o nível de compressão desejado. Essa restrição não foi considerada durante o trabalho, pois o objetivo dos métodos foi produzir a menor versão comprimida das sentenças baseando-se na observação dos córpus compilados (treinamento dos modelos de Aprendizado de Máquina).

Os métodos de SAA baseados na síntese Compressiva foram desenvolvidos por meio de uma Arquitetura que pode ser facilmente incorporada em métodos diversos de SA que fazem uso de somente a síntese Extrativa. Tal arquitetura é definida pelos seguintes

três componentes principais: (i) etapa de seleção (métodos de SA); (ii) um método de CS, para produzir versões comprimidas das sentenças selecionadas; e (iii) fase de síntese, na qual o método identifica qual a sentença (original ou alguma versão comprimida) mais adequada para o sumário. Contudo, tendo em vista que não se constatou grandes avanços de informatividade (ROUGE e Noveau-ROUGE) dos sumários compressivos gerados em relação aos sumários extrativos, pretende-se investigar técnicas mais robustas para compor os componentes da Arquitetura Compressiva proposta, sobretudo, para o terceiro módulo proposto.

Durante o desenvolvimento dos métodos de SAA mais informados, foram empregados três conhecimentos linguísticos distintos, que foram selecionados baseando-se em resultados prévios da literatura. Porém, após a avaliação dos métodos, não foram identificados ganhos significativos na informatividade dos sumários. Dessa forma, possibilidades futuras de pesquisa, englobam a aplicação de outras modelagens linguísticas. Além disso, pode-se também adotar outras abordagens de representação textual para aplicar algum conhecimento linguístico em métodos de SAA. Tal proposta é possivelmente interessante pois os métodos desenvolvidos neste trabalho incorporam informação linguística em modelos de representação textual já adotados na literatura, que, eventualmente, já representam essas informações linguísticas de forma implícita. Por exemplo, observou-se que os conhecimentos linguísticos auxiliaram mais efetivamente os métodos com representações menos complexas, como o Fator de Novidade.

REFERÊNCIAS

AGOSTINI, V.; CONDORI, R. E. L.; PARDO, T. A. S. Automatic alignment of news texts and their multi-document summaries: Comparison among methods. In: **Proceedings of the 11st International Conference on Computational Processing of Portuguese (PROPOR)**. São Carlos, SP, Brazil: Springer, 2014. p. 220–231. Citado nas páginas 58 e 124.

AIRES, R. V. X. **Implementação, Adaptação, Combinação e Avaliação de Etiquetadores para o Português do Brasil**. Dissertação (Mestrado) — Instituto de Ciências Matemáticas e de Computação, 2000. Citado nas páginas 63 e 108.

ALEIXO, P.; PARDO, T. A. S. *CSTNews: Um Córpus de Textos Jornalísticos Anotados segundo a Teoria Discursiva Multidocumento CST (Cross-document Structure Theory)*. [S.l.], 2008. Citado nas páginas 26, 28, 30, 36, 42, 60, 64, 119 e 138.

ALMEIDA, M. B.; ALMEIDA, M. S. C.; MARTINS, A. F. T.; FIGUEIRA, H.; MENDES, P.; PINTO, C. A new multi-document summarization corpus for european portuguese. In: **Language Resources and Evaluation Conference (LREC)**. Reykjavik, Iceland: ELRA, 2014. p. 1–7. Citado nas páginas 38, 42, 61 e 64.

ALMEIDA, M. B.; MARTINS, A. F. T. Fast and robust compressive summarization with dual decomposition and multi-task learning. In: **Proceedings of the Annual Meeting of the Association for Computational Linguistics**. Sofia, Bulgaria: ACL, 2013. p. 196–206. Citado nas páginas 37, 46, 85, 87, 89, 91 e 155.

ANTIQUEIRA, L.; JR., O. N. O.; COSTA, L. d. F.; NUNES, M. d. G. V. A complex network approach to text summarization. **Information Sciences**, v. 179, n. 5, p. 584–599, 2009. ISSN 0020-0255. Citado na página 26.

BALDI, P.; ITTI, L. Of bits and wows: A bayesian theory of surprise with applications to attention. **Official Journal of the International Neural Network Society**, v. 23, p. 649–666, 2010. Citado na página 76.

BARZILAY, R.; ELHADAD, M. Using lexical chains for text summarization. In: **Proceedings of the Intelligent Scalable Text Summarization Workshop (ISTS)**. Madrid, Spain: ACL, 1997. p. 10–17. Citado na página 25.

