Vinícius Camargo da Silva

Sumarização Extrativa de Texto Utilizando Modelos Aditivos Generalizados com Interações para Seleção de Sentenças

Bauru, SP, Brasil

2023

Vinícius Camargo da Silva

Sumarização Extrativa de Texto Utilizando Modelos Aditivos Generalizados com Interações para Seleção de Sentenças

Dissertação apresentada como parte dos requisitos para obtenção do título de Mestre em Ciência da Computação, junto ao Programa de Pós-Graduação em Ciência da Computação, da Faculdade de Ciências da Universidade Estadual Paulista "Júlio de Mesquita Filho", Câmpus de Bauru.

Financiadora: Capes

Orientador: Prof. Dr. João Paulo Papa

Bauru, SP, Brasil
2023

S586s

Silva, Vinícius Camargo da
Sumarização Extrativa de Texto Utilizando Modelos Aditivos Generalizados com Interações para Seleção de Sentenças / Vinícius Camargo da Silva. -- Bauru, 2023
63 f. : il., tabs.

Dissertação (mestrado) - Universidade Estadual Paulista (Unesp), Faculdade de Ciências, Bauru
Orientador: João Paulo Papa

1. Ciência da computação. 2. Aprendizado do computador. 3. Processamento de linguagem natural (Computação). I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Biblioteca da Faculdade de Ciências, Bauru. Dados fornecidos pelo autor(a).

Essa ficha não pode ser modificada.

UNIVERSIDADE ESTADUAL PAULISTA

Câmpus de Bauru

ATA DA DEFESA PÚBLICA DA DISSERTAÇÃO DE MESTRADO DE VINICIUS CAMARGO DA SILVA, DISCENTE DO PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO, DA FACULDADE DE CIÊNCIAS - CÂMPUS DE BAURU.

Aos 10 dias do mês de março do ano de 2023, às 14:00 horas, por meio de Videoconferência, realizou-se a defesa de DISSERTAÇÃO DE MESTRADO de VINICIUS CAMARGO DA SILVA, intitulada **"Sumarização Extrativa de Texto Utilizando Modelos Aditivos Generalizados com Interações para Seleção de Sentenças"**. A Comissão Examinadora foi constituída pelos seguintes membros: Prof. Dr. JOAO PAULO PAPA (Orientador(a) - Participação Virtual) do(a) Departamento de Computação/Faculdade de Ciências - UNESP - Bauru, Prof. Dr. APARECIDO NILCEU MARANA (Participação Virtual) do Departamento de Computação /Faculdade de Ciências - UNESP - Bauru, Prof. Dr. TIAGO AGOSTINHO DE ALMEIDA (Participação Virtual) do Departamento de Computação/Universidade Federal de São Carlos (UFSCAR) - Campus Sorocaba. Após a exposição pelo mestrando e arguição pelos membros da Comissão Examinadora que participaram do ato, de forma presencial e/ou virtual, o discente recebeu o conceito final APROVADO. Nada mais havendo, foi lavrada a presente ata, que após lida e aprovada, foi assinada pelo Presidente da Comissão Examinadora.

João Paulo Papa

Prof. Dr. JOAO PAULO PAPA

Para meus pais, Batista e Silvia, por tudo.

Agradecimentos

Agradeço aos meus pais, que me ensinaram, do seu jeitinho, a sonhar, e também sonharam comigo.

Agradeço aos meus irmãos, Bruna e Mateus, que me acolheram e me entenderam tantas vezes.

Agradeço ao meu orientador João Paulo, pelo aprendizado, paciência e inspiração.

Agradeço à UNESP, pelas inúmeras oportunidades acadêmicas, e à CAPES, pelo apoio financeiro durante o desenvolvimento deste projeto.

Por último e em especial, agradeço a Deus, que cuida, com tanto carinho, de um ser tão pequeno, resmungão e limitado como eu.

"No dia da prosperidade, goza do bem; mas, no dia da adversidade, considera em que Deus fez tanto este como aquele, para que o homem nada descubra do que há de vir depois dele."

(Bíblia Sagrada, Eclesiastes 7, 14)

Resumo

A explicabilidade de modelos inteligentes se tornou um importante tópico de pesquisa recentemente. Em função da evolução de diversos algoritmos estatísticos e de Aprendizado de Máquina, hoje, modelos do gênero são capazes de executar tarefas altamente complexas, entretanto, diversos exemplares carecem de transparência sobre seu processo de decisão, culminando em inferências muitas vezes acuradas, segundo métricas e taxas de acerto, porém pouco explicáveis ao usuário em questão. Assim, o termo Inteligência Artificial Explicável ganhou notoriedade nos últimos anos, almejando metodologias capazes de aliar inteligência computacional à explicabilidade na execução de tarefas. A Sumarização Automática de Texto tem se tornado relevante com o crescimento de dados no formato textual, no entanto, com a popularização de grandes bases de dados públicas, abordagens recentes de Aprendizado de Máquina têm se concentrado em modelos e arquiteturas densos que, apesar de produzirem resultados notáveis, geralmente culminam em modelos difíceis de interpretar. Em contrapartida, seria interessante contar com sistemas que promovessem, em paralelo aos resumos gerados, capacidade de oferecer interpretações acerca de seu comportamento ou decisões de maneira transparente, entretanto, essa prática ainda está distante da realidade, uma vez que a interpretabilidade de modelos de sumarização de texto ainda é um assunto desafiador e pouco estudado. Modelos Aditivos Generalizados com Interações (do inglês, *Generalized Additive Models with Interactions* ou GAMI) são conhecidos por aliar poder preditivo a interpretabilidade em tarefas supervisionadas, assim, este trabalho investiga dois desses modelos, a saber, EBM e GAMI-Net, em uma abordagem à tarefa de Sumarização Extrativa, visando explorar sua aplicabilidade ao desafio de sumarização de texto, dado o interesse latente de metodologias interpretáveis. A abordagem proposta, baseada em treinar exemplares de GAMI na forma de um problema de classificação binária, mostrou-se uma alternativa simples, mas atraente a certos algoritmos caixa-preta, cuja avaliação foi realizada utilizando as bases de dados CNN/Dailymail e PubMed.

Palavras-chave: Processamento de linguagem natural; Sumarização automática de texto; Aprendizado de máquina interpretável.

Abstract

The explainability of intelligent models has recently become an important research topic. Due to the evolution of several statistical algorithms and Machine Learning, today, models of this kind are capable of performing highly complex tasks, however, several examples lack transparency about their decision process, culminating in inferences that are often accurate, according to metrics and accuracy rates, but barely explainable to the user. Thus, the term Explainable Artificial Intelligence has gained notoriety in recent years, aiming for methodologies capable of combining computational intelligence with explainability in the execution of tasks. Automatic Text Summarization has become relevant with the growth of data in textual format, however, with the popularization of large public datasets, recent Machine Learning approaches have focused on dense models and architectures that, despite producing notable results often culminate in models that are difficult to interpret. On the other hand, it would be interesting to have systems that promote, in parallel with the summaries generated, the ability to offer interpretations about their behavior or decisions in a transparent way, however, this practice is still far from reality, since the interpretability of text summarization models is still a challenging and understudied subject. Generalized Additive Models with Interactions (GAMI) are known for combining predictive power with interpretability in supervised tasks, as such, this work investigates two of these models, namely, EBM and GAMI-Net, in an approach to the Extractive Summarization task, aiming to explore their applicability to the challenge of text summarization, given the latent interest in interpretable methodologies. The proposed approach, based on training GAMI instances in the form of a binary classification problem, proved to be a simple but attractive alternative to certain black-box algorithms, whose evaluation was performed using the CNN/Dailymail and PubMed datasets.

Keywords: Natural language processing; Automatic text summarization; Interpretable machine learning.

Lista de ilustrações

Lista de quadros

Lista de tabelas

Lista de abreviaturas e siglas

EBM	*Explainable Boosting Machine*
GAM	*Generalized Additive Model*
GAMI	*Generalized Additive Model with Interaction*
IA	Inteligência Artificial
LR	*Logistic Regression*
LRP	*Layer-wise Relevance Propagation*
LSA	*Latent Semantic Analysis*
LSTM	*Long Short-Term Memory*
NBC	*Naive Bayes Classifier*
PLN	*Processamento de Linguagem Natural*
PSVM	*Probabilistic Support Vector Machine*
RF	*Random Forest*
RNN	*Recurrent Neural Network*
SA	Sumarização Abstrativa
SAT	Sumarização Automática de Texto
SE	Sumarização Extrativa
SVD	*Singular Value Decomposition*
TF-IDF	*Term Frequency -Inverse Document Frequency*
TF-ISF	*Term Frequency -Inverse Sentence Frequency*
XAI	*Explainable Artificial Intelligence*

Sumário

1 Introdução

Vivemos em um contexto com volume de dados crescente onde, não por acaso, tópicos como processamentos de dados ganharam espaço. Nesse contexto, sistemas que automatizem ou facilitem processos cresceram em demanda e têm avançado em diferentes cenários. Existem atividades onde a interação humana é importante e não pode ser dispensada, mas que a presença de ferramentas específicas possibilitaria melhoraria de eficiência, servindo, por exemplo, no auxílio à tomada de decisões ou na automatização de tarefas. Com a interação entre homem e máquina ganha-se robustez e confiabilidade nos resultados finais, um complementando o outro em suas deficiências.

No que diz respeito aos dados, a Internet possibilitou um compartilhamento praticamente incessante de informações. A todo instante, publicações são veiculadas em portais de notícias, redes sociais e postagens on-line. Artigos científicos de diversas áreas, por exemplo, são indexados diariamente em bases da Web, o que denota a concentração cada vez maior de conhecimento no meio digital, muitas vezes, em formato de texto.

Uma tarefa de Processamento de Linguagem Natural (PLN) que ganha destaque com a concentração de dados em formato textual e a possibilidade de examiná-los de maneira mais eficiente é a Sumarização Automática de Texto (SAT), que aborda métodos capazes de compilar documentos em porções menores de maneira automática, gerando resumos inteligíveis.

O desafio, entretanto, é que, embora resumir informação possa ser uma tarefa cotidiana dos seres humanos, trata-se de um problema que contém em si paradigmas de linguagem complexos de se imitar computacionalmente. Assim, diferentes técnicas têm sido estudadas ao longo dos anos em busca de soluções mais capazes. Dentre as abordagens promissoras recentes está a utilização de técnicas de Aprendizado de Máquina, que ganharam bastante atenção nos últimos anos graças aos resultados obtidos.

Apesar do sucesso eminente, no entanto, um ponto importante é que a medida que os algoritmos de Aprendizado de Máquina se sofisticaram ao longo de sua evolução para contornar a complexidade dos problemas, a compreensão prática de suas operações internas se tornou mais complicada e a desconfiança envolvida no seu uso aumentou (ZHU et al., 2018).

Pela forma como diversos desses algoritmos trabalham, pode ser difícil rastrear o "raciocínio" envolvido no seu funcionamento mesmo que sob o olhar de especialistas da área, o que faz com que, ainda que eficazes segundo métricas como acurácia, tais algoritmos possam cair em descrédito por erros ingênuos ou acertos fortuitos.

Mais do que analisar taxas de acerto, hoje, entende-se que é importante investigar mais profundamente o comportamento desses modelos. Algoritmos que simplesmente "funcionam",

mas que o usuário envolvido não sabe "como" ou "porque" funcionam, também não manifestam confiabilidade e clareza sobre as decisões sugeridas (SAMEK; MÜLLER, 2019).

A realidade é que a Inteligência Artificial ainda causa ceticismo na sociedade (DOŠILOVIĆ; BRČIĆ; HLUPIĆ, 2018) e que muitos dos modelos aparentemente acurados, na prática, podem estar cometendo erros crassos – como a falha de tradução automática que levou um homem a ser preso (HERN, 2017) – oriundos de limitações que podem passar despercebidas pela falta de transparência.

Hoje, já estão crescendo as demandas sociais por métodos algorítmicos de fato interpretáveis (GOODMAN; FLAXMAN, 2017), o que se conecta à explicabilidade e à transparência no contexto dos modelos de Aprendizado de Máquina. Os esforços aumentaram, culminando na origem da chamada XAI ou Inteligência Artificial Explicável, que compreende o estudo e desenvolvimento de metodologias apropriadas dentro dessa problemática.

Da mesma forma, modelos de SAT, especialmente os baseados em técnicas de Aprendizado de Máquina, herdam essas discussões, já que explicar como resumos são elaborados passa a ser considerado uma necessidade (SARKHEL et al., 2020). No entanto, graças as particularidades do problema, são necessários estudos que levem em conta suas próprias propriedades e interesses dentro desse contexto.

No contexto da SAT, a modelagem interpretável diz respeito a dar transparência ao processo de sumarização do modelo, o que pode contribuir para uma melhor percepção das limitações e capacidades do modelo, ajudar na investigação de por que o modelo comete erros ou até mesmo auxiliar na obtenção de *insights* sobre o problema de sumarização em si. Tais informações podem ser úteis, por exemplo, para evoluir abordagens e esclarecer o que o modelo realmente satisfaz em contraste com as expectativas do usuário.

Trabalhos existentes nessa temática ainda são escassos, desse modo, pensando na importância e no impacto da criação e utilização de metodologias de SAT intepretáveis, surge a motivação do presente trabalho. Modelos Aditivos Generalizados com Interações (do inglês, *Generalized Additive Models with Interactions* ou GAMI) ganharam destaque recentemente como uma classe de modelos que utiliza formulação aditiva e funções não lineares para equilibrar desempenho preditivo a interpretabilidade em problemas de aprendizado. Assim sendo, este trabalho visa investigar o problema de SAT através da aplicação de exemplares desses modelos à tarefa de Sumarização Extrativa.

Modelos EBM e GAMI-Net são dois tipos de GAMI, construídos, respectivamente, utilizando árvores de decisão e redes neurais, sobre uma proposta de equilibrar inteligibilidade e poder preditivo em problemas supervisionados, combinando efeitos principais e interações de pares de forma aditiva, fazendo uso de abordagens modernas de Aprendizado de Máquina.

A ideia por trás dos modelos é aproveitar a formulação aditiva para facilitar a inspeção de seu comportamento e contribuições considerando características explicativas e saídas,

fomentando interpretabilidade. Para algumas tarefas, esses modelos alcançaram desempenho preditivo que compete com o de conhecidas técnicas supervisionadas de Aprendizado de Máquina, ainda que estas outras se utilizassem de formulações mais complexas, que culminam por torná-las pouco transparentes, humanamente falando.

Até onde o autor tem conhecimento, EBMs, GAMI-Nets ou mesmo Modelos Aditivos Generalizados nunca foram antes explorados no contexto de SAT, assim, este trabalho visa investigar a aplicabilidade desses dois tipos de modelo à tarefa de Sumarização Extrativa, na forma de um problema de classificação binária, para posteriormente inferir a relevância das sentenças nos documentos de interesse. Uma vez que esses modelos são considerados interpretáveis, obter desempenho semelhante ou superior a outras técnicas nessa tarefa, e ainda contar com a sua capacidade de transparecer o comportamento aprendido durante o treinamento, poderia simbolizar um passo importante em direção a metodologias interpretáveis para o problema de SAT.

1.1 Hipótese

Exemplares de GAMI como EBM e GAMI-Net podem ser aplicados com êxito à tarefa de Sumarização Extrativa e competir com técnicas menos transparentes de SAT.

1.2 Objetivos

1.2.1 Objetivo Geral

Desenvolver uma abordagem interpretável baseada em GAMI para a tarefa de Sumarização Extrativa.

1.2.2 Objetivos Específicos

a) Verificar a eficácia de GAMI para o problema de Sumarização Extrativa em comparação a abordagens variadas, baseadas em técnicas de Aprendizado de Máquina;

b) Entre EBM e GAMI-Net, verificar qual é mais eficaz na solução do problema exposto.

1.3 Estrutura da Dissertação

O restante da dissertação estão organizados da seguinte maneira:

- O Capítulo 2 apresenta um panorama teórico acerca de três assuntos discutidos na dissertação: Sumarização Extrativa, Inteligência Artificial Explicável e Modelos Aditivos Generalizados;

- O Capítulo 3 apresenta a metodologia utilizada para desenvolver e testar a abordagem proposta, bem como bases de dados e detalhes de experimentação;
- O Capítulo 4 apresenta os resultados obtidos;
- O Capítulo 5 apresenta as conclusões da dissertação.

2 Referencial Teórico

Este capítulo tem como objetivo apresentar o referencial teórico em três seções principais, concernindo o referencial teórico do projeto. Na Seção 2.1, será apresentado o problema de Sumarização Automática e Sumarização Extrativa, na Seção 2.2, o assunto de Inteligência Artificial Explicável e, na Seção 2.3, os Modelos Aditivos Generalizados.

2.1 Sumarização Automática e a Sumarização Extrativa

Nos dias de hoje, os sistemas de computador assumiram um papel importante no acesso e na construção de informação de fácil acesso. Algoritmos que contribuam para este processo, seja ele em pequena ou larga escala, são desejáveis e têm recebido cada vez mais atenção nos últimos anos.

As técnicas de SAT ganham força com essa necessidade, já que sistemas que permitam processar e condensar informação útil estão se tornando cada vez mais relevantes se levarmos em conta a quantidade de dados que são gerados diariamente, especialmente em linguagem natural (EL-KASSAS et al., 2020). Nesse sentido, gerar resumos automaticamente pode ser uma forma de facilitar processos. De maneira geral, um resumo bem elaborado tem a capacidade de denotar um conteúdo de maneira mais sucinta e acelerar o entendimento contido em uma ou mais fontes de texto.

Elaborar resumos manualmente se trata de uma tarefa custosa em tempo e esforço (EL-KASSAS et al., 2020), geralmente requerendo familiaridade com o assunto, o que pode culminar em mão de obra qualificada, que poderia se concentrar em outras tarefas, a despender tempo vasculhando documentos ou facilitando informação (LUHN, 1958).

Para Moratanch e Chitrakala (2017) o objetivo da sumarização é condensar um texto original em uma versão que preserve seu sentido total, enquanto Maybury (1995) alega que o bom resumo é o que destila as informações mais importantes da fonte levando em conta tanto os usuários como as tarefas de interesse. Em suma, a ideia é balancear a compreensão e os detalhes presentes no texto de entrada com o comprimento desejado, que é mais curto que o original por definição.

Do ponto de vista algorítmico, um programa de SAT recebe texto como entrada e gera uma versão resumida como saída (TAS; KIYANI, 2007). Na prática, no entanto, o processo não é tão simples, dado que a linguagem natural é bastante complexa, além de ser uma forma de dado pouco estruturada da perspectiva de máquina. Para isso, são necessários processamentos e modelagens adequados de texto, objeto de estudo da área de Processamento de Linguagem Natural (PLN).

O primeiro trabalho em sumarização automática publicado foi desenvolvido por Luhn (1958). Sua ideia é bastante simples, mas serviu como base para diversas outras técnicas da área. Segundo sua abordagem, quando analisamos um documento, algumas palavras são mais descritivas que outras e isto estaria atrelado à frequência com que elas ocorrem. Desse modo, as sentenças mais relevantes no texto seriam as que possuem um maior número dessas palavras, bastando, então, extrair-se as sentenças mais descritivas do texto para a formulação do resumo.

Abordagens como a de Luhn ficaram posteriormente conhecidas como técnicas de Sumarização Extrativa (SE), que caracterizam os modelos que selecionam partes do texto na íntegra, organizando de maneira adequada, para a confecção do resumo. Diferentemente há, ainda, a chamada Sumarização Abstrativa (SA) onde o resumo pode conter reuso de partes do texto original, no entanto, termos e sentenças próprios podem aparecer (NENKOVA; MCKEOWN, 2011).

A SE é o nicho da SAT que estuda a geração de resumos a partir de segmentos (usualmente sentenças) presentes no texto original. A ideia consiste em escolher partes relevantes do texto e rearranjá-las na forma de um resumo. Em geral, o processo envolve três pontos vitais: segmentar o texto original em representações intermediárias, quantificar, de alguma forma, a importância desses segmentos e selecionar, dentre eles, os mais apropriados para o resumo segundo algum critério (NENKOVA; MCKEOWN, 2012). Usualmente, as técnicas variam pela forma como executam essas etapas.

De outro modo, a SA é caracterizada pela criação de novas sentenças, seja reformulando frases ou utilizando novas palavras (GUPTA; GUPTA, 2019). Na teoria, o que a SA propõe é mais próximo do processo utilizado por humanos na produção de resumos, que normalmente usam tais estratégias para dar naturalidade ao texto. No entanto, a complexidade se intensifica ao abordar o problema dessa forma, já que a geração de linguagem ainda é um desafio pouco trivial.

Desse modo, a SE ainda é atraente, pois, em geral, pode ser feita através de metodologias menos complexas, menos custosas computacionalmente e que sofrem menos com problemas gramaticais ou semânticos (NALLAPATI; ZHAI; ZHOU, 2017), já que os segmentos são extraídos na íntegra. O papel do sumarizador extrativo é mais claro: ressaltar e compilar os segmentos importantes do texto de entrada.

Existem, também, outras formas menos expressivas de qualificar problemas de sumarização. Quanto ao conteúdo, alguns autores categorizam os resumos como indicativos ou informativos (TAS; KIYANI, 2007; EL-KASSAS et al., 2020), dependendo do objetivo final. Enquanto resumos indicativos são aqueles cujo objetivo é informar sobre o escopo de um texto, auxiliando na escolha de lê-lo no todo ou não (EL-KASSAS et al., 2020), resumos informativos devem compreender as informações importantes contidas nos mesmos.

Outra forma de discriminação é entre resumos monodocumento e resumos multido-

cumento (TAS; KIYANI, 2007; EL-KASSAS et al., 2020). Na sumarização monodocumento apenas um corpo de texto é utilizado como base para o resumo. Já na sumarização multidocumento, um conjunto desses textos é usado no processo e o objetivo principal é que não haja informação repetitiva (JOSHI; WANG; MCCLEAN, 2018), o que pode ser uma tarefa complexa, haja vista que espera-se que o resumo mantenha coerência e coesão (TAS; KIYANI, 2007).

A seguir, a Seção 2.1.1 discorre sobre algumas soluções ao problema de SE, enquanto a Seção 2.1.2 aborda treinamento e validação de modelos de Aprendizado de Máquina nesse contexto.

2.1.1 Sumarização Extrativa

Com relação a estudos desenvolvidos para SE, inúmeras estratégias já foram propostas para o problema. Dentre as abordagens mais clássicas, Neto et al. (2000) apresentam um sistema de sumarização de informações baseado em TF-ISF (*Term Frequency - Inverse Sentence Frequency*), uma variante do popular TF-IDF (*Term Frequency - Inverse Document Frequency*). Para o processo de sumarização de texto, o TF-ISF adapta a noção original do algoritmo, representando um documento como um conjunto de sentenças, sobre as quais a frequência de termos (TF) e a frequência inversa de aparições (ISF) incidem.

A estratégia de Neto et al. é baseada na obtenção de uma pontuação para cada sentença, obtida pelo pela média dos pesos TF-ISF dos termos relativos à sentença em questão. Todas as sentenças cuja pontuação obtida for maior do que um limite estabelecido estariam presentes no resumo, na ordem de aparição do documento original. Intuitivamente, cada pontuação simboliza a relevância de uma dada sentença do texto base, logo pontuações maiores significam maior prioridade de aparição no resumo a ser obtido.

Uma outra possibilidade seria utilizar o TF-IDF (ou o TF-ISF) para a vetorização das sentenças (JOACHIMS, 1996) e então usar esses vetores como representações intermediárias, que possam servir como entrada para outros perfis de algoritmos. Um exemplo de trabalho que faz uso desse tipo de representação é o de Ozsoy, Alpaslan e Cicekli (2011), que propõe um sistema genérico de sumarização de texto utilizando Análise de Semântica Latente (do inglês, *Latent Semantic Analysis* ou LSA) (LANDAUER; FOLTZ; LAHAM, 1998), método algébrico baseado em Decomposição em Valores Singulares (do inglês, *Singular Value Decomposition* ou SVD), que apresenta algumas possibilidades quanto à utilização dessa técnica no contexto da SE. A LSA objetiva a extração de relações entre sentenças, palavras e conceitos com base em um texto de entrada:

> "A LSA se baseia na ideia que o agregado de todos os contextos de palavras nos quais uma dada palavra aparece ou não aparece fornece um conjunto de restrições mútuas que determina em grande parte a similaridade de significado de palavras e conjuntos de palavras entre si." (LANDAUER; FOLTZ; LAHAM, 1998, tradução nossa).

De acordo com Ozsoy, Alpaslan e Cicekli, os algoritmos de sumarização com base na LSA usualmente contêm três principais passos: criação da matriz de entrada, SVD e seleção de sentenças, cujas escolhas de metodologia, especificamente do primeiro e do último, interferem diretamente no resultado final.

A matriz de entrada teria o papel de representar o documento de entrada, com as colunas indicando as sentenças e as linhas indicando as palavras/frases, de forma que cada célula quantifique a importância de cada palavra em cada sentença. Os autores também discorrem sobre as possibilidades de diferentes técnicas nesse sentido, como o TF-IDF e a frequência pura de ocorrência das palavras, entre outras.

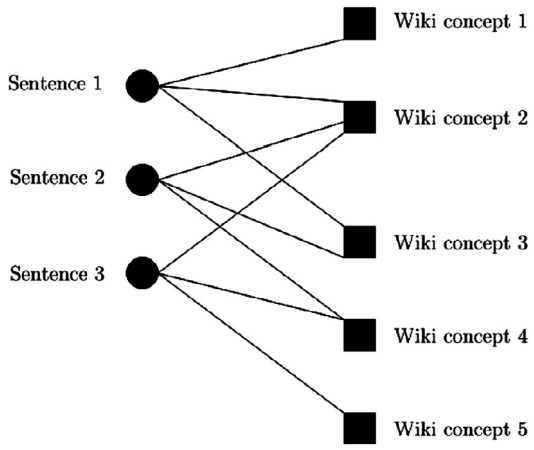
A seleção de sentenças seria então executada a partir da saída obtida da aplicação de decomposição sobre a matriz de entrada. As estratégias de seleção estão fundamentadas na noção de que a LSA permite que conceitos (ou tópicos) sejam encontrados estatisticamente no documento de origem. Uma estratégia simplista, por exemplo, seria, para cada conceito encontrado, escolher a sentença mais pertinente ao conceito e incutir no resumo.

Um outro trabalho que faz uso da noção de sumarização baseada em conceitos é o de Ramanathan et al. (2009). Sua principal contribuição, no entanto, é a utilização de conhecimento externo nesse processo. O problema é representado como um grafo bipartido entre sentenças e conceitos representados através de artigos de Wikipedia, encarregados de trazer conhecimento comum para a escolha de sentenças (Figura 1). A partir das sentenças do documento original, os autores aplicam o motor de busca Lucene na recuperação de artigos Wiki relevantes, que são ranqueados levando em conta o número de sentenças relacionadas, assumindo-se que os mais importantes estariam associados a um maior número de sentenças. Dessa forma, as sentenças relacionadas aos conceitos mais importantes, são colocadas no resumo. Em um segundo trabalho (SANKARASUBRAMANIAM; RAMANATHAN; GHOSH, 2014), os autores refinam o processo, dessa vez, adotando o plano de ranquear também a importância das sentenças, além da estratégia original de ranquear a importância dos conceitos, na argumentação que uma importância determina a outra de maneira "mutuamente reforçada".

Se por um lado, sentenças relacionadas a diversos conceitos poderiam conter muita informação do texto compilada em si, o que seria potencialmente importante para o resumo, também podem satisfazer essa condição por serem extremamente vagas ou genéricas demais. O equilíbrio entre o número e a relevância de conceitos ou sentenças não pode ser trivializado e comumente é um fator sensível de se considerar no emprego desse tipo de técnica. A dificuldade em relativizar essas questões é uma das motivações por trás da ideia de modelar o problema estatisticamente.

Com o crescimento da área de Aprendizado de Máquina na resolução de problemas variados, surgiu também o interesse em aplicá-la ao problema de SAT. Um fator importante nesse contexto é o volume dos dados necessário pelas técnicas supervisionadas (WONG; WU; LI, 2008), uma vez que esse tipo de técnica se baseia na modelagem estatística entre pares de

Figura 1 – Grafo bipartido de sentenças e conceitos



Fonte: Sankarasubramaniam, Ramanathan e Ghosh (2014)

dados de entrada e rótulos, chamados de base de dados ou *datasets*.

As técnicas extrativas mencionadas até então não são técnicas de Aprendizado de Máquina, sendo possível afirmar que não requerem treinamento supervisionado. Embora isso represente uma vantagem com relação a simplicidade e aplicabilidade de um modelo, abordagens baseadas em Aprendizado de Máquina supervisionado (e mais recentemente, Aprendizado Profundo (LECUN; BENGIO; HINTON, 2015)) têm apresentado possibilidades promissoras nesse contexto.

Um exemplo que faz uso de Aprendizado de Máquina clássico é o trabalho de Wong, Wu e Li (2008), que investiga a aplicação de Máquinas de Vetores de Suporte Probabilísticas (do inglês *Probabilistic Support Vector Machines* ou PSVMs) (WU; LIN; WENG, 2004) e Classificadores Naive Bayes (do inglês, *Naive Bayes Classifiers*, ou NBCs) na extração de resumos. Os autores analisam a combinação de diferentes procedimentos na fase de extração de características, apresentando uma comparação de performance. Mais especificamente, eles testam quatro tipos de características de sentenças, referidas como caraterísticas de superfície, de conteúdo, de eventos e de relevância. De maneira geral, o trabalho evidencia como a etapa de processamento de linguagem é preponderante quanto a utilização de técnicas mais clássicas de aprendizado máquina na extração de resumos.

O trabalho de Wong, Wu e Li (2008) ainda discute a quantidade grande de dados anotados necessária para o treinamento de modelo supervisionados. O processo de anotação manual pode ser bastante laborioso na prática, o que incentiva estudos focados em contornar a questão. Pensando nisso, o trabalho também introduz o uso de co-treinamento (BLUM; MITCHELL, 1998) nesse contexto, mostrando que o uso da técnica semi-supervisionada pôde reduzir a quantia de dados de treino pela metade sem que houvessem grandes perdas na

capacidade de sumarização.

Mais recentemente, assim como em outras áreas de estudo, redes neurais profundas alavancaram muitas tarefas dentro de PLN, oferecendo resultados competitivos ou superando metodologias existentes (LECUN; BENGIO; HINTON, 2015). Modelos como as Redes Neurais Recorrentes (do inglês, *Recurrent Neural Networks* ou RNNs) e os *Transformers* (VASWANI et al., 2017) foram dois tipos de modelos bastante explorados, visto que conseguem processar informações sequenciais pouco estruturadas, tal como as palavras de um texto, capturando características como morfologia e sintaxe de sentenças automaticamente. Em contrapartida, usualmente são necessários ainda mais dados do que técnicas supervisionadas tradicionais para que os modelos entreguem resultados superiores.