BARZILAY, R.; MCKEOWN, K. R.; ELHADAD, M. Information fusion in the context of multi-document summarization. In: **Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics**. College Park, Maryland: ACL, 1999. (ACL '99), p. 550–557. ISBN 1-55860-609-3. Citado na página 26.

BAWAKID, A.; OUSSALAH, M. A semantic summarization system: University of birmingham at TAC 2008. In: **Proceedings of the first Text Analysis Conference (TAC)**. Gaithersburg, Maryland, USA: NIST, 2008. p. 6. Citado nas páginas 69, 70, 83 e 84.

BERG-KIRKPATRICK, T.; GILLICK, D.; KLEIN, D. Jointly learning to extract and compress. In: **Proceedings of the International Conference on Computational Linguistics (Coling)**. Portland, Oregon: ACL, 2011. p. 481–490. Citado nas páginas 37, 46, 85, 91, 155 e 160.

BHASKAR, P.; BANDYOPADHYAY, S. A query focused multi document automatic summarization. In: **Proceedings of the 24th Pacific Asia Conference on Language, Information and Computation**. Tohoku University, Sendai, Japan: Institute of Digital Enhancement of Cognitive Processing, Waseda University, 2010. p. 545–554. Citado na página 26.

BICK, E. **The Parsing System Palavras, Automatic Grammatical Analysis of Portuguese in a Constraint Grammar Framework**. [S.l.]: Aarhus University Press, 2000. Citado nas páginas 63, 86, 99, 100, 106 e 108.

BLEI, D. M.; NG, A. Y.; JORDAN, M. I. Latent dirichlet allocation. **Journal of Machine Learning Research**, JMLR, v. 3, p. 993–1022, 2003. ISSN 1532-4435. Citado nas páginas 60, 74 e 75.

BOUDIN, F.; EL-BÈZE, M.; MORENO, J. M. T. A scalable MMR approach to sentence scoring for multi-documentupdate summarization. In: **Proceedings of the 20th International Conference on Computational Linguistics (Coling) – Posters and Demonstrations**. Manchester, UK: ACL, 2008. p. 23–26. Citado nas páginas 26, 67, 68, 74, 83 e 84.

BRIN, S.; PAGE, L. The anatomy of a large-scale hypertextual web search engine. In: **Proceedings of 17th International World-Wide Web Conference (WWW)**. Beijing, China: ACM, 1998. p. 20. Citado nas páginas 68 e 129.

CAMARGO, R. T.; AGOSTINI, V.; FELIPPO, A. D.; PARDO, T. A. Manual typification of source texts and multi-document summariesalignments. **Procedia - Social and Behavioral Sciences**, v. 95, n. 0, p. 498–506, 2013. ISSN 1877-0428. Citado na página 26.

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**. Melbourne, Australia: ACM, 1998. p. 335–336. ISBN 1-58113-015-5. Citado nas páginas 25, 68, 80 e 84.

CARDOSO, P.; PARDO, T. A. S. Multi-document summarization using semantic discourse models. **Processamiento de Lenguaje Natural**, v. 56, p. 57–64, 2016. Citado nas páginas 26 e 48.

CARDOSO, P. C. F. **Exploração de Métodos de Sumarização Multidocumento com Base em Conhecimento Semântico-Discursivo**. Tese (Doutorado) — Instituto de Ciências Matemáticas e de Computação,, 2014. Citado nas páginas 34, 35, 36, 38, 39, 48, 58, 138 e 158.

CARDOSO, P. C. F.; MAZIERO, E. G.; Castro Jorge, M. L. R.; SENO, E. M. R.; FELIPPO, A. D.; RINO, L. H. M.; NUNES, M. d. G. V.; PARDO, T. A. S. CSTNews – a discourse-annotated corpus for single and multi-document summarization of news texts in

brazilian portuguese. In: **Anais do III Workshop "A RST e os Estudos do Texto"**. Cuiabá, MT, Brasil: [s.n.], 2011. p. 88–105. Citado nas páginas 26, 28, 30, 36, 42, 60, 61, 64, 119 e 138.

CARDOSO, P. C. F.; TABOADA, M.; PARDO, T. A. S. On the contribution of discourse structure to topic segmentation. In: **Proceedings of the 14th annual SIGdial Meeting on Discourse and Dialogue**. Metz, France: [s.n.], 2013. p. 92–96. Citado na página 63.