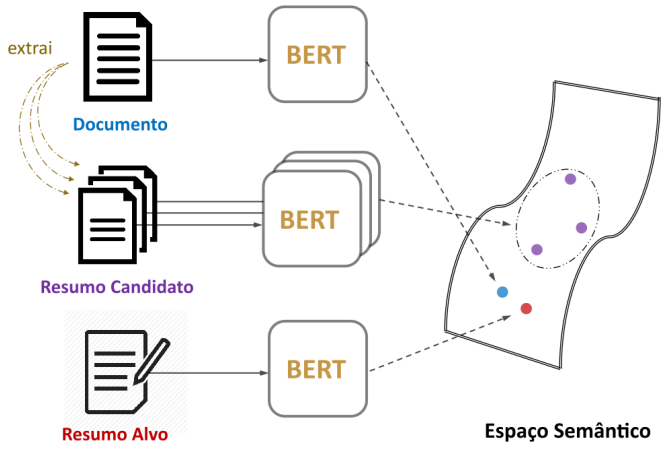
Nallapati, Zhai e Zhou (2017) propõem uma metodologia baseadas em LSTMs (do inglês, *Long Short-Term Memory* (HOCHREITER; SCHMIDHUBER, 1997), um tipo especial de RNN), que introduz uma metodologia de treinamento extrativa baseada em resumos abstrativos. Narayan, Cohen e Lapata (2018), por sua vez, utilizam uma aquitetura hierárquica de CNNs (*Convolutional Neural Networks*) e LSTMs em combinação com aprendizado por reforço no ranqueamento de sentenças. Eles argumentam que a abordagem padrão por entropia cruzada de máxima probabilidade (como é o caso no trabalho de Nallapati, Zhai e Zhou) seria deficiente para o ranqueamento de sentenças na sumarização de texto. A partir dessa motivação, propõem um algoritmo baseado em combinar essa métrica com a avaliação ROUGE (*Recall-Oriented Understudy for Gisting Evaluation*) (LIN, 2004) em uma política de aprendizado por reforço, superando o desempenho de modelos similares treinados com aprendizado supervisionado.

No contexto das redes *Transformers*, o trabalho de Liu (2019), por exemplo, adapta a arquitetura original do modelo de representação de linguagem BERT (DEVLIN et al., 2019). O BertSum, como é chamado o modelo, aproveita a ideia do BERT modificando a forma como as sentenças são inseridas no modelo de linguagem. No BertSum, um documento segmentado em mútiplas sentenças serve como uma única entrada, cujas sentenças são separadas por um *token* especial, além de receberem um de dois *embeddings* que são intercalados ao longo das sentenças para que o modelo possa distingui-las dentro do documento. Essa mudança permite que o modelo possa ser adequado ao problema de SE. A base do BertSum, então, fica encarregada de gerar representações vetoriais das sentenças, que são oferecidas a algum classificador neural que recebe ajuste fino junto com o restante do modelo, assim como nas aplicações do BERT.

Também nesse contexto, Zhong et al. (2020) introduz o modelo MatchSum, outra abordagem baseada na arquitetura BERT. O MatchSum, no entanto, interpreta o problema de SE como um problema de *matching* semântico. Em outras palavras, o objetivo é obter o resumo que, no todo, mais se aproxima semanticamente do documento original, em vez de analisar as sentenças em separado. Para tanto, eles investigam a utilização de uma arquitetura siamesa (BROMLEY et al., 1993) baseada em BERT, cuja intenção é medir quão semanticamente

correspondentes são documentos e candidatos a resumo (Figura 2). Os candidatos de resumo são obtidos a partir de todas as combinações de sentenças identificadas salientes aplicando-se uma adaptação do BertSum.

Figura 2 – Correspondência da representação de resumos e documento no espaço latente.



Fonte: Traduzida de Zhong et al. (2020).

2.1.2 Treinamento e teste de modelos

No contexto de Aprendizado de Máquina, quando problemas são analisados com relação a, principalmente, os dados prévios que se tem, uma distinção comumente feita ao categorizar sua modelagem é dada entre modelos supervisionados e não supervisionados. Dentro da SA, técnicas não supervisionadas aprenderiam e elaborariam resumos diretamente com base em documentos de interesse, enquanto técnicas supervisionadas demandariam treinamento sobre rótulos.

Confeccionar essas bases, no entanto, é um processo demorado; além do tempo de construí-las, existe também um ponto controverso para sumarização, que é a dificuldade em estabelecer padrões claros para a inclusão de sentenças no resumo. Não seria difícil que pessoas distintas escolhessem também de maneira distinta as sentenças no momento da construção de um resumo (ALLAHYARI et al., 2017).

Ainda assim, técnicas supervisionadas são comuns, pois tendem a direcionar melhor os modelos, mesmo que utilizando rótulos sintéticos (NALLAPATI; ZHAI; ZHOU, 2017; KEDZIE; MCKEOWN; III, 2018; XIAO; CARENINI, 2019; LIU, 2019). A questão é que para o treinamento de modelos supervisionados, em geral, são necessárias bases grandes de dados, fator que pode se intensificar de acordo com as técnicas empregadas. Desse modo, em vez de construir bases próprias do zero, trabalhos recentes vêm utilizando bases já estruturadas e de fácil acesso à

comunidade acadêmica no processo de criação e validação de metodologias supervisionadas, como a CNN-Dailymail (HERMANN et al., 2015).

Uma etapa importante na preparação de um sumarizador é avaliar a qualidade dos resumos que são elaborados, o que pode ser feito tanto através de avaliação humana ou utilizando métodos automáticos que se baseiem nos resumos-candidato, isto é, gerados pelo modelo, e resumos-referência para mensurar qualidade. A principal das métricas automáticas de desempenho, especialmente para sumarizadores extrativos, são as pontuações ROUGE, uma família de métricas que leva em conta a coocorrência de n-gramas presentes no resumo-candidato comparado a um ou mais resumos-referência. Com base em um resumo-alvo S e um resumo-candidato R as pontuações ROUGE$_{recall}$ (Equação 2.1) e ROUGE$_{precision}$ (Equação 2.2) são calculadas como se segue:

$$\text{ROUGE}_{recall-n} = \frac{\sum_{gram_n \in S} count_{match(S,R)}(gram_n)}{\sum_{gram_n \in S} count(gram_n)} \tag{2.1}$$

$$\text{ROUGE}_{precision-n} = \frac{\sum_{gram_n \in S} count_{match(S,R)}(gram_n)}{\sum_{gram_n \in R} count(gram_n)} \tag{2.2}$$

onde $count_{match(S,R)}(gram_n)$ corresponde ao número máximo de coocorrências considerando do n-grama $gram_n$, e $count(gram_n)$ é o número de ocorrências de $gram_n$.

Costumeiramente, é empregada a média harmônica entre as pontuações ROUGE$_{recall-n}$ e ROUGE$_{precision-n}$, referida como ROUGE F$_n$, ou simplesmente ROUGE$_n$, para avaliar o desempenho dos modelos de maneira unificada. Existe ainda a medida ROUGE$_L$ que considera a maior subsequência de n-gramas para realizar o cálculo.

2.2 Inteligência Artificial Explicável

Recentemente, algoritmos de aprendizado de máquina demonstraram capacidades avançadas, muito em função do avanço na área do Aprendizado Profundo (LECUN; BENGIO; HINTON, 2015). Hoje, modelos do gênero são capazes de executar tarefas altamente complexas e até humanamente inviáveis.

As redes neurais profundas, nome utilizado para endereçar redes neurais com múltiplas camadas escondidas (LECUN; BENGIO; HINTON, 2015), rapidamente se tornaram tendência de estudo e aplicação, impulsionando pesquisas e o mercado, além de contribuir para o crescimento de novas carreiras.

Mesmo que muitas dessas tecnologias tenham evoluído ao ponto de entregarem resultados sobre-humanos, ainda existem diversos desafios a serem sobrepostos nessa área. Um particularmente relevante é a falta de transparência e explicabilidade desses modelos (DOŠILOVIĆ; BRČIĆ; HLUPIĆ, 2018; SAMEK; MÜLLER, 2019; ARRIETA et al., 2020), que

diminui a confiabilidade e a clareza por trás das decisões tomadas por esses algoritmos. Com a sofisticação das técnicas de aprendizado de máquina, os modelos têm se tornado cada vez mais complexos, aumentando a desconfiança envolvida (ZHU et al., 2018). Se levarmos em conta ambientes de natureza mais crítica, como as aplicações de algoritmos de IA no auxílio a decisões médicas ou financeiras, a falta de transparência das técnicas pode ser um fator limitante ou até desqualificante (SAMEK; MÜLLER, 2019).

Em resposta, termos como interpretabilidade e explicabilidade de modelos têm sido cada vez mais recorrentes em publicações da área de IA (ARRIETA et al., 2020), o que denota a preocupação crescente da comunidade científica com relação a essas questões. A Inteligência Artificial Explicável (do inglês, *Explainable Artificial Intelligence* ou XAI) é área de estudo que surge para compreender essa problemática. A XAI estuda o universo que abrange desde a criação de modelos transparentes, compreensíveis e interpretáveis a técnicas que visam promover a explicabilidade de modelos caixa-preta – outro termo bastante presente para denotar a ausência de interpretabilidade.

Se, em geral, o foco das técnicas de aprendizado de máquina é resolver problemas através de modelos estatísticos inteligentes, o foco da XAI é concentrado em aliar essas questões ao entendimento das pessoas envolvidas. Arrieta et al. (2020) argumentam que uma IA, para ser dita explicável, precisa ser não só simples de entender, mas especialmente entendível na perspectiva do usuário em questão. Na visão dos autores, tanto as razões quanto a clareza das explicações depende completamente do público, e portanto, o sistema precisa ser desenvolvido sob essa ótica. Ainda sobre isso, Samek e Müller (2019), por sua vez, acreditam que diferentes perfis de usuários podem vir a requerer diferentes tipos explicações, o que eventualmente implicará na utilização de diferentes abordagens.

Um primeiro motivo para a elaboração desses modelos é que explicações fomentam confiança e verificabilidade (SAMEK; MÜLLER, 2019). Receber explicações proporciona aos humanos mais segurança e convicção quanto a decisões que estão sendo tomadas. Além da confiança incutida, fica também mais claro as potencialidades e limitações do algoritmo; se é entendido "como" as decisões são tomadas, também é verificado "onde" e "porque" o modelo funciona (ou deixa de funcionar).

Um episódio curioso a respeito disso é o do classificador capaz de predizer "acuradamente" imagens de cavalos que, na prática, estava decidindo com base na presença de marcas d'água de direitos autorais no canto das imagens – que passaram despercebidas na coleta de dados para treino e teste – e não no que era esperado, os cavalos (LAPUSCHKIN et al., 2016). Um modelo com um viés indesejado ou que se baseia demais em características impraticáveis, ainda que aparente ser acurado, é indesejado e pouco confiável. Nesse sentido, as explicações poderiam cumprir um papel fundamental ao dar credibilidade ou não ao funcionamento do algoritmo.

Explicações também poderiam contribuir para a formação de *insights* a respeito do

problema e seus dados (SAMEK; MÜLLER, 2019), o que poderia ser útil em diferentes cenários, especialmente em contextos acadêmicos ou de pesquisa. Usuários poderiam, a partir da modelagem de uma tarefa ou problema específico, descobrir relações inicialmente desconhecidas dentro das características dos dados. Na biomedicina, por exemplo, entender melhor as relações entre as características de entrada pode contribuir para elaboração de um teste clínico mais simples e menos custoso (LIBBRECHT; NOBLE, 2015), a partir do reconhecimento dos atributos mais relevantes para uma dada tarefa. Muito além da modelagem de tarefas em si, por vezes seria interessante aprender com o problema que está sendo tratado.

Um outro motivo importante é a ética e a responsabilidade atrelada ao uso da IA. Em 2016 o Parlamento Europeu instituiu várias diretrizes quanto ao uso de dados pessoais e o direito de pedir por explicações sobre decisões baseadas em algoritmos de computador (GOODMAN; FLAXMAN, 2017). Com o uso da IA muito presente no cotidiano, adequações legais são importantes e tendem a se tornar frequentes, uma vez que é esperado que os algoritmos não firam princípios éticos e de privacidade de pessoas. Para tanto, técnicas explicáveis são imprescindíveis para a melhor compreensão e entendimento a respeito das saídas desses algoritmos.

Técnicas interpretáveis e explicáveis se tornaram um tópico muito discutido na área de IA e representa o próximo grande passo em direção a modelos confiáveis e responsáveis. A seguir, serão apresentados conceitos gerais a respeito de XAI, além de assuntos pertinentes ao presente trabalho nesse contexto.

2.2.1 Tipos de explicação

Assim como discutido no início do capítulo, o que determina a qualidade das explicações é o entendimento do usuário. Além do nível de detalhamento das explicações, o conteúdo que está sendo apresentado é muito importante em ditar a percepção do que está sendo efetivamente explicado. Tipos diferentes de explicação abordam diferentes aspectos do modelo (SAMEK; MÜLLER, 2019) de acordo com a intenção. Isto posto, para Samek e Müller (2019) são quatro os tipos principais de explicação:

- **Representações aprendidas**: visam explicitar o entendimento contido em representações intermediárias geradas pelo algoritmo (KIM et al., 2018), especialmente quando lidando com redes neurais. Isto contribui para a compreensibilidade de que tipo de informação pode ser encontrada nas representações internas de um modelo, seja para entendê-lo melhor ou para proteger informações. Por exemplo, não entender ou se atentar ao que está armazenado nestas representações poderia implicar em brechas de segurança, caso terceiros o façam (ARRIETA et al., 2020).
- **Predições individuais**: pretendem gerar explicações a respeito de predições individuais, especialmente com relação aos dados de entrada. Por exemplo, ao gerar um mapa de

calor que denote áreas salientes (SIMONYAN; VEDALDI; ZISSERMAN, 2014) para uma dada predição do algoritmo. Pode ser útil para a prospecção de *insights* mais sutis entre as características ou do porquê uma classe foi atribuída a uma amostra específica.

- **Comportamento do modelo**: a ideia geral consiste em gerar um entendimento abrangente do comportamento do modelo. Um exemplo seria elencar as características mais relevantes globalmente com base no aprendizado obtido no todo. Além de ajudar a encontrar causalidade entre características de entrada e as predições, torna as decisões do modelo no geral mais transparentes, promovendo confiabilidade.

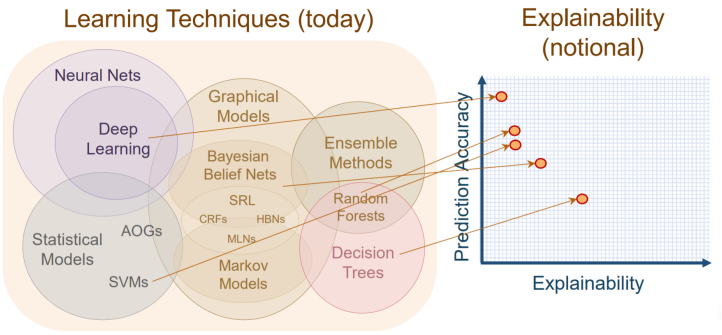
- **Exemplos representativos**: identificam exemplos de treino representativos para a tarefa em questão. Como elencado por Samek e Müller(2019), esse tipo de explicação pode propiciar um melhor entendimento da base de dados além ajudar identificar os tipos de vieses que o modelo está sofrendo a partir dela.

2.2.2 Metodologias explicáveis

Técnicas de XAI pretendem, de modo geral, promover explicabilidade na modelagem de problemas de IA. Nesse contexto, metodologias transparentes e interpretáveis estão mais relacionadas a modelos explicáveis em essência. Por outro lado, existem também as técnicas *post-hoc* que se referem a produção de explicações a partir de algoritmos não naturalmente explicáveis (ou caixa-preta) (ARRIETA et al., 2020).

Como apresentado na Figura 3, por padrão, algoritmos que apresentam melhores resultados em termos de desempenho são menos explicáveis em contrapartida. Contornar essa dificuldade, seja tornando modelos performáticos mais interpretáveis ou elaborando algoritmos mais transparentes, é o grande interesse das metodologias de XAI.

Figura 3 – Desempenho vs. Explicabilidade



Fonte: Gunning (2017)

A seguir, serão apresentados alguns conceitos básicos acerca de modelos transparentes e técnicas de explicabilidade.

2.2.2.1 Transparência e modelos caixa-de-vidro

As técnicas interpretáveis, caixa-de-vidro, ou transparentes são aquelas onde o processo de entender o modelo em questão é direto ou simplificado. A natureza de inúmeras técnicas de aprendizado de máquina clássicas está inclinada a abordagens mais simples de entender e interpretar, tornando-as mais transparentes do que técnicas avançadas como, por exemplo, Redes Neurais Profundas. O problema, entretanto, não está no uso de neurônios em si, mas na densidade dessas redes. Um único *perceptron* (ROSENBLATT, 1958), provavelmente seria mais interpretável do que um sistema baseado em um número grande de regras (ARRIETA et al., 2020), mesmo que as regras, individualmente, sejam mais intuitivas que a ativação neural. A densidade, porém, seja de um número grande de neurônios ou de regras, é um dos pontos que afeta a transparência do modelo.

Em alguns cenários, a interpretabilidade pode ser mais importante do que o desempenho em si, pois, como mencionado, as métricas de desempenho não garantem que o modelo é necessariamente confiável e aplicável no mundo real (LAPUSCHKIN et al., 2019). Nesse contexto, técnicas transparentes ganham força em detrimento de abordagens mais sofisticadas e menos interpretáveis.

As Arvores de Decisão, por exemplo, possuem transparência considerável, razão pela qual são bastante utilizadas na tomada de decisões ainda que seu desempenho normalmente não acompanhe outras metodologias. Mesmo para um não-especialista, a noção por trás do processo de decisão do algoritmo é bastante intuitiva, facilitando a compreensão e o convencimento do algoritmo.

Regressores Lineares/Logísticos também favorececem a sua interpretabilidade graças a relação linear que constroem entre as características de entrada e a saída. Se por um lado, muitos problemas podem ser difíceis de se resolver linearmente, por outro, isto pode simplificar a visualização da relevância e possíveis causalidades das características para o problema.

Os Modelos Aditivos Generalizados (do inglês, *Generalized Additive Models* ou GAMs), que constituem uma classe de modelos que substituem a função linear dos regressores por uma de agregação de funções suaves (HASTIE; TIBSHIRANI, 1987), também podem ser vistos como transparentes, baseado na aditividade do modelo.

Em resumo, as chamadas técnicas transparentes são aquelas que, na sua natureza, podem ser facilmente apresentadas ao humano, seja conceitualmente ou em termos visuais, permitindo que suas partes sejam explicadas sem que ferramentas adicionais muito elaboradas sejam usadas (ARRIETA et al., 2020).

2.2.2.2 Explicabilidade e modelos caixa-preta

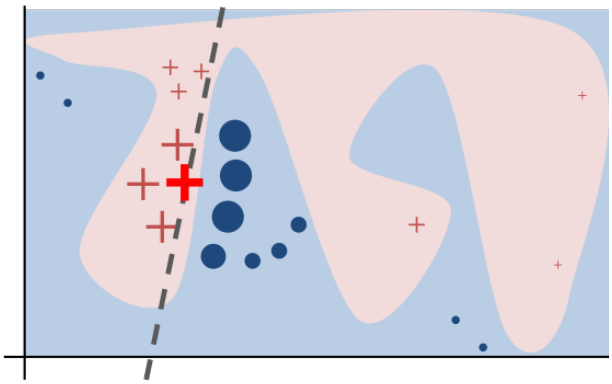
Diferentemente dos modelos transparentes, os modelos caixa-preta ou com baixa interpretabilidade são aqueles onde o entendimento das decisões tomadas é dificultoso. Alguns

exemplares, como arquiteturas neurais profundas, usualmente são procurados por aliviar a etapa de engenharia de características, e entregar, para muitas tarefas, alto desempenho estatístico. No entanto, quanto mais densas e complexas as arquiteturas, menos transparentes e mais difíceis de compreender se tornam as predições.

Nesse caso, técnicas à parte que fomentem explicabilidade de modelos inicialmente densos são chamadas técnicas post-hoc. As técnicas post-hoc visam preencher a lacuna entre modelos caixa-preta e as explicações (ARRIETA et al., 2020). Para tanto, algumas técnicas se utilizam de perturbações do modelo original para a obtenção de explicações. O método de oclusão (ZEILER; FERGUS, 2014), por exemplo, diz respeito à investigação de modelos através da obstrução de diferentes áreas das entradas (no trabalho em questão, imagens), visando identificar quais partes seriam mais sensíveis às perturbações, baseando-se nas mudanças verificadas na intensidade das saídas intermediárias ao longo das camadas da rede. As áreas de sensibilidade podem ser entendidas como pontos-chave no entendimento das predições.

O LIME (RIBEIRO; SINGH; GUESTRIN, 2016), por sua vez, advém da ideia de utilizar perturbações para o treinamento de novos modelos interpretáveis, que expliquem o modelo caixa-preta. O algoritmo usa perturbações sobre uma amostra original para aprender um modelo simplificado que aproxime o comportamento do modelo original nos arredores daquela amostra. Como pode ser visto na Figura 4, a ideia é obter um processo de decisão que represente a modelagem do problema localmente, dessa forma, o modelo simplificado poderia ser utilizado para elaborar explicações para a predição da amostra a ser investigada. A Figura 5 denota o processo, onde a linha tracejada representa a aproximação obtida via LIME que visa explicar o comportamento do modelo original, representado pela classificação em áreas azuis e vermelhas.

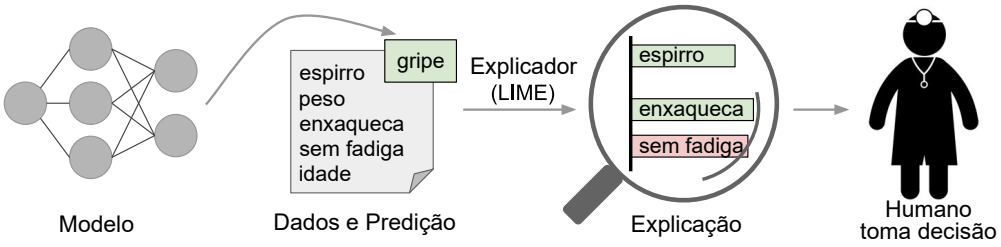
Figura 4 – LIME



Fonte: Ribeiro, Singh e Guestrin (2016)

Existem, também, técnicas baseadas em propagação, como é o caso do *framework* LRP (do inglês, *Layer-wise Relevance Propagation*) (BACH et al., 2015). A ideia consiste em explicar decisões individuais de modelos neurais retro-propagando a predição em direção às entradas usando regras locais de redistribuição (SAMEK; MÜLLER, 2019). O interesse é em

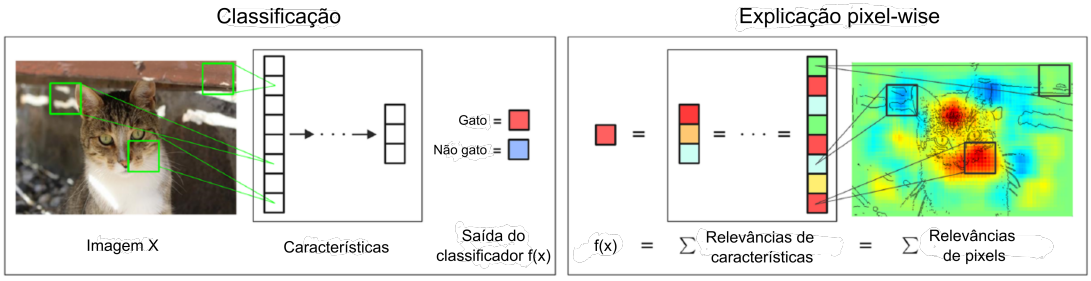
Figura 5 – Explicando predições individuais



Fonte: Traduzida de Ribeiro, Singh e Guestrin (2016)

investigar as ativações dos neurônios de maneira facilitada, o que também pode ser usado para identificar áreas relevantes nas entradas, como na Figura 6, que denota o processo de utilizar uma medida de saliência e mapas de calor para representar as áreas consideradas mais importantes durante a classificação da imagem à esquerda.

Figura 6 – Explicação de pixels



Fonte: Traduzida de Bach et al. (2015)

Lapuschkin et al.(2019) mostram, que meta-explicações, nesse contexto, também poderiam ser úteis para a elaboração de explicações. Mais especificamente, os autores propõem a metodologia SpRAy (acrônimo para *Spectral Relevance Analysis*) para a inspeção de comportamentos de decisão do modelo, através da clusterização de explicações geradas pelo LRP. O principal objetivo é identificar as diferentes estratégias de predição do modelo e seus comportamentos típicos e atípicos. Na prática, isso ajudaria a entender o que o modelo leva em consideração para funcionar, possivelmente ajudando na identificação de comportamentos indesejados.

Em resumo, as técnicas post-hoc são ferramentas complementares que, de alguma forma, tentam contornar a carência de interpretabilidade inerente a modelos originalmente caixa-preta.

2.2.3 Trabalhos correlatos

Diversos trabalhos de destaque, hoje em dia, estão relacionados a modelos neurais profundos, que, apesar de terem elevado o nível, tornaram a resolução de tarefas muito menos intuitiva (DANILEVSKY et al., 2020). Na Sumarização Abstrativa, por exemplo, os modelos evoluíram consideravelmente após o advento do Aprendizado Profundo (WU et al., 2021).

A intepretabilidade na área de sumarização poderia trazer uma nova visão ao problema e à evolução da tarefa, no entanto, trabalhos nesse sentido ainda são escassos, principalmente considerando o latente interesse nos modelos de linguagem grandes (do inglês, *Large Language Models* ou LLM), que são comumente complexos e densos (DEVLIN et al., 2019; SUTSKEVER; VINYALS; LE, 2014), e estão distantes das ideias fomentadas pela XAI.

Um estudo recente – o primeiro que se tem conhecimento, fez um levantamento da aplicação de técnicas de XAI no domínio de PLN (DANILEVSKY et al., 2020). Segundo os autores, as abordagens mais comuns são as voltadas à investigação da importância de características, que tendem a possibilitar intuitividade dentro do processo lógico dos modelos, o que justificaria seu uso.

A utilização de mecanismos de atenção (BAHDANAU; CHO; BENGIO, 2015), por exemplo, foi uma das abordagens mais utilizadas, justamente por apelar à intuição humana e ajudar a indicar onde o modelo neural está "focando" (DANILEVSKY et al., 2020). GHAEINI; FERN; TADEPALLI (2018), por exemplo, fazem um mapeamento da saliência (nesse caso, derivadas de primeira ordem) dos mecanismos de atenção e dos portões de redes LSTMs. No entanto, o debate sobre o uso e quão explicáveis esses mecanismos são na prática, ainda permanece em aberto (JAIN; WALLACE, 2019; SERRANO; SMITH, 2019).

A recente utilização de LRP dentro de PLN (ARRAS et al., 2017) também é um sinal positivo de avanço, visto que a técnica é aplicável a muitos tipos de arquitetura neural. Entretanto, seu uso ainda é preponderante para tarefas de classificação padrão, não tendo sido estudado em arquiteturas mais complexas e bem-sucedidas como *sequence-to-sequence*, até onde se tem conhecimento.

Na área de visualização da informação, o trabalho de Strobelt et al. (2018) propôs uma ferramenta de visualização para depuração de modelos *sequence-to-sequence* em geral. A ferramenta permite investigar as decisões do modelo utilizando busca em feixe sobre o processo de decisão, além de possibilitar relacionar o estado interno de amostras similares. A ideia é permitir que usuário examine as decisões do modelo tanto do ponto de vista geracional, quanto do de "entendimento" da sentença de entrada.

Em um outro trabalho, este com foco em Sumarização Extrativa, Nallapati, Zhai e Zhou (2017) argumentam promover interpretabilidade de maneira automática ao treinamento do modelo. A ideia consiste em calcular a probabilidade de uma dada sentença ser adicionada ao resumo através de uma função elaborada pelos autores que, segundo argumentam, indicam

propriedades como adequação de conteúdo, saliência e novidade da sentença, que poderiam ser utilizadas como informação interpretável ao usuário. Na prática, entretanto, pode ser difícil validar a qualidade dessas interpretações e garantir que elas realmente preservam os significados propostos, dada a profundidade e a complexidade da rede.

Sarkhel et al. (2020), por sua vez, propõem um novo mecanismo de atenção para problemas de Sumarização Abstrativa que, segundo discutem os autores, é mais leve e interpretável do que o utilizado em outras abordagens. O método consiste em construir um resumo protótipo a partir de uma arquitetura neural estabelecida e depois adequá-lo usando um mecanismo de atenção baseado em 3 núcleos. Cada núcleo está associado a uma propriedade (a saber, cobertura de tópicos principais, palavras-chave e redundância de informações), expressa pelas características que são aproveitadas no núcleo, e possibilitando que, durante a inferência, a contribuição de cada sentença protótipo seja medida com relação as propriedades, segundo argumentam os autores.

Também recentemente, Ghodratnama et al. (2020) propuseram uma metodologia interpretável baseada em mapeamento de características para Sumarização Extrativa. A metodologia, que aprende e atribui pesos às características que podem ser posteriormente usados como explicações, apresentou resultados superiores a outra técnicas robustas e menos interpretáveis, como por exemplo o modelo neural de Nallapati, Zhai e Zhou (2017). A estratégia se baseia em um algoritmo que mistura aprendizados supervisionado e não-supervisionado em um procedimento que permite indicar a importância de cada uma das características seja para a decisão de inclusão ou exclusão de uma dada sentença no resumo.

De modo geral, a formulação e o uso de metodologias interpretáveis aplicadas ao PLN ainda está em seus estágios iniciais. A sumarização de texto, especificamente, também foi pouco explorada nesse contexto, sendo raramente mencionada junto a estratégias e metodologias que fomentem diretamente interpretabilidade.

2.3 Modelos Aditivos Generalizados

Como discutido anteriormente, a sofisticação dos modelos de Aprendizado de Máquina afetou os algoritmos e sua explicabilidade, corroborando para a criação de modelos mais acurados ao preço de torná-los, também, menos transparentes (ZHU et al., 2018). Os GAM (HASTIE; TIBSHIRANI, 1987), por sua vez, se destacam como uma espécie de meio termo entre modelos de complexidade total e modelos lineares, mesclando aspectos positivos de ambos os contextos (LOU; CARUANA; GEHRKE, 2012).

Em modelos de alta complexidade, como redes neurais profundas, é comum serem utilizadas funções não-lineares que agregam múltiplos componentes de entrada por vez, sejam eles características extraídas ou dados sem qualquer processamento (como os pixeis de uma imagem). Em geral, isso requer conjuntos maiores de dados para treinamento, mas propicia

ao modelo um potencial maior de aprendizado. Por outro lado, isso também dificulta a interpreatabilidade do modelo, já que analisar a relação construída entre componentes e a saída durante o aprendizado do modelo se torna mais abstrata.

Em contrapartida, a formulação de modelos lineares resulta em simplicidade no que diz respeito a sua interpretabilidade, já que o exercício de analisar os coeficientes lineares presentes na formulação do modelo pode trazer certa clareza sobre sua relevância (módulo) ou até como se relacionam com a saída das predições (sinal).

Desse modo, a estratégia dos GAM pode ser vista como uma tentativa de conciliar o desempenho oferecido por abordagens não-lineares ao ajustar problemas avançados e a interpretabilidade atrelada a elaborar funções de baixa dimensionalidade que são combinadas de maneira aditiva. Os GAM representam a classe de modelos cuja fórmula é descrita pela Equação 2.3:

$$g(y) = \sum f_i(x_i) \tag{2.3}$$

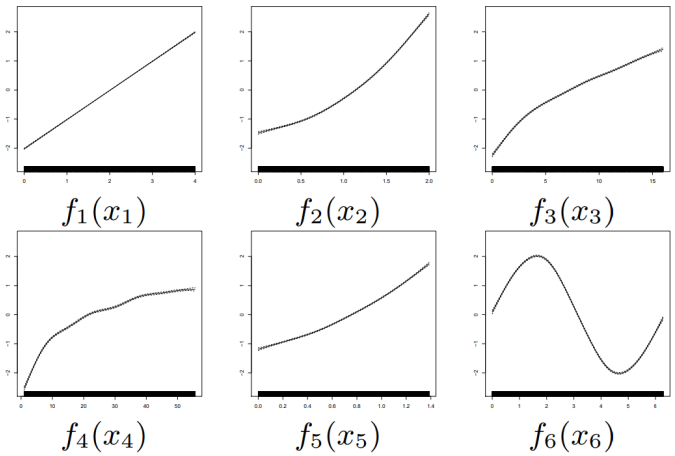
onde a função g é chamada função *link* ou função de ligação e as funções f são chamadas funções $shape$ ou funções de forma. Dessa forma, cada componente ou característica de entrada x_i é ajustada por sua respectiva função de forma f_i.