Castro Jorge, M. L.; PARDO, T. A. S. A generative approach for multi-document summarization using the noisy channel model. In: **Proceedings of the 3rd RST Brazilian Meeting**. Cuiabá/MT, Brazil: [s.n.], 2011. p. 75–87. Citado na página 130.

Castro Jorge, M. L. R. **Sumarização automática multidocumento: seleção de conteúdo com base no Modelo CST (Cross-document Structure Theory)**. Dissertação (Mestrado) — Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2010. Citado nas páginas 34, 36, 38, 48, 58, 61, 138 e 158.

COHN, T.; LAPATA, M. Sentence compression beyond word deletion. In: **Proceedings of the International Conference on Computational Linguistics (Coling)**. Manchester, UK: ACL, 2008. p. 137–144. Citado nas páginas 37, 85 e 91.

CONROY, J. M.; SCHLESINGER, J. D.; O'LEARY, D. P. Squibs: Nouveau-ROUGE: A novelty metric for update summarization. **Computational Linguistics**, v. 37, n. 1, p. 1–9, 2011. Citado nas páginas 51, 53, 142, 143 e 158.

DANG, H.; OWCZARZAK, K. Overview of the TAC 2009 summarization track. In: **Proceedings of the second Text Analysis Conference (TAC)**. Gaithersburg, Maryland USA: NIST, 2009. Citado na página 66.

DANG, H. T. Overview of DUC 2005. In: **Proceedings of the Document Understanding Conference (DUC)**. Vancouver, Canada: NIST, 2005. Citado nas páginas 54 e 55.

DANG, H. T.; OWCZARZAK, K. Overview of the TAC 2008 update summarization task. In: **Proceedings of the first Text Analysis Conference (TAC)**. Gaithersburg, Maryland USA: NIST, 2008. p. 1–16. Citado na página 66.

Daumé III, H.; MARCU, D. Bayesian query-focused summarization. In: **Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL**. [S.l.: s.n.], 2006. p. 305—312. Citado na página 26.

DELORT, J.-Y.; ALFONSECA, E. DualSum: a topic-model based approach for update summarization. In: **Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics (EACL)**. Avignon, France: ACL, 2012. p. 214–223. Citado nas páginas 30, 33, 34, 35, 36, 39, 40, 75, 76, 83, 84, 130, 132 e 152.

DU, P.; GUO, J.; ZHANG, J.; CHENG, X. Manifold ranking with sink points for update summarization. In: **Proceedings of the 19th ACM International Conference on Information and Knowledge Management**. Toronto, ON, Canada: ACM, 2010. p. 1757–1760. Citado nas páginas 34, 78, 79, 81, 82, 83 e 84.

EDMUNDSON, H. P. New methods in automatic extracting. **Journal of the Association for Computing Machinery (JACM)**, ACM, v. 16, n. 2, p. 264–285, 1969. Citado na página 25.

ERKAN, G.; RADEV, D. R. Lexrank: Graph-based lexical centrality as salience in text summarization. **Journal of Artificial Intelligence Research**, v. 22, n. 1, p. 457–479, 2004. Citado nas páginas 34, 39 e 126.

FELLBAUM, C. **WordNet An Eletronic Lexical Database**. [S.l.]: MIT Press, 1998. 423 p. Citado na página 69.

FILIPPOVA, K.; ALFONSECA, E. Fast k-best sentence compression. **CoRR**, abs/1510.08418, 2015. Citado na página 41.

FILIPPOVA, K.; ALFONSECA, E.; COLMENARES, C.; KAISER, L.; VINYALS, O. Sentence compression by deletion with LSTMs. In: **Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)**. Lisbon, Portugal: ACL, 2015. p. 360–368. Citado nas páginas 46, 85, 87, 88, 89, 94, 98, 111, 113, 114, 115, 116, 117, 155, 156, 157 e 160.

FILIPPOVA, K.; ALTUN, Y. Overcoming the lack of parallel data in sentence compression. In: **Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP)**. Seattle, Washington, USA: ACL, 2013. p. 1481–1491. Citado nas páginas 42, 85 e 94.

FONSECA, E. R.; ROSA, J. ao L. G. Mac-morpho revisited: Towards robust part-of-speech tagging. In: SOCIEDADE BRASILEIRA DE COMPUTAçãO. **Proceedings of the 9th Brazilian Symposium in Information and Human Language Technology (STIL)**. Fortaleza, CE, Brazil, 2013. p. 98–107. Citado na página 108.