No contexto de explicabilidade, um modelo de ordem linear pode ser considerado intuitivo, no entanto, isto não significa necessariamente que as interpretações que oferece são fidedignas a natureza do problema, como exemplificado por Lou, Caruana e Gehrke (2012), que advertem que tal tentativa pode se fazer "enganosa". Em outras palavras, poderíamos obter uma *solução* com boa interpretabilidade, mas que quanto mais distante do comportamento de mundo real do problema, como consequência, pode não oferecer a melhor interpretação acerca do *problema*, em si. Assim sendo, a presença de funções de forma não-lineares encontrada nos GAM pode, além de elevar a acurácia final, contribuir para um modelo interpretável mais adequado ao problema se comparados a modelos transparentes menos robustos.

Analisar as funções de forma uma a uma, como ilustrado na Figura 7, pode ser uma forma de ajudar o usuário a visualizar as relações aprendidas pelo modelo. Ademais, graças a aditividade do modelo, o processo de entender "quanto" cada característica contribui para resultado final é mais direto do que em modelos de complexidade total. Assim, o modelo também facilita a visualização de seu processo de inferência para amostras individuais, chamadas de explicações locais, bem como para conjuntos de múltiplas amostras (LOU; CARUANA; GEHRKE, 2012; LOU et al., 2013; NORI et al., 2019).

É possível, por exemplo, observar quais funções f contribuem mais em uma dada predição, ou, até, calcular a contribuição média de uma dada f para um conjunto de múltiplas amostras, oferecendo uma certa noção de como a característica atrelada foi impactante nas predições daquelas observações (LOU; CARUANA; GEHRKE, 2012; NORI et al., 2019).

Figura 7 – Funções de forma



Fonte: Lou, Caruana e Gehrke (2012)

O Quadro 1 resume conceitualmente a relação entre a complexidade de modelos, inteligibilidade e acurácia potencial, onde ML, MLG, MA, MAG e MCT denotam, respectivamente, Modelo Linear, Modelo Linear Generalizado, Modelo Aditivo, Modelo Aditivo Generalizado e Modelo de Complexidade Total. Enquanto modelos lineares são altamente inteligíveis, modelos de alta complexidade ganham em acurácia, ao preço de sua inteligibilidade; por sua vez, Modelos Aditivos e Modelos Aditivos Generalizados apresentam um balanço entre inteligibilidade e acurácia (LOU; CARUANA; GEHRKE, 2012).

Quadro 1 – Inteligibilidade e Acurácia

Modelo	**Fórmula**	**Inteligibilidade**	**Acurácia**
ML	$y = \beta_0 + \beta_1 x_1 + ... + \beta_n x_n$	+++	+
MLG	$g(y) = \beta_0 + \beta_1 x_1 + ... + \beta_n x_n$	+++	+
MA	$y = f_1(x_1) + ... + f_n(x_n)$	++	++
MAG	$g(y) = f_1(x_1) + ... + f_n(x_n)$	++	++
MCT	$y = f(x_1, ..., x_n)$	+	+++

Fonte: Extraído de Lou, Caruana e Gehrke (2012).

Uma forma popular de aprender Modelos Aditivos é empregando o algoritmo *Backfitting* (HASTIE; TIBSHIRANI, 1987; LOU; CARUANA; GEHRKE, 2012). O conceito do algoritmo consiste em, iterativamente, aprender cada função de forma sobre os resíduos das demais funções, na intenção de alcançar um modelo total que possibilite aproximar suas predições de y.

A seguir, o Algoritmo 1 denota o processo para um conjunto de dados $\mathcal{D} = \{(x_i,\ y_i)\}_1^N$, onde $x = (x_1, ..., x_K)$ são vetores com K características e $y \in \mathbb{R}$ é o alvo. Como explicita Lou, Caruana e Gehrke (2012), a primeira função de forma f_1 é aprendida com o objetivo

de predizer y, a segunda (f_2) de predizer os resíduos $y - f_1(x_1)$, a terceira (f_3) os resíduos $y - f_1(x_1) - f_2(x_2)$, e assim por diante até o obter-se as K funções, cada uma modelando uma das K características. Feito isso, a primeira função de forma é descartada e reaprendida nos resíduos das outras $n-1$ funções, e assim por diante. O algoritmo termina com o encerramento do laço externo M, ou utilizando alguma medida de convergência.

Algoritmo 1 Backfitting (regressão)

1: $f_j \leftarrow 0$
2: **para** m $\leftarrow$ 1 **até** M **faça**
3: **para** j $\leftarrow$ 1 **até** K **faça**
4: $\mathcal{R} \leftarrow \{x_{ij},\ y_i - \sum_k f_k(x_{ik})\}_i^N,\ k \neq j$
5: Aprende a função de forma $f_j : x_j \rightarrow y$ usando R como conjunto de treino
6: **fim para**
7: **fim para**

No caso de um problema logístico, como é a tarefa de classificação binária em que $y_i \in \{0, 1\}$, deduzimos a Equação 2.4 a partir de 2.3:

$$\text{logit } p(x) = \sum f_i(x_i) \tag{2.4}$$

onde $p(x) = P(y = 1|x)$ e $\text{logit } p(x) = \log[p(x)/(1 - p(x))]$.

Nesse caso, o modelo é treinado pelo algoritmo de *Local Scoring*, uma generalização do *Backfitting*, proposta por Hastie e Tibshirani (1987). Seja $F(x_i) = \sum_k f_k(x_{ik})$, a Equação 2.5 formula $p(x_i)$ como se segue:

$$\begin{aligned} p(x_i) &= \text{logit}^{-1}(F(x_i)) \\ &= \frac{\exp(F(x_i))}{1 + \exp(F(x_i))} \\ &= \frac{1}{1 + \exp(-F(x_i))} \end{aligned} \tag{2.5}$$

Para classificação binária, o *Local Scoring* consiste, então, em obter as funções f_k da iteração $m + 1$ aproximando a resposta z_i, em vez de y_i, usando o algoritmo *Backfitting* e pesos de observação w_i, de acordo com as Equações 2.6 e 2.7:

$$z_i = F(x_i)^m + \frac{y_i - p(x_i)}{p(x_i)(1 - p(x_i))} \tag{2.6}$$

$$w_i = p(x_i)(1 - p(x_i)) \tag{2.7}$$

Os Modelos Aditivos Generalizados receberam atenção em estudos recentes, cujos ajustes propostos permitiram melhorias de desempenho e resultados próximos de técnicas de

complexidade total (LOU; CARUANA; GEHRKE, 2012; LOU et al., 2013; NORI et al., 2019). As chamadas *Explainable Boosting Machine* e *GAMI-Net* abordadas nas Seções 2.3.1 e 2.3.3, respectivamente, são resultados de alguns desses estudos.

2.3.1 Explainable Boosting Machine

Como apresentado anteriormente, GAM combinam aditivamente uma sequência de funções de forma, cada uma modelando uma característica distinta da entrada, em um modelo maior e mais robusto, que pode ser usado para aproximar comportamentos esperados. Por definição, as funções de forma admitem comportamentos variados, no entanto, considerando a viabilidade do modelo esperado, a escolha da estratégia para obtenção das funções de forma é uma etapa importante para potencializar o desempenho desses modelos.

Assim, o algoritmo GA^{2}M compreende estratégias avançadas que combinam GAM a técnicas modernas de Aprendizado de Máquina (LOU et al., 2013; NORI et al., 2019). *Explainable Boosting Machine* (EBM) é a nomenclatura utilizada a partir da disponibilização pública do algoritmo GA^{2}M (LOU et al., 2013) em conjunto ao *framework* InterpretML (NORI et al., 2019).

Uma das estratégias é a de utilizar árvores de decisão combinadas através de *bagging* para a composição das funções. As árvores de decisão (QUINLAN, 1986) são conhecidas pela sua simplicidade e eficiência de execução, e funcionam tal qual um fluxograma, nesse caso, de decisões tomadas sobre as entradas. Entretanto, suas propriedades de generalização são pobres quando comparadas a outras famílias de modelos, o que as torna limitadas em contextos onde mais capacidade de aprendizado é necessária (ARRIETA et al., 2020). Por outro lado, quando inúmeras delas são arranjadas através de estratégias de combinação (BAUER; KOHAVI, 1999) ganhos de acurácia podem ser observados (LOU; CARUANA; GEHRKE, 2012), através da mitigação de variância do modelo total.

Com esses ganhos em vista, as EBMs utilizam árvores combinadas em uma estratégia de *bagging* (BAUER; KOHAVI, 1999), que visa a obtenção de diversos modelos agindo em cooperação. O conceito envolvido pela técnica consiste em construir classificadores/regressores fortes a partir do arranjo de inúmeros e mais simples classificadores/regressores.

O processo tradicional se baseia em criar múltiplos subconjuntos (amostras com reposição) a partir da base de dados, usando-os para treinar diferentes modelos de predição, cada qual sobre um dos subconjuntos (BAUER; KOHAVI, 1999). Depois de treinados, as predições são combinadas usando média ou votação por maioria, e a partir daí, a predição do modelo, no todo, é obtida.

No caso das EBMs, cada função de forma, individualmente, é uma combinação de inúmeras árvores treinadas via *bagging* e, assim, tentam conciliar eficiência computacional a comportamentos complexos e não lineares para mapear as características de entrada, o que

incrementa o desempenho em problemas variados (LOU; CARUANA; GEHRKE, 2012; LOU et al., 2013).

Outra abordagem incluída nas EBMs diz respeito ao procedimento de treinamento aplicado. De modo geral, Lou, Caruana e Gehrke (2012) propõem uma abordagem baseada em *boosting* de gradiente (FRIEDMAN, 2001; FRIEDMAN, 2002) para treinamento do modelo total. Nela, as funções são aprendidas fazendo uso de resíduos do preditor, tal como no *Backfitting*, no entanto, o algoritmo baseado em *boosting* objetiva, a cada ciclo m, melhorar a aproximação f_j atual combinando-a explicitamente a aproximações f_j anteriores, em vez de substituir absolutamente f_j.

Seja $y \in \{-1, 1\}$ o alvo das predições, o objetivo do algoritmo utilizado é minimizar a função de custo L:

$$L(y, F) = \log(1 + \exp(-2yF)) \tag{2.8}$$

onde $F(x) = \frac{1}{2}\log\left[\frac{P(y=1|x)}{P(y=-1|x)}\right]$ será obtida via estratégia de *boosting*. Desse modo, o procedimento ajusta o modelo utilizando o Algoritmo 2 e as predições podem ser feitas pela estimativa $p(x) = P(y = 1|x) = \frac{1}{1+\exp(-2F(x))}$. O algoritmo vale-se de obter a pseudo-resposta $\hat{y}$, que é usada para ajustar uma nova função que sera combinada à f_j anterior.

Algoritmo 2 *Boosting* de GAM baseados em árvore (classificação)

1: $f_j \leftarrow 0$
2: **para** m $\leftarrow$ 1 **até** M **faça**
3: **para** j $\leftarrow$ 1 **até** K **faça**
4: $\hat{y}_i \leftarrow \frac{2y_i}{1+\exp(2y_i F(x_i))}$, $i = 1, ..., N$
5: Aprende $\{R_{lm}\}_1^L \leftarrow$ uma árvore com L nós folha usando $\{(x_i,\ \hat{y}_i)\}_i^N$ como conjunto de treino
6: $\gamma_{lm} \leftarrow \frac{\sum_{x_{ij} \in R_{lm}} \hat{y}_i}{\sum_{x_{ij} \in R_{lm}} |\hat{y}_i|(2-|\hat{y}_i|)}$, $l = 1, ..., L$
7: $f_j = f_j + \sum_{l=1}^{L} \gamma_{lm} 1(x_{ij} \in R_{lm})$
8: **fim para**
9: **fim para**

Assim como para o *bagging*, as estratégias de *boosting* mitigam variância e contribuem para a acurácia dos modelos. No caso das EBMs, Lou, Caruana e Gehrke (2012) propõem que ambas as técnicas sejam usadas em conjunto para um melhor aproveitamento em acurácia do modelo, além de contar com a possibilidade utilizar interações de pares, apresentadas na Seção 2.3.2.

2.3.2 Interações de pares

Modelos Aditivos Generalizados são modelos estatísticos da forma 2.3, cuja interpretabilidade está relacionada, de um modo geral, ao fato de cada função f ser univariada. Lou et al. (2013) abordam a lacuna de desempenho entre os GAM padrão e os Modelos de Complexidade

Total utilizando o fato de que a Equação 2.3 não modela interações diretas entre características distintas.

O fato de cada função de forma estar associada a uma única característica x_i contribui diretamente para a interpretabilidade do modelo, no entanto Lou et al. (2013) argumentam que seria possível, sem grande perda nesse aspecto, melhorar a acurácia acrescentando-se funções de forma bivariadas e valer-se de técnicas de visualização como, por exemplo, mapas de calor.

Desse modo, Lou et al. (2013) propõem o algoritmo GA^{2}Ms e a utilização de modelos da classe Modelos Aditivos Generalizados com Interações (GAMI), formulados pela Equação 2.9:

$$g(y) = \sum h_i(x_i) + \sum f_{ij}(x_i, x_j) \tag{2.9}$$

onde h são funções de forma univariadas, também chamadas de efeitos principais (YANG; ZHANG; SUDJIANTO, 2021), e f são chamadas de interações de pares.

Nessa variante, os autores introduzem a construção do modelo em duas etapas, usando uma estratégia gananciosa. Na primeira, o melhor GAM é construído normalmente, com base em funções de forma univariadas. Na segunda, as funções do primeiro modelo são corrigidas e funções bivariadas são modeladas nos resíduos de maneira eficiente.

De modo geral, o algoritmo de seleção de pares de características mantém dois conjuntos $\mathcal{S}$ e $\mathcal{Z}$, onde $\mathcal{S}$ contém os pares selecionados até então e $\mathcal{Z}$ os pares restantes. Assim, a cada nova rodada de iteração da segunda etapa, em uma estratégia gulosa, o algoritmo aproxima as funções de pares e tenta selecionar o melhor candidato de $\mathcal{Z}$, ou seja, o que melhor ajusta os resíduos, a partir daí removendo-o de $\mathcal{Z}$ e adicionando-o a $\mathcal{S}$. Assim, o modelo segue se reajustando, calculando resíduos e incluindo novos pares até que não haja melhoria considerável. Afim de evitar o custo computacional alto de vasculhar $\mathcal{Z}$ completamente, Lou et al. (2013) propõem uma heurística baseada em cortes e uma tabela de pesquisa visando acelerar o processo de busca.

2.3.3 GAMI-Net

GAMI-Net (YANG; ZHANG; SUDJIANTO, 2021) é uma arquitetura de redes neurais proposta com a intenção de mesclar aspectos de interpretabilidade presentes em modelos aditivos e a capacidade de redes neurais profundas em aprender comportamentos não-lineares. De modo geral, a arquitetura consiste em múltiplas subredes com camadas escondidas, cada qual correspondendo a uma função de forma distinta, que, então, são combinadas aditivamente, conforme a Equação 2.9 que formula os GAMI.