GANTZ, J.; REINSEL, D. The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east. **IDC iView: IDC Analyze the Future**, v. 1, p. 11, 2012. Citado na página 25.

GAO, D.; LI, W.; ZHANG, R. Sequential summarization: A new application for timely updated twitter trending topics. In: **Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics – Short Papers**. Sofia, Bulgaria: ACL, 2013. v. 2, p. 567–571. Citado na página 84.

GILLICK, D.; FAVRE, B.; HAKKANI-TUR, D. The ICSI summarization system at TAC 2008. In: **Proceedings of the first Text Analysis Conference (TAC)**. Gaithersburg, Maryland, USA: NIST, 2008. p. 8. Citado nas páginas 77, 78, 83 e 84.

GILLICK, D.; FAVRE, B.; HAKKANI-TUR, D.; BOHNET, B.; LIU, Y.; XIE, S. The ICSI/UTD summarization system at TAC 2009. In: **Proceedings of the second Text Analysis Conference (TAC)**. Gaithersburg, Maryland, USA: NIST, 2009. p. 8. Citado nas páginas 78, 83 e 84.

HAGHIGHI, A.; VANDERWENDE, L. Exploring content models for multi-document summarization. In: **Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)**. Boulder, Colorado, USA: ACL, 2009. p. 362–370. ISBN 978-1-932432-41-1. Citado nas páginas 36 e 130.

HE, R.; QIN, B.; LIU, T. A novel approach to update summarization using evolutionary manifold-ranking and spectral clustering. **Expert Systems with Applications**, v. 39, n. 3, p. 2375 – 2384, 2012. ISSN 0957-4174. Citado nas páginas 34, 80, 81, 83 e 84.

HEARST, M. A. Texttiling: Segmenting text into multi-paragraph subtopic passages. **Computational Linguistics**, MIT Press, Cambridge, MA, USA, v. 23, n. 1, p. 33–64, 1997. ISSN 0891-2017. Citado na página 63.

HOVY, E.; LIN, C. yew; ZHOU, L. Evaluating DUC 2005 using basic elements. In: **Proceedings of DUC-2005**. Vancouver, Canada: NIST, 2005. p. 6. Citado nas páginas 50 e 51.

HUANG, L.; HE, Y. Corrrank: Update summarization based on topic correlation analysis. In: HUANG, D.-S.; ZHANG, X.; GARCíA, C. R.; ZHANG, L. (Ed.). **Advanced Intelligent Computing Theories and Applications with Aspects of Artificial Intelligence**. [S.l.]: Springer Berlin Heidelberg, 2010, (Lecture Notes in Computer Science, v. 6216). p. 641–648. ISBN 978-3-642-14931-3. Citado nas páginas 34, 35, 36, 39, 74, 83, 84 e 132.

JURAFSKY, D.; MARTIN, J. H. **Speech and Language Processing: an Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition**. 2nd. ed. [S.l.]: Pearson, 2009. (Prentice Hall). Citado na página 47.

KATRAGADDA, R.; PINGALI, P.; VARMA, V. Sentence position revisited: A robust light-weight update summarization baseline algorithm. In: **Proceedings of the Third International Workshop on Cross Lingual Information Access: Addressing the Information Need of Multilingual Societies (CLIAWS)**. Boulder, Colorado: ACL, 2009. p. 46–52. Citado nas páginas 39, 70, 72, 81, 83, 84, 123 e 124.

KAWAMOTO, D.; PARDO, T. A. S. Learning sentence reduction rules for brazilian portuguese. In: **Proceedings of the 7th International Workshop on Natural Language Processing and Cognitive Science**. Funchal, Madeira, Portugal: SCITEPRESS, 2010. p. 90–99. Citado nas páginas 86 e 89.

KNIGHT, K.; MARCU, D. Statistics-based summarization – step one: Sentence compression. In: **Proceedings of the AAAI Conference on Artificial Intelligence**. [S.l.: s.n.], 2000. p. 703–711. Citado nas páginas 37, 46, 85, 86, 89, 91 e 155.

KOCH, I. **Introdução à linguística textual**. [S.l.]: Contexto, 2009. Citado na página 58.

LANDAUER, T. K.; DUTNAIS, S. T. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. **Psychological review**, v. 104, p. 211–240, 1997. Citado nas páginas 60 e 73.