Para tanto, os autores (YANG; ZHANG; SUDJIANTO, 2021) propõem uma treinamento baseado em gradiente descendente que ajusta os efeitos principais e interações de pares em etapas separadas. Considerando as funções de forma, a arquitetura foi projetada com a

perspectiva de preservar três princípios: esparsidade, no sentido de preservar apenas efeitos principais que forem considerados relevantes; hereditariedade, mantendo apenas interações de pares com pelo menos um efeito principal pai relacionado; e clareza marginal, que diz respeito a tentativa de evitar que efeitos principais e interações de pares se confundam.

Inicialmente, o algoritmo treina somente subredes de efeitos principais, lançando fora aquelas cuja contribuição for considerada de baixa importância, utilizando uma medida calculada. Basicamente, a importância de uma dada função de forma D pode ser obtida com base em sua variância amostral, conforme propõem os autores Yang, Zhang e Sudjianto:

$$D(h_i) = \frac{1}{n-1} \sum h_i^2(x_i) \tag{2.10}$$

$$D(f_{ij}) = \frac{1}{n-1} \sum f_{ij}^2(x_i, x_j) \tag{2.11}$$

Quanto mais efeitos principais são incluídos no modelo, menor tende a ser o erro de validação, no entanto, incluir efeitos principais demasiadamente poderia culminar em *overfitting*, dessa forma, a ideia consiste em limitar o modelo apenas a funções de forma que não sejam triviais, estimulando a parcimônia da modelagem em questão (YANG; ZHANG; SUDJIANTO, 2021). Isso é realizado ranqueando os efeitos com base na sua na importância D e podando os que contribuem menos com base em um limite de tolerância que leva em conta o erro de validação (YANG; ZHANG; SUDJIANTO, 2021).

Em um segundo momento, o algoritmo treina um conjunto de interações de pares, onde, de maneira similar, o mesmo princípio de esparsidade é aplicado, além da restrição de hereditariedade mencionada anteriormente. O algoritmo seleciona um número fixo de interações de pares usando o procedimento de ranqueamento proposto por (LOU et al., 2013) e retreina a arquitetura com o acréscimo destas interações e os efeitos principais congelados, minimizando a função de custo L utilizando gradiente descendente (YANG; ZHANG; SUDJIANTO, 2021):

$$L(\theta) = l(\theta) + \lambda \sum_{i \in S_1} \sum_{(i,j) \in S_2} \Omega(h_i, f_{ij}) \tag{2.12}$$

onde os conjuntos ativos de efeitos principais e interações de pares S_1 e S_2 são determinados sujeitos às restrições de esparsidade e hereditariedade e o custo $l(\theta)$ é determinado pela tarefa em questão (por exemplo, regressão ou classificação). O segundo termo é a regularização da clareza marginal, onde $\lambda \geq 0$ é o fator de regularização e Ω é definido como se segue (YANG; ZHANG; SUDJIANTO, 2021):

$$\Omega(h_i, f_{ij}) = \left| \frac{1}{n} \sum h_i(x_i) f_{ij}(x_i, x_j) \right| \tag{2.13}$$

Segundo propõem YANG; ZHANG; SUDJIANTO, quanto menor o valor de $\Omega(h_j, f_{jk})$, mais claramente o efeito marginal h_j é separado de sua interação filha f_{jk}, dessa forma, o

termo de regularização da Equação (2.12) seria responsável por penalizar a não-ortogonalidade denotada por Ω, incentivando a chamada clareza marginal.

Por fim, em um terceiro e último estágio, todas as subredes da arquitetura, incluindo efeitos principais e interações de pares, são retreinadas simultaneamente em um procedimento de refinamento – também sujeito a Equação (2.12), visando consolidar o desempenho preditivo do modelo e contornar possíveis vieses deixados pela remoção de efeitos principais ou interações de pares.

Com relação a interpretabilidade, os autores sugerem a utilização da Razão de Importância (do inglês, *Importance Ratio* ou IR) para estimar quantitativamente a contribuição de efeitos principais e interações de pares. O IR de cada efeito principal e interação de pares pode ser calculado, respectivamente, pelas Equações (2.14) e (2.15):

$$\text{IR}(i) = D(h_i)/T, \tag{2.14}$$

$$\text{IR}(i,j) = D(f_{ij})/T. \tag{2.15}$$

onde $T = \sum_{i \in S_1} D(h_i) + \sum_{(i,j) \in S_2} D(f_{ij})$. Os IRs de todas as funções de forma somados igualam a um. Na prática, a importância das funções pode ser ordenada com os valores de IR em ordem decrescente para ranquear efeitos principais e funções de pares em importância (YANG; ZHANG; SUDJIANTO, 2021).

3 Metodologia

Trabalhos como o de Sarkhel et al. (2020) e Ghodratnama et al. (2020), discutidos na Seção 2.2.3, dão mostras de que a combinação do uso adequado de características elaboradas e um modelo com alguma transparência pode indicar um caminho para obter interpretabilidade em modelos de PLN e, mais especificamente, Sumarização Automática.

Em paralelo, no contexto da interpretabilidade, as EBM e GAMI-Net foram apresentadas com a proposta de aliar a transparência de algoritmos estatísticos mais simples a robustez de técnicas complexas de Aprendizado de Máquina, explorando diretamente o comportamento das características de entrada, possibilitando um modelo final tão interpretável quanto as características de entrada e sua engenharia permitirem.

Com isso em vista, este trabalho investigou a aplicabilidade dos algoritmos EBM e GAMI-Net ao problema de SE e avaliou seu desempenho enquanto algoritmos de sumarização. O capítulo tem como objetivo descrever a abordagem proposta (3.1), detalhes da extração de características via PLN (3.2), bases de dados utilizadas (3.3) e detalhes de experimentação (3.4).

3.1 Abordagem proposta

Tanto EBMs quanto GAMI-Net podem ser configuradas para problemas de classificação ou regressão com um ajuste adequado na função de ligação. No entanto, dada a natureza pouco estruturada que documentos e resumos podem apresentar, é necessário que isso seja adequadamente considerado durante a elaboração da solução, para que o problema de SAT possa ser abordado pelas técnicas.

Uma das formas de abordar o problema de SE é observá-lo como um problema probabilístico, onde é desejado computar o quão relevantes as sentenças do documento de entrada são para o resumo projetado (WONG; WU; LI, 2008), classificando-as como pertinentes ou não. Solucionar o problema de SAT dessa forma seria uma maneira conveniente – no sentido de ser mais direta, simplificando o desenho da solução – de permitir a utilização de GAMI na tarefa de sumarização.

Assim, neste trabalho, instâncias de GAMI, nesse caso EBM e GAMI-Net, são utilizadas dessa forma, como mecanismos que decidem se as sentenças são relevantes ou não ao resumo que será obtido, na forma de um problema supervisionado, trazendo consigo a interpretabilidade atrelada à sua utilização. Para que os GAMI cumpram seu papel enquanto motores de decisão, sem deixar de trazer explicabilidade à solução, é necessário um treinamento baseado em características de entrada bem definidas, e sua transparência está vinculada a interpretabilidade

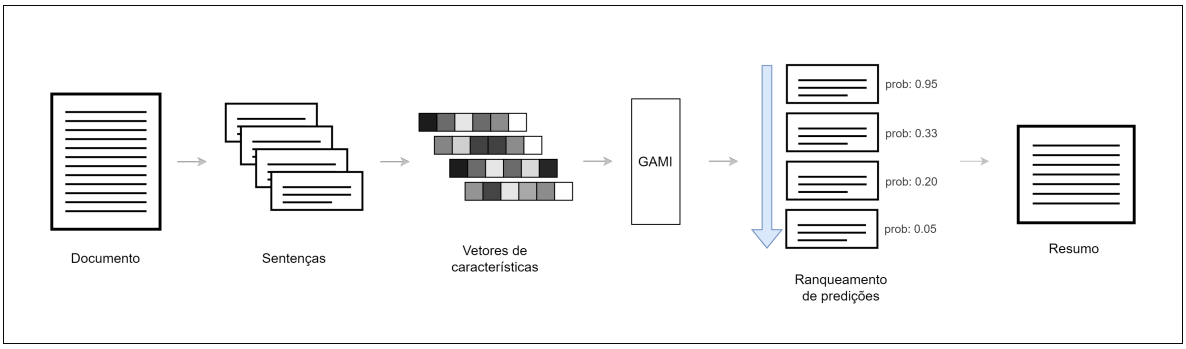
de tais características. Desse modo, simplificar o desenho da solução e a etapa de engenharia de características é importante para equilibrar essas questões.

Seja $D = \{s_0, ..., s_n\}$ um documento composto por uma sequência de sentenças s_i. O objetivo do processo de sumarização aplicado é obter uma sequência S das sentenças mais relevantes em D, onde S é limitado em comprimento para ser mais curto que D. Para cada sentença s_i, é extraído um vetor x de K características fixas envolvendo atributos de s_i com relação a D, extraídas via PLN, conforme descreve a Seção 3.2.

O procedimento de treinamento consiste em minimizar o erro binário de classificação, em uma abordagem onde pares de vetores x e suas respectivas classes-alvo y, sinalizando exclusão ou inclusão das sentenças em seus respectivos resumos, são utilizados como conjunto de treinamento. Após o treinamento, a capacidade do modelo de distinguir entre sentenças "importantes" e "não-importantes" para, dado um documento de entrada, ranquear e selecionar sentenças apropriadas dentre as demais é utilizada. Note que abordar o problema desse modo não é contribuição deste trabalho (vide Nallapati, Zhai e Zhou (2017), Kedzie, Mckeown e III (2018) e Xiao e Carenini (2019), como exemplos), mas sim, a utilização de GAMI como tal mecanismo.

O processo de ranqueamento consiste em obter a probabilidade de uma sentença ser parte do resumo dado, para cada uma das sentenças do documento. Então, o resumo do documento é obtido selecionando as sentenças com ranque mais alto, conforme sua ordem de aparição no documento de entrada, até atingir o limite estabelecido de comprimento, desse modo, a ordem lógica na qual os assuntos aparecem no documento é preservada no resumo. A Figura 8 ilustra as etapas do processo de elaboração de um resumo, após treinamento.

Figura 8 – Processo de sumarização utilizando GAMI



Fonte: elaborada pelo autor.

3.2 Extração de características

As características utilizadas para treinar algoritmos GAMI ditam muito da interpretabilidade do modelo, já que a proposta desses algoritmos é justamente utilizar o comportamento mapeado através das características como forma de elucidar o processo das decisões tomadas. Dito isso, visando interpretabilidade para o problema de SE, é desejável que o conjunto de características alie riqueza na representatividade das sentenças de origem à significados mais diretos – e interpretáveis, tanto quanto for possível.

Assim, em contrapartida a utilização de representações densas ou de alta dimensionalidade como vetores de palavras (por exemplo, Word2Vec (MIKOLOV et al., 2013)) que pudessem depender de treinamento apartado e cujo significado possa ser visto como abstrato, este trabalho adota a utilização de um conjunto menor de características, baseadas em estatísticas mais elementares acerca das sentenças, cuja simplicidade deveria contribuir tanto para a interpretabilidade do modelo quanto sua eficiência.

A etapa de pré-processamento é responsável por transformar documentos brutos em sequências de sentenças, capturando informações úteis para posterior extração de características, em si. Nela, o processo inicia pela segmentação de documentos brutos em sentenças, que, por sua vez, passam pelo processo de *tokenização*, isto é, cada sentença é segmentada em uma sequência de termos ou *tokens*. Em seguida, é executada a remoção de pontuação e *stopwords*, ou palavras de parada, que são termos frequentes da língua que acrescentam pouco significado para as sentenças. Depois, é realizada a marcação de nomes próprios e termos numéricos, e as palavras são submetidas ao processo de *stemming*, que almeja reduzir as palavras ao seu radical, removendo possíveis prefixos e sufixos.

No desenvolvimento do trabalho, esse processo é realizado com o auxílio da biblioteca Python spaCy (MONTANI et al., 2021), com exceção do procedimento de *stemming*, realizado com apoio do módulo SnowballStemmer da biblioteca NLTK (BIRD; KLEIN; LOPER, 2009).

3.2.1 Definições das características utilizadas

Depois do pré-processamento, seis características são extraídas das sentenças de modo a obter os vetores $x = \{x_1, x_2, ..., x_6\}$ utillizados para treinamento e predição, conforme as definições apresentadas a seguir.

3.2.1.1 TF-ISF

O TF-ISF é uma variante do método TF-IDF aplicada em nível de sentença para sumarização de texto (OLIVEIRA et al., 2016; MUTLU; SEZER; AKCAYOL, 2019). A ideia é computar uma pontuação para cada sentença, com base na importância e na descritividade dos termos dentro do documento (OLIVEIRA et al., 2016), que são medidos pela frequência de termo (TF) e frequência inversa de sentença (ISF) para os termos. No presente trabalho, é

utilizado um TF-ISF baseado em bigramas – sequências de dois termos adjacentes dada uma sentença *tokenizada*, de modo que cada sentença s_i de um documento receba uma pontuação de saliência (Equação 3.2):

$$w(s_i) = \sum_{j=1}^{J_i} \left[\mathrm{F}(b_j) \times \log \left(\frac{n}{n_{b_j}} \right) \right], \tag{3.1}$$

$$x_1(s_i) = \frac{w(s_i)}{\max(w(s_i))}. \tag{3.2}$$

onde $\mathrm{F}(b_j)$ é a frequência do bigrama b_j no documento, n é o número de sentenças do documento, n_{b_j} é o número de sentenças documento em que b_j ocorre e J_i é o número de bigramas distintos em s_i.

3.2.1.2 Posição

Considerando que a ordem em que as sentenças aparecem pode fornecer informações importantes sobre sua relevância (FERREIRA et al., 2013; OLIVEIRA et al., 2016), a característica *position* (Equação 3.3) representa a posição da sentença dentro do documento:

$$x_2(s_i) = \frac{p_i}{n}, \quad 1 \leq p_i \leq n. \tag{3.3}$$

onde p_i é a posição da sentença s_i no documento.

3.2.1.3 Comprimento

O característica *length* (Equação 3.4) é calculada com base no comprimento da sentença s_i em termos, em relação ao comprimento máximo de sentença do documento relacionado (OLIVEIRA et al., 2016; MUTLU; SEZER; AKCAYOL, 2019):

$$x_3(s_i) = \frac{\text{número de termos na sentença } s_i}{\max(\text{número de termos em uma sentença})}. \tag{3.4}$$

3.2.1.4 Nomes próprios e númericos

A proporção individual de nomes próprios e termos numéricos na frase s_i pode indicar a presença de informações relevantes (OLIVEIRA et al., 2016). As características respectivas são calculadas da seguinte maneira:

$$x_4(s_i) = \frac{\text{número de nomes próprios em } s_i}{\text{número de termos em } s_i} \quad \text{e} \tag{3.5}$$

$$x_5(s_i) = \frac{\text{número de termos numéricos em } s_i}{\text{número de termos em } s_i}. \tag{3.6}$$

3.2.1.5 Similaridade sentença-sentença

A similaridade sentença-sentença denota quão semelhante uma sentença é das demais sentenças no documento (MUTLU; SEZER; AKCAYOL, 2019). A característica *cos_sims_uni* é calculada usando a similaridade de cosseno c entre sentenças como denota a Equação 3.7:

$$x_6(s_i) = \frac{\sum_{j=1}^{n} c(s_i, s_j)}{\max_{s_k}(\sum_{j=1}^{n} c(s_k, s_j))}, \quad i \neq j. \tag{3.7}$$

onde s_k é a k-ésima sentença do documento, que maximiza o denominador.

3.3 Bases de dados

Neste trabalho, EBM e GAMI-Net são comparadas a outras abordagens em dois conjuntos de dados públicos de sumarização de texto, CNN/Dailymail (HERMANN et al., 2015; SEE; LIU; MANNING, 2017) e Pubmed (COHAN et al., 2018). Essas bases têm sido adotadas em trabalhos recentes de SAT, especialmente por abordagens baseadas em redes neurais recorrentes e *Transformers*, devido ao número grande de documentos presentes nas bases.

A base CNN/Dailymail (NALLAPATI et al., 2016; HERMANN et al., 2015) conta com pares de artigos de notícia em língua inglesa e seus respectivos resumos, compostos pela configuração padrão de aproximadamente 287,1 mil pares treinamento, 13,4 mil pares de validação e 11,5 mil pares de teste. Nela, os documentos possuem uma média de 781 tokens, enquanto os resumos possuem uma média de 56 tokens (SEE; LIU; MANNING, 2017). No desenvolvimento deste trabalho, foi utilizada a versão não anonimizada do conjunto de dados (SEE; LIU; MANNING, 2017).

Por sua vez, a base PubMed (COHAN et al., 2018) é uma coleção de artigos científicos em língua inglesa nos quais a seção de resumo é usada como referência para sumarização, na configuração de 115,5 mil pares de treino, 6,6 mil pares de validação e 6,6 mil pares de teste. Este conjunto de dados tem sido usado para avaliar abordagens de sumarização de documentos longos, já que tanto documentos quanto os resumos são, em geral, mais longos que bases populares como a CNN/Dailymail, com uma média 3016 tokens para documentos e 203 tokens para resumos (XIAO; CARENINI, 2019).

Conforme mencionado na Seção 3.1, a abordagem proposta utiliza sentenças individuais como instâncias de entrada para o treinamento. Considerando a necessidade de rótulos extrativos baseados em sentenças para execução desta etapa, no desenvolvimento do trabalho, foram utilizados rótulos sintéticos obtidos via heurísticas automáticas com base nos resumos de referência, uma vez que, originalmente, ambas as bases contariam apenas com resumos abstrativos. Apesar de não ser ideal, este tipo de abordagem tem sido uma estratégia recorrente para obtenção de rótulos de treinamento na ausência de rotulação humana de resumos extrativos.

Dese modo, o presente trabalho faz uso de estratégias já utilizadas por outros autores para a obtenção dos rótulos artificiais de ambas as bases (NALLAPATI; ZHAI; ZHOU, 2017; KEDZIE; MCKEOWN; III, 2018; XIAO; CARENINI, 2019; LIU, 2019).

Em suma, tais estratégias baseia-se em uma seleção gulosa de sentenças do documento em questão para um conjunto extrativo, maximizando a pontuação ROUGE entre o conjunto e o resumo abstrativo de referência a cada iteração. Ao fim, as sentenças inclusas no conjunto recebem o rótulo positivo, enquanto as demais recebem o negativo. Neste trabalho, para CNN/Dailymail, os rótulos foram gerados utilizando os mesmos scripts fornecidos por Liu (2019)[1] e, para Pubmed, são utillizados os rótulos extraídos e tornados públicos por Xiao e Carenini (2019)[2]. O Quadro 2 traz um exemplo de documento, resumo de referência e resumo obtido via heurística. Além disso, a segmentação de sentenças pode resultar em pequenas inconsistências de sentido no resumo sintético, ocasionada por eventuais quebras dentro de uma mesma frase (no quadro, entre a sentença 2 e a sentença 3).

Quadro 2 – Exemplo de documento e resumo da base CNN/Dailymail

Documento original (segmentado em sentenças)
1. (CNN) For the first time in eight years , a TV legend returned to doing what he does best . 2. Contestants told to " come on down ! " 3. on the April 1 edition of " The Price Is Right " encountered not host Drew Carey but another familiar face in charge of the proceedings . 4. Instead , there was Bob Barker , who hosted the TV game show for 35 years before stepping down in 2007 . 5. Looking spry at 91 , Barker handled the first price - guessing game of the show , the classic " Lucky Seven , " before turning hosting duties over to Carey , who finished up, 6. Despite being away from the show for most of the past eight years , Barker did n't seem to miss a beat .
Resumo original
Bob Barker returned to host " The Price Is Right " on Wednesday . Barker , 91 , had retired as host in 2007
Resumo sintético
On the April 1 edition of " The Price Is Right " encountered not host Drew Carey but another familiar face in charge of the proceedings . Instead , there was Bob Barker , who hosted the TV game show for 35 years before stepping down in 2007 .

Fonte: Extraído da base CNN/Dailymail

[1] https://github.com/nlpyang/BertSum
[2] https://github.com/Wendy-Xiao/Extsumm_local_global_context

3.4 Detalhes da experimentação

Neste trabalho, EBM e GAMI-Net são comparadas aos resultados reportados por outras abordagens recentes, boa parte das quais produzida por arquiteturas neurais profundas, no sentido de delinear a capacidade de sumarização da abordagem proposta em contraste com esses modelos, apesar da diferença assumida em termos de interpretabilidade. Além disso, comparamos os modelos EBM e GAMI-Net a outros classificadores de Aprendizado de Máquina supervisionado, nominalmente, Regressão Logística (LR), Floresta Aleatória (RF) e *XGBoost*, usando o mesmo procedimento para treinamento e predição descritos na Seção 3.1, cada qual treinado e testado dez vezes e as pontuações médias são consideradas para fins de comparação.

Os modelos de EBM foram treinados com auxílio da biblioteca InterpretML (NORI et al., 2019) [3], parametrizados com 8 *inner bags* – número de amostras utilizadas via *bagging* para o elaboração das árvores obtidas via *boosting* – e 10 interações de pares. Semelhantemente, os modelos GAMI-Net foram treinados com um valor máximo de 10 interações de pares, utilizando a implementação disponibilizada pelos autores (YANG; ZHANG; SUDJIANTO, 2021)[4]. Os modelos de RF e XGBoost foram treinados com 100 estimadores, utilizando a biblioteca scikit-learn[5]. Demais parâmetros foram configurados pelos valores padrão. Além disso, adotamos subamostragem aleatória para lidar com o desequilíbrio de rótulos durante a etapa de treinamento.

As abordagens, em geral, foram avaliadas usando a métrica de pontuação ROUGE (LIN, 2004), considerando sua ampla adoção para sistemas de SE, com relação aos resumos originais das bases em questão. Foram consideradas pontuações ROUGE$_n$ para $n = 1$ (*R-1*) e $n = 2$ (*R-2*), além de ROUGE$_L$ (*R-L*). Além disso, avaliamos a capacidade de seleção de sentenças dos classificadores tradicionais, calculando pontuações F1 (Equação (3.10)) com base nas predições para sentenças dos resumos obtidos e rótulos artificiais:

$$Prec = \frac{vp}{vp + fp}, \tag{3.8}$$

$$Rec = \frac{vp}{vp + fn}, \tag{3.9}$$

$$F1 = \frac{2 * Prec * Rec}{Prec + Rec}. \tag{3.10}$$

onde vp denota verdadeiros positivos, fp falsos positivos e fn falsos negativos.

A linha de base *Lead* corresponde à pontuação de atribuir as primeiras sentenças presentes nos documentos aos resumos (respeitando o limite de comprimento respectivo das bases de dados) e *Oracle* denota as pontuações obtidas pelos rótulos artificiais, que

[3] https://github.com/interpretml/interpret
[4] https://github.com/interpretml/interpret
[5] https://scikit-learn.org/

corresponde ao limite superior de utilizá-los. As pontuações ROUGE foram calculadas utilizando a biblioteca pyrouge[6], uma interface em Python para os scripts ROUGE-1.5.5 originais. Em relação aos tamanhos, os resumos CNN/Dailymail foram limitados arbitrariamente a três sentenças (ZHONG et al., 2020) enquanto os resumos Pubmed foram limitados a 200 palavras (XIAO; CARENINI, 2019), para fins de cálculo da medida ROUGE e comparação com outros resultados reportados.

[6] https://pypi.org/project/pyrouge/

4 Resultados e discussão

As Tabelas 1 e 2 apresentam os resultados dos modelos considerados para as bases de dados CNN/Dailymail e Pubmed, respectivamente. A coluna T indica o tipo do modelo apresentando, que pode ser de SA ou SE. A coluna I denota a categoria de interpretabilidade do modelo em questão, entre baixa (B) e alta (A). Nas demais colunas, temos as pontuações ROUGE e F1, conforme apresentado.

Conforme mostra a Tabela 1, comparando com outras abordagens, os modelos EBM e GAMI-Net foram capazes de competir com as redes SummaRuNNer (NALLAPATI; ZHAI; ZHOU, 2017) e Pointer-Generator (SEE; LIU; MANNING, 2017) – duas arquiteturas profundas baseadas em RNN, superando a primeira em relação a R-L e ambas considerando R-2 no conjunto de dados CNN/Dailymail. Por outro lado, não conseguiram superar BART+RD (WU et al., 2021) e MatchSum (ZHONG et al., 2020), exemplares baseados em *Transformer* ou mesmo o ExDoS (GHODRATNAMA et al., 2020), em termos de pontuações. Na base de dados Pubmed, como mostra a Tabela 2, EBM e GAMI-Net alcançaram pontuações R-L mais altas que o SummaRuNNer, mas falharam em competir com ExtSum-LG (XIAO; CARENINI, 2019) e ExtSum-LG+MMR-S+ (XIAO; CARENINI, 2020), que são modelos especializados em sumarização de documentos longos. Além disso, de maneira geral, EBM e GAMI-Net obtiveram resultados semelhantes em ambos os conjuntos de dados. Considerando as pontuações ROUGE, a arquitetura GAMI-Net está à frente no CNN/Dailymail enquanto a EBM é superior no Pubmed, por menos de 0,1% para cada variante.

Tabela 1 – Resultados para a base CNN/Dailymail.

Modelo	**T**	**I**	**ROUGE F (%)**			**F1 (%)**
			R-1	***R-2***	***R-L***	
Lead	–	–	40.12	17.54	36.30	–
Oracle	–	–	56.09	33.67	52.21	–
P-Gen (SEE; LIU; MANNING, 2017)	SA.	B	39.53	17.28	36.38	–
BART + RD (WU et al., 2021)	SA.	B	44.51	21.58	41.24	–
SummaRuNNer (GHODRATNAMA et al., 2020)	SE.	B	39.90	16.30	35.10	–
MatchSum (ZHONG et al., 2020)	SE.	B	44.41	20.86	40.55	–
ExDoS (GHODRATNAMA et al., 2020)	SE.	A	42.1	18.9	37.7	–
LR	SE.	A	38.51	16.66	34.74	31.96
RF	SE.	B	39.46	17.34	35.71	32.71
XGBoost	SE.	B	39.58	17.50	35.82	33.52
EBM	SE.	A	39.48	17.40	35.74	33.40
GAMI-Net	SE.	A	39.52	17.42	35.76	33.40

Tabela 2 – Resultados para a base PubMed.

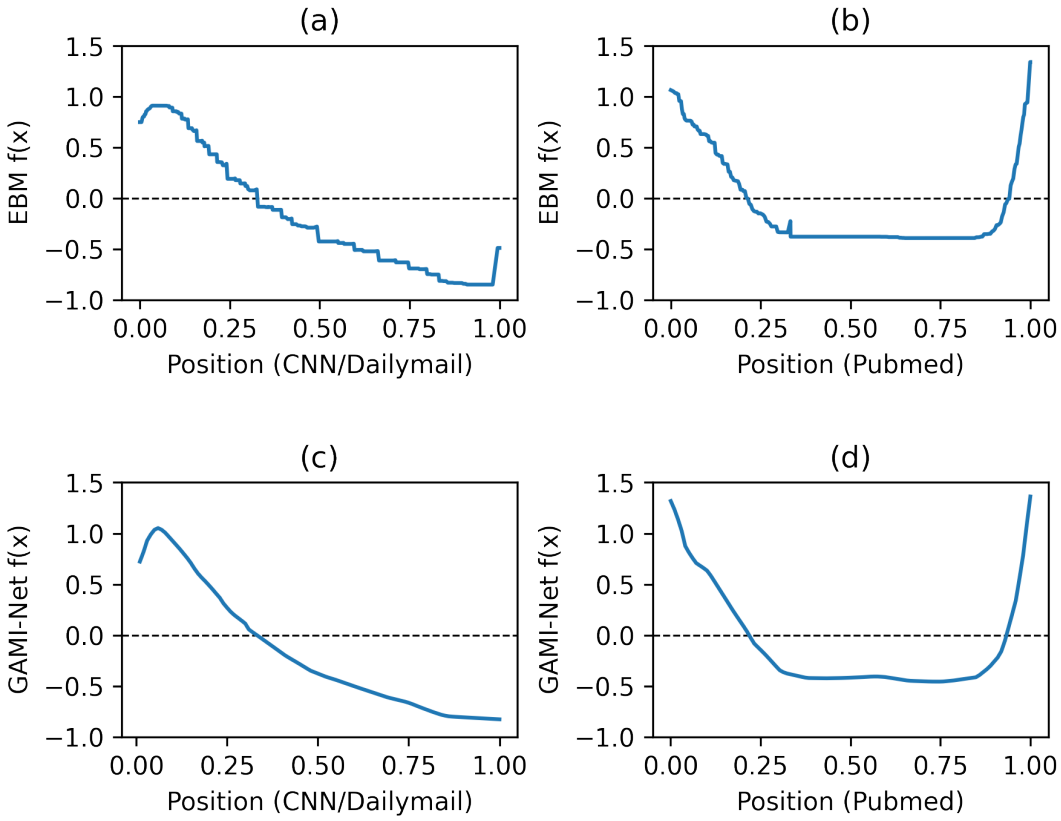
Modelo	T	I	ROUGE F (%)			F1
			R-1	*R-2*	*R-L*	(%)
Lead	–	–	37.38	12.65	33.71	–
Oracle	–	–	55.37	26.31	49.07	–
Discourse-aware (COHAN et al., 2018)	SA.	B	38.93	15.37	35.21	–
SummaRuNNer (XIAO; CARENINI, 2019)	SE.	B	43.89	18.78	30.36	–
ES-LG (XIAO; CARENINI, 2020)	SE.	B	45.18	20.20	40.72	–
ES-LG+MMR-S+ (XIAO; CARENINI, 2020)	SE.	B	45.39	20.37	40.99	–
LR	SE.	A	38.07	12.70	32.94	27.61
RF	SE.	B	40.16	14.13	34.98	30.26
XGBoost	SE.	B	40.16	14.18	34.97	30.86
EBM	SE.	A	39.86	13.96	34.65	30.70
GAMI-Net	SE.	A	39.78	13.92	34.57	30.76

Embora EBM e GAMI-Net tenham ficado abaixo de algumas abordagens em termos de pontuações ROUGE, elas ainda contam com a capacidade de fornecer níveis mais altos de transparência acerca de seu aprendizado. Para SE, tal habilidade poderia ser útil para esclarecer o que está sendo priorizado pelos modelos enquanto decide a importância das sentenças. A Figura 9 apresenta um gráfico de função de forma construído sobre a característica *Posição* (Equação 3.3) considerando exemplares de modelo e a base de dados em questão, onde os eixos horizontais representam valores de entrada das funções de forma, ou seja, da característica atrelada, e os eixos verticais denotam as saídas das funções. Na prática, esse tipo de visualização permite uma investigação mais aprofundada dos modelos de sumarização, por exemplo, possibilitando a inspeção de como alterações nos modelos treinados, nesse caso, de base de dados, impacta no comportamento aprendido.

Como mostra a Figura 9, as funções de forma aprendidas na CNN/Dailymail (referências 9a e 9c) tendem a gerar saídas maiores para sentenças no início dos documentos, diferentemente do que acontece para a base Pubmed (referências 9b e 9d), onde tanto sentenças do início quanto do final dos documentos estão mais suscetíveis a isso. Nesse caso, isso mostra que os modelos treinados na CNN/Dailymail, uma base de notícias, tendem a evitar sentenças do final dos documentos ao procurar por relevantes, o que geraria impacto especialmente se a posição for o fator decisivo para desempatar a probabilidade entre duas sentenças. Tal comportamento pode diferir para outros tipos de documentos, como é o caso de modelos treinados na Pubmed, composta de artigos científicos – onde introdução e conclusão usualmente destacam pontos principais. De maneira mais indireta, esse viés também pode ser observado com base nos resultados *Lead* (Tabelas 1 e 2), que representam uma linha de base mais forte no conjunto CNN/Dailymail.