LEITE, D. S.; RINO, L. H. M.; PARDO, T. A. S.; NUNES, M. das G. V. Extractive automatic summarization: Does more linguistic knowledge make a difference? In: **Proceedings of the HLT/NAACL Workshop on TextGraphs-2: Graph-Based Algorithms for Natural Language Processing**. Rochester, NY, USA: ACL, 2007. p. 17–24. Citado nas páginas 36 e 38.

LERMAN, K.; MCDONALD, R. Contrastive summarization: An experiment with consumer reviews. In: **Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL) – Short Papers**. Boulder, Colorado: ACL, 2009. p. 113–116. Citado na página 49.

LESKOVEC, J.; MILIC-FRAYLING, N.; GROBELNIK, M. **Extracting Summary Sentences Based on the Document Semantic Graph**. [S.l.], 2005. 9 p. Citado nas páginas 34, 48 e 126.

LI, C.; LIU, F.; WENG, F.; LIU, Y. Document summarization via guided sentence compression. In: **Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)**. Seattle, Washington, USA: ACL, 2013. p. 490–500. Citado nas páginas 46, 85, 155 e 160.

LI, J.; LI, S. Evolutionary hierarchical dirichlet process for timeline summarization. In: **Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics – Short Papers**. Sofia, Bulgaria: ACL, 2013. p. 556–560. Citado na página 84.

LI, J.; LI, S.; WANG, X.; TIAN, Y.; CHANG, B. Update summarization using a multi-level hierarchical dirichlet process model. In: **Proceedings of the 24th International Conference on Computational Linguistics (Coling)**. Mumbai, India: ACL, 2012. p. 1603–1618. Citado nas páginas 39, 76, 83, 84 e 132.

LI, M.; VITáNYI, P. **An Introduction to Kolmogorov Complexity and Its Applications**. [S.l.]: Springer-Verlag, 2007. Citado na página 79.

LI, X.; DU, L.; SHEN, Y.-D. Graph-based marginal ranking for update summarization. In: **Proceedings of the 2011 SIAM International Conference on Data Mining**. Mesa, Arizona, USA: SIAM, 2011. p. 486–497. Citado nas páginas 34, 80, 83, 84 e 126.

_____. Update summarization via graph-based sentence ranking. **IEEE Transactions on Knowledge and Data Engineering**, IEEE Computer Society, Los Alamitos, CA, USA, v. 25, n. 5, p. 1162–1174, 2013. ISSN 1041-4347. Citado nas páginas 39, 81, 82, 83, 84 e 126.

LI, Y.; LI, S. Query-focused multi-document summarization: Combining a topic model with graph-based semi-supervised learning. In: **Proceedings of the 25th International Conference on Computational Linguistics (Coling) – Technical Papers**. Dublin, Ireland: Dublin City University and ACL, 2014. p. 1197–1207. Citado na página 34.

LIN, C.-Y. Rouge: A package for automatic evaluation of summaries. In: **Text Summarization Branches Out: Proceedings of the ACL-04 Workshop**. Barcelona, Spain: ACL, 2004. p. 74–81. Citado nas páginas 50, 142 e 158.

LIN, Z.; KAN, M.-Y. Timestamped graphs: Evolutionary models of text for multi-documentsummarization. In: **Proceedings of the Second Workshop on TextGraphs: Graph-Based Algorithms for Natural Language Processing**. Rochester, NY, USA: ACL, 2007. p. 25–32. Citado nas páginas 34, 39 e 126.

LONG, C.; HUANG, M.-L.; ZHU, X.-Y.; LI, M. A new approach for multi-document update summarization. **Journal of Computer Science and Technology**, Springer US, v. 25, n. 4, p. 739–749, 2010. ISSN 1000-9000. Citado nas páginas 34, 79, 80, 83 e 84.

LOUIS, A. A bayesian method to incorporate background knowledge during automatic text summarization. In: **Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics – Short Papers**. Baltimore, Maryland: ACL, 2014. p. 333–338. Citado nas páginas 76, 77, 83, 84 e 159.

LOUIS, A.; NENKOVA, A. Automatically evaluating content selection in summarization without human models. In: **Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP)**. Singapore: ACL, 2009. p. 306–314. Citado nas páginas 54, 143, 146, 148 e 158.

LUOTOLAHTI, J.; GINTER, F. Sentence compression for automatic subtitling. In: **Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)**. [S.l.]: ACL, 2015. p. 135–143. Citado na página 85.

MADNANI, N.; ZAJIC, D.; DORR, B.; AYAN, N. F.; LIN, J. Multiple alternative sentence compressions for automatic text summarization. In: **Proceedings of the Document Understanding Conference (DUC)**. Rochester, New York, USA: NIST, 2007. p. 8. Citado nas páginas 46, 85, 155 e 160.

MANI, I. **Automatic Summarization**. [S.l.]: John Benjamins Publishing Company, 2001. v. 3. Citado nas páginas 25, 35, 45, 47 e 48.

MANN, W. C.; THOMPSON, S. A. **Rhetorical Structure Theory: A Theory of Text Organization**. [S.l.], 1987. Citado nas páginas 59 e 63.

MARCU, D. From discourse structures to text summaries. In: **Proceedings of the Intelligent Scalable Text Summarization Workshop (ISTS)**. Madrid, Spain: ACL, 1997. p. 82–88. Citado na página 25.

MARGARIDO, P. R. A.; PARDO, T. A. S.; ANTONIO, G. M.; FUENTES, V. B.; AIRES, R.; ALUÍSIO, S. M.; FORTES, R. P. M. Automatic summarization for text simplification: Evaluating text understanding by poor readers. In: **Anais do VI Workshop em Tecnologia da Informação e da Linguagem Humana (TIL)**. Vila Velha, ES, Brazil: [s.n.], 2008. p. 26–28. Citado nas páginas 36, 38 e 138.

MARTINS, A. F. T.; SMITH, N. A. Summarization with a joint model for sentence extraction and compression. In: **Proceedings of the NAACL HLT Workshop on Integer Linear Programming for Natural Language Processing**. Boulder, Colorado: ACL, 2009. p. 1–9. Citado nas páginas 86, 87 e 89.

MAZIERO, E.; Castro Jorge, M. L. R.; PARDO, T. A. S. Revisiting cross-document structure theory for multi-document discourse parsing. **Information Processing & Management**, v. 50, n. 2, p. 297–314, 2014. Citado nas páginas 27 e 62.

MAZIERO, E. G.; Castro Jorge, M. L. d. R.; PARDO, T. A. S. Identifying multidocument relations. In: **Proceedings of the 7th International Workshop on Natural Language Processing and Cognitive Science – NLPCS**. Funchal, Madeira, Portugal: [s.n.], 2010. p. 60–69. Citado nas páginas 15, 57, 58 e 62.

MAZIERO, E. G.; PARDO, T. A. S. CSTParser–a multi-document discourse parser. In: **Proceedings of the PROPOR 2012 Demonstrations**. Coimbra, Portugal: Springer, 2012. p. 1–3. Citado na página 128.

MCDONALD, R. Discriminative sentence compression with soft syntactic evidence. In: **Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL)**. Trento, Italy: ACL, 2006. p. 297–304. Citado nas páginas 37, 85 e 91.

MCKEOWN, K.; RADEV, D. Generating summaries of multiple news articles. In: **Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval**. Seattle, WA, USA: ACM, 1995. p. 74–82. Citado nas páginas 26 e 45.

MIHALCEA, R.; CEYLAN, H. Explorations in automatic book summarization. In: **Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP–CoNLL)**. Prague, Czech Republic: ACL, 2007. p. 380–389. Citado na página 25.

MIHALCEA, R.; TARAU, P. Textrank: Bringing order into texts. In: **Proceedings of Empirical Methods in Natural Language Processing (EMNLP)**. Barcelona, Spain: ACL, 2004. p. 404–411. Citado na página 36.

MIKOLOV, T.; SUTSKEVER, I.; CHEN, K.; CORRADO, G. S.; DEAN, J. Distributed representations of words and phrases and their compositionality. In: BURGES, C. J. C.; BOTTOU, L.; WELLING, M.; GHAHRAMANI, Z.; WEINBERGER, K. Q. (Ed.). **Advances in Neural Information Processing Systems 26**. [S.l.]: Curran Associates, Inc., 2013. p. 3111–3119. Citado na página 88.

NASTASE, V. Topic-driven multi-document summarization with encyclopedic knowledge and spreading activation. In: **Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)**. Stroudsburg, PA, USA: ACL, 2008. p. 763–772. Citado nas páginas 34 e 36.

NENKOVA, A.; PASSONNEAU, R. Evaluating content selection in summarization: The pyramid method. In: **Proceedings of the Human Language Technology and Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL)**. Boston, USA: ACL, 2004. p. 145–152. Citado nas páginas 50, 51, 52, 70, 124 e 143.

NENKOVA, A.; VANDERWENDE, L. **The impact of frequency on summarization**. [S.l.], 2005. v. 1, 8 p. Citado na página 34.

NóBREGA, F. A. A.; AGOSTINI, V.; CAMARGO, R. T.; FELIPPO, A. D.; PARDO, T. A. S. Alignment-based sentence position policy in a news corpus for multi-document summarization. In: **Proceedings of the 11st International Conference on Computational Processing of Portuguese (PROPOR)**. São Carlos, SP, Brazil: Springer, 2014. p. 6–9. Citado na página 36.

NóBREGA, F. A. A.; PARDO, T. A. Explorando métodos de uso geral para desambiguação lexical de sentidos para a língua portuguesa. In: **Anais do 9o Encontro Nacional de Inteligência Artificial – ENIA**. Curitiba, PR, Brazil: [s.n.], 2012. p. 1–12. Citado na página 61.

OUYANG, Y.; LI, W.; LU, Q.; ZHANG, R. A study on position information in document summarization. In: **Proceedings of the 23rd International Conference on Computational Linguistics (Coling) – Posters**. Beijing, China: ACL, 2010. p. 919–927. Citado nas páginas 39, 72, 73, 81, 82, 83, 84, 121, 122 e 123.

OWCZARZAK, K.; DANG, H. Overview of the TAC 2010 summarization track. In: **Proceedings of the third Text Analysis Conference (TAC)**. Gaithersburg, Maryland, USA: NIST, 2010. Citado nas páginas 39 e 66.

_____. Overview of the TAC 2011 summarization track: Guided task and AESOP task. In: **Proceedings of the Text Analysis Conference (TAC)**. Gaithersburg, Maryland, USA: NIST, 2011. Citado na página 66.

QIAN, X.; LIU, Y. Fast joint compression and summarization via graph cuts. In: **Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)**. Seattle, Washington, USA: ACL, 2013. p. 1492–1502. Citado nas páginas 34, 46, 85 e 155.

RADEV, D. R. A common theory of information fusion from multiple text sources step one: Cross-document structure. In: **Proceedings of 1st ACL SIGDIAL Workshop on Discourse and Dialogue**. Hong Kong, China: ACL, 2000. p. 10. Citado nas páginas 17, 35, 56, 61 e 62.

RADEV, D. R.; MCKEOWN, K. R. Generating natural language summaries from multiple on-line sources. **International Journal of Computational Linguistics**, MIT Press, v. 24, n. 3, p. 470–500, 1998. ISSN 0891-2017. Citado nas páginas 26 e 45.

RATNAPARKHI, A. **A Maximum Entropy Model for Part-Of-Speech Tagging**. 1986. Citado nas páginas 63 e 108.

REEVE, L. H.; HAN, H. A term frequency distribution approach for the DUC-2007 update task. In: **Proceedings of Document Understanding Conference (DUC)**. Rochester, New York USA: [s.n.], 2007. p. 7. Citado nas páginas 67, 73, 75, 82, 83, 84 e 132.

RIBALDO, R.; AKABANE, A. T.; RINO, L. H. M.; PARDO, T. A. S. Graph–based methods for multi-document summarization: Exploring relationship maps, complex networks and discourse information. In: **Proceedings of the 10th International Conference on Computational Processing of Portuguese (PROPOR)**. Coimbra, Portugal: Springer, 2012. p. 260–271. Citado nas páginas 34, 35, 36, 38, 39, 48, 121, 126, 135 e 138.

RIBALDO, R.; CARDOSO, P. F.; PARDO, T. A. S. Exploring the subtopic-based relationship map strategy for multi-document summarization. **Journal of Theoretical and Applied Computing – RITA**, v. 23, n. 1, p. 183–211, 2016. Citado nas páginas 34, 35, 39, 48, 138 e 158.

RINO, L. H. M.; PARDO, T. A. S.; JR, C. N. S.; KAESTNER, C. A. A.; POMBO1, M. A comparison of automatic summarization systems for brazilian portuguese texts. In: **Proceedings of the 17th Brazilian Symposium on Artificial Intelligence (BRACIS)**. Sao Luis, Maranhao, Brazil: Springer Berlin Heidelberg, 2004. p. 235–244. Citado nas páginas 25, 36, 38 e 138.

ROSS, S.; GORDON, G. J.; BAGNELL, J. A. A reduction of imitation learning and structured prediction to no-regret online learning. In: **Proceedings of the 14th International Conference on Artificial Intelligence and Statistics (AISTATS)**. Fort Lauderdale, FL, USA: [s.n.], 2011. p. 627–635. Citado nas páginas 105 e 108.

SALTON, G.; WONG, A.; YANG, C. S. A vector space model for automatic indexing. **Communications of the ACM**, ACM, v. 18, n. 11, p. 613–620, 1975. Citado nas páginas 33, 68, 79, 80, 126 e 135.

Sobrevilla Cabezudo, M. A.; MAZIERO, E. G.; SOUZA, J. W. C.; DIAS, M. S.; CARDOSO, P. C. F.; BALAGE, P. P. F.; AGOSTINI, V.; NóBREGA, F. A. A.; BARROS, C. D.; FELIPPO, A. D.; PARDO, T. A. S. Anotação de sentidos de verbos em notícias jornalisticas em português do brasil. In: **Proceedings of the XII Encontro de Linguística de Corpus (ELC)**. Uberlândia, MG, Brazil: [s.n.], 2014. p. 1–7. Citado na página 61.

STEINBERGER, J.; JEVZEK, K. SUTLER: Update summarizer based on latent topics. In: **Proceedings of the second Text Analysis Conference (TAC)**. Gaithersburg, Maryland USA: NIST, 2009. Citado na página 34.

STEINBERGER, J.; JEžEK, K. Update summarization based on latent semantic analysis. In: MATOUšEK, V.; MAUTNER, P. (Ed.). **Text, Speech and Dialogue**. [S.l.]: Springer Berlin, Heidelberg, 2009, (Lecture Notes in Computer Science, v. 5729). p. 77–84. ISBN 978-3-642-04207-2. Citado nas páginas 39, 73, 74, 83, 84 e 132.

THADANI, K.; MCKEOWN, K. Sentence compression with joint structural inference. In: **Proceedings of the Seventeenth Conference on Computational Natural Language Learning**. Sofia, Bulgaria: ACL, 2013. p. 65–74. Citado nas páginas 37, 85, 87, 89 e 91.

TURNER, J.; CHARNIAK, E. Supervised and unsupervised learning for sentence compression. In: **Proceedings of the 43rd Annual Meeting on Association for Computational**. Ann Arbor: ACL, 2005. p. 290–297. Citado nas páginas 37, 85, 86, 89 e 91.

VANDERWENDE, L.; BANKO, M.; MENEZES, A. Event-centric summary generation. In: **Working notes of the Document Understanding Conference (DUC)**. Boston, USA: NIST, 2004. p. 1–6. Citado nas páginas 34 e 48.

VARMA, V.; BHARAT, V.; KOVELAMUDI, S.; BYSANI, P.; GSK, S.; N, K. K.; KUMAR, K. R. K.; MAGANTI, N. IIIT hyderabad at TAC 2009. In: **Proceedings of the second Text Analysis Conference (TAC)**. Gaithersburg, Maryland USA: NIST, 2009. p. 1–15. Citado nas páginas 34, 70, 71, 72, 83, 84 e 125.

WANG, D.; LI, T. Document update summarization using incremental hierarchical clustering. In: **Proceedings of the 19th ACM international conference on Information and knowledge management**. Toronto, ON, Canada: ACM, 2010. p. 279–288. ISBN 978-1-4503-0099-5. Citado nas páginas 34, 74, 75, 82, 83 e 84.

WANG, D.; ZHU, S.; LI, T.; GONG, Y. Multi-document summarization using sentence-based topic models. In: **Proceedings of the ACL-IJCNLP 2009 Conference – Short Papers**. Suntec, Singapore: ACL, 2009. p. 297–300. Citado na página 130.

WENJIE, L.; FURU, W.; QIN, L.; YANXIANG, H. PNR^2: ranking sentences with positive and negative reinforcement for query-oriented update summarization. In: **Proceedings of the 22nd International Conference on Computational Linguistics (Coling)**. Manchester, United Kingdom: ACL, 2008. p. 489–496. ISBN 978-1-905593-44-6. Citado nas páginas 34, 39, 68, 82, 83, 84, 126, 127 e 129.

WITTE, R.; KRESTEL, R.; BERGLER, S. Generating update summaries for DUC 2007. In: **Proceedings of the Document Understanding Conference (DUC)**. Rochester, New York USA: NIST, 2007. p. 5. Citado nas páginas 30 e 84.


ICMC
USP
SÃO CARLOS