A capacidade de modelos como EBM e GAMI-Net de capturar esse tipo de propriedade

Figura 9 – Função de forma da característica *Posição* (Eq. 3.3).

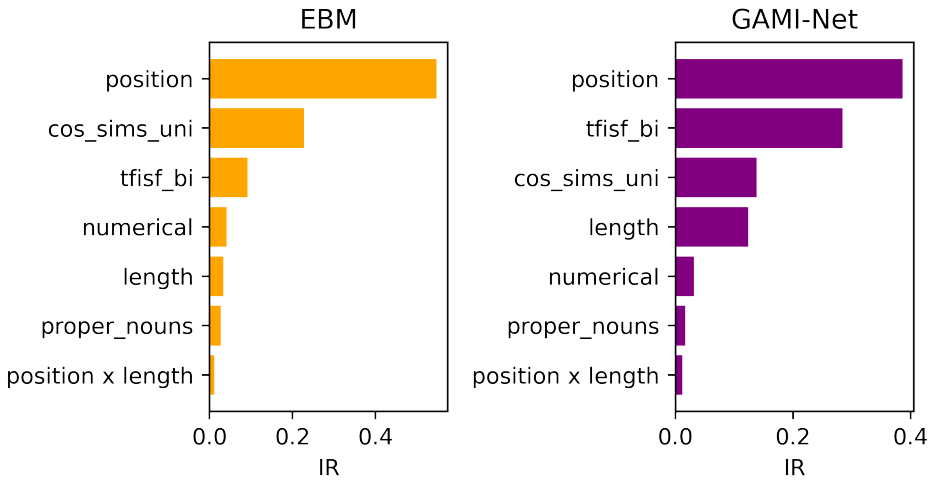


Fonte: elaborada pelo autor.

intrinsecamente, concedendo a possibilidade de exploração adicional das próprias funções de decisão, em si, pode ser vista como um benefício quando comparados aos modelos de baixa interpretabilidade. Apesar da robustez aparente de outros modelos, a ausência de transparência pode deixar as interpretações mais "imaginativas" do usuário a respeito de seu comportamento; não é possível afirmar exatamente "o que" estes aprendem, quais vieses absorvem e onde se diferenciam dos demais.

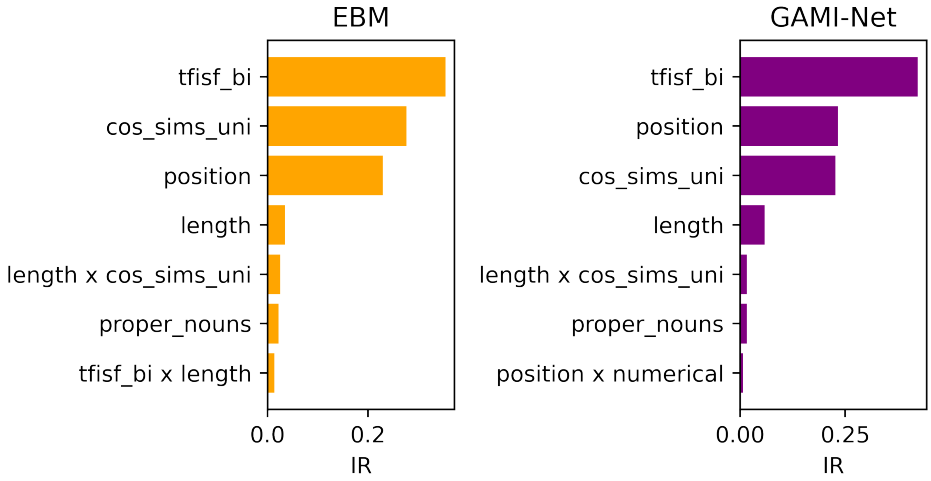
Além disso, as saídas de funções de forma de EBM e GAMI-Net podem ser comparadas para previsões únicas ou um grupo de amostras, ajudando a entender melhor as contribuições de cada característica ao fazer predições. As Figuras 10 e 11 mostram as características mais contributivas quantificadas por seu IR (YANG; ZHANG; SUDJIANTO, 2021) no conjunto de teste considerando os conjuntos de dados CNN/Dailymail e Pubmed, colocando lado a lado EBM e GAMI-Net. No exemplo, as funções de forma das características *TF-ISF* (Equação 3.2) e *Posição* (Equação 3.3) e *Similaridade sentença-sentença* (Equação 3.7) são, em geral, predominantes em importância sobre as demais. Nos exemplares em questão, as demais funções de forma, isto é, com IR menor que as apresentadas, foram omitidas da figura.

Figura 10 – Top-7 funções de forma em Razão de Importância (IR) na base CNN/Dailymail.



Fonte: elaborada pelo autor.

Figura 11 – Top-7 funções de forma em Razão de Importância (IR) na base Pubmed.



Fonte: elaborada pelo autor.

Comparando com os demais classificadores binários, tanto EBM quanto GAMI-Net se posicionaram entre os modelos XGBoost e Regressão Logística, reforçando sua posição como um equilíbrio entre poder preditivo e transparência na tarefa de SE, com base nas características utilizadas. Uma limitação importante de utilizar essa abordagem binária de sentenças para SE é a ausência de um mecanismo explícito para selecionar bons "conjuntos" de sentenças, em vez de sentenças individualmente relevantes. Além disso, a lacuna de informação semântica nas características da abordagem utilizada, decorrente do desafio de incorporá-la por características de baixa dimensionalidade, pode ser uma das razões que a distancia dos modelos mais robustos como os baseados em *Transformers*.

5 Conlusões

O desafio de construir modelos interpretáveis é considerável e ainda foi pouco explorado na literatura de Sumarização Automática de Texto. Em um histórico recente, em especial, após a popularização de grandes bases de dados publicamente, diferentes modelos de Sumarização Extrativa ganharam níveis de complexidade que em geral comprometem sua interpretabilidade. No contexto da Sumarização Automática de Texto, a interpretabilidade de modelos poderia oferecer benefícios, como melhorar as percepções das limitações e capacidades de modelos de sumarização, esclarecendo o que o modelo realmente satisfaz em contraste com as expectativas do usuário, ou mesmo ajudando a obter *insights* sobre o problema de sumarização em si.

Neste trabalho, uma abordagem que aplica modelos EBM e GAMI-Net à Sumarização Extrativa foi apresentada, como uma alternativa interpretável simples, mas atraente aos algoritmos de classificação tradicionais. Os resultados mostram que, apesar de mais restritivos que modelos de maior complexidade em termos de formulação, os dois tipos de GAMI foram capazes de alcançar resultados semelhantes a certos modelos caixa-preta. Ademais, comparando a eficácia de EBM e GAMI-Net entre si, não houve clara superioridade de uma com relação à outra, uma vez que, no geral, suas pontuações foram razoavelmente semelhantes para as bases de dados CNN/Dailymail e PubMed.

Embora a necessidade de engenharia de características possa ser vista como uma desvantagem ao comparar abordagens tradicionais com modelos neurais, com um conjunto conciso de características, os modelos EBM e GAMI-Net mostraram resultados promissores para Sumarização Extrativa. A combinação de características inteligíveis e a transparência dos GAMI pode ser uma ferramenta para iluminar a visão do processo decisivo empregado durante o processo de Sumarização Automática de Texto.

Este trabalho pode ser considerado um esforço introdutório sobre o tópico de Sumarização Extrativa interpretável baseada em GAMI e as percepções aqui apresentas podem vir a ajudar pesquisas futuras explorando o assunto de inteligibilidade para sistemas de Sumarização Automática de Texto. Algumas possibilidades para trabalhos futuros poderiam ser explorar uma maneira sistemática de estudar um grupo maior de características para utilização de GAMI, ou, até, investir na elaboração de uma abordagem que considere escolher sentenças em conjunto, trazendo interpretações contextuais, em nível de resumo, por exemplo.

5.1 Publicação realizada

No decorrer do desenvolvimento do presente projeto de mestrado, um artigo científico foi elaborado e apresentado na *18th International Joint Conference on Computer Vision, Imaging*

and Computer Graphics Theory and Applications. A publicação pode ser encontrada através da referência:

- SILVA, V.; PAPA, J.; DA COSTA, K. Extractive Text Summarization Using Generalized Additive Models with Interactions for Sentence Selection. In: *Proceedings of the 18th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 4: VISAPP*, p. 737-745, 2023. ISBN 978-989-758-634-7. ISSN 2184-4321. DOI: 10.5220/0011664100003417.

Referências

ALLAHYARI, M. et al. Text summarization techniques: a brief survey. *arXiv preprint arXiv:1707.02268*, 2017. Citado na página 25.

ARRAS, L. et al. "what is relevant in a text document?": An interpretable machine learning approach. *PloS one*, Public Library of Science San Francisco, CA USA, v. 12, n. 8, p. e0181142, 2017. Citado na página 33.

ARRIETA, A. B. et al. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, Elsevier, v. 58, p. 82–115, 2020. Citado 7 vezes nas páginas 26, 27, 28, 29, 30, 31 e 38.

BACH, S. et al. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, Public Library of Science, v. 10, n. 7, p. e0130140, 2015. Citado 2 vezes nas páginas 31 e 32.

BAHDANAU, D.; CHO, K.; BENGIO, Y. Neural machine translation by jointly learning to align and translate. In: *3rd International Conference on Learning Representations, ICLR 2015*. [S.l.: s.n.], 2015. Citado na página 33.

BAUER, E.; KOHAVI, R. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine learning*, Springer, v. 36, n. 1, p. 105–139, 1999. Citado na página 38.

BIRD, S.; KLEIN, E.; LOPER, E. *Natural language processing with Python: analyzing text with the natural language toolkit*. [S.l.]: "O'Reilly Media, Inc.", 2009. Citado na página 45.

BLUM, A.; MITCHELL, T. Combining labeled and unlabeled data with co-training. In: *Proceedings of the eleventh annual conference on Computational learning theory*. [S.l.: s.n.], 1998. p. 92–100. Citado na página 23.

BROMLEY, J. et al. Signature verification using a "siamese" time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, World Scientific, San Francisco, CA, USA, v. 7, n. 04, p. 669–688, 1993. Citado na página 24.

COHAN, A. et al. A discourse-aware attention model for abstractive summarization of long documents. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. [S.l.: s.n.], 2018. p. 615–621. Citado 2 vezes nas páginas 47 e 52.

DANILEVSKY, M. et al. A survey of the state of explainable ai for natural language processing. In: *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*. [S.l.: s.n.], 2020. p. 447–459. Citado na página 33.

DEVLIN, J. et al. Bert: Pre-training of deep bidirectional transformers for language understanding. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, 2019. p. 4171–4186. Citado 2 vezes nas páginas 24 e 33.

DOŠILOVIĆ, F. K.; BRČIĆ, M.; HLUPIĆ, N. Explainable artificial intelligence: A survey. In: IEEE. *2018 41st International convention on information and communication technology, electronics and microelectronics (MIPRO)*. [S.l.], 2018. p. 0210–0215. Citado 2 vezes nas páginas 16 e 26.

EL-KASSAS, W. S. et al. Automatic text summarization: A comprehensive survey. *Expert Systems with Applications*, Elsevier, v. 165, p. 113679, 2020. Citado 3 vezes nas páginas 19, 20 e 21.

FERREIRA, R. et al. Assessing sentence scoring techniques for extractive text summarization. *Expert systems with applications*, Elsevier, v. 40, n. 14, p. 5755–5764, 2013. Citado na página 46.

FRIEDMAN, J. H. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, JSTOR, p. 1189–1232, 2001. Citado na página 39.

FRIEDMAN, J. H. Stochastic gradient boosting. *Computational statistics & data analysis*, Elsevier, v. 38, n. 4, p. 367–378, 2002. Citado na página 39.

GHAEINI, R.; FERN, X.; TADEPALLI, P. Interpreting recurrent and attention-based neural models: a case study on natural language inference. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. [S.l.: s.n.], 2018. p. 4952–4957. Citado na página 33.

GHODRATNAMA, S. et al. Extractive document summarization based on dynamic feature space mapping. *IEEE Access*, IEEE, v. 8, p. 139084–139095, 2020. Citado 3 vezes nas páginas 34, 43 e 51.

GOODMAN, B.; FLAXMAN, S. European union regulations on algorithmic decision-making and a "right to explanation". *AI magazine*, v. 38, n. 3, p. 50–57, 2017. Citado 2 vezes nas páginas 16 e 28.

GUNNING, D. Explainable artificial intelligence (xai). *Defense Advanced Research Projects Agency (DARPA), nd Web*, v. 2, n. 2, 2017. Citado na página 29.

GUPTA, S.; GUPTA, S. Abstractive summarization: An overview of the state of the art. *Expert Systems with Applications*, Elsevier, v. 121, p. 49–65, 2019. Citado na página 20.

HASTIE, T.; TIBSHIRANI, R. Generalized additive models: some applications. *Journal of the American Statistical Association*, Taylor & Francis, v. 82, n. 398, p. 371–386, 1987. Citado 4 vezes nas páginas 30, 34, 36 e 37.

HERMANN, K. M. et al. Teaching machines to read and comprehend. *Advances in neural information processing systems*, v. 28, p. 1693–1701, 2015. Citado 2 vezes nas páginas 26 e 47.

HERN, A. *Facebook translates 'good morning' into 'attack them', leading to arrest*. 2017. Acessado em 11 de Dez. de 2020. Disponível em: <https://www.theguardian.com/technology/2017/oct/24/facebook-palestine-israel-translates-good-morning-attack-them-arrest>. Citado na página 16.

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. *Neural computation*, MIT Press, v. 9, n. 8, p. 1735–1780, 1997. Citado na página 24.

JAIN, S.; WALLACE, B. C. Attention is not explanation. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. [S.l.: s.n.], 2019. p. 3543–3556. Citado na página 33.

JOACHIMS, T. *A Probabilistic Analysis of the Rocchio Algorithm with TFIDF for Text Categorization*. [S.l.], 1996. Citado na página 21.

JOSHI, M.; WANG, H.; MCCLEAN, S. Dense semantic graph and its application in single document summarisation. In: *Emerging Ideas on Information Filtering and Retrieval*. [S.l.]: Springer, 2018. p. 55–67. Citado na página 21.

KEDZIE, C.; MCKEOWN, K.; III, H. D. Content selection in deep learning models of summarization. In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. [S.l.: s.n.], 2018. p. 1818–1828. Citado 3 vezes nas páginas 25, 44 e 48.

KIM, B. et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In: PMLR. *International conference on machine learning*. [S.l.], 2018. p. 2668–2677. Citado na página 28.

LANDAUER, T. K.; FOLTZ, P. W.; LAHAM, D. An introduction to latent semantic analysis. *Discourse processes*, Taylor & Francis, v. 25, n. 2-3, p. 259–284, 1998. Citado na página 21.

LAPUSCHKIN, S. et al. Analyzing classifiers: Fisher vectors and deep neural networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. [S.l.: s.n.], 2016. p. 2912–2920. Citado na página 27.

LAPUSCHKIN, S. et al. Unmasking clever hans predictors and assessing what machines really learn. *Nature communications*, Nature Publishing Group, v. 10, n. 1, p. 1–8, 2019. Citado 2 vezes nas páginas 30 e 32.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *nature*, Nature Publishing Group, v. 521, n. 7553, p. 436–444, 2015. Citado 3 vezes nas páginas 23, 24 e 26.

LIBBRECHT, M. W.; NOBLE, W. S. Machine learning applications in genetics and genomics. *Nature Reviews Genetics*, Nature Publishing Group, v. 16, n. 6, p. 321–332, 2015. Citado na página 28.

LIN, C.-Y. Rouge: A package for automatic evaluation of summaries. In: *Text summarization branches out*. [S.l.: s.n.], 2004. p. 74–81. Citado 2 vezes nas páginas 24 e 49.

LIU, Y. Fine-tune bert for extractive summarization. *arXiv preprint arXiv:1903.10318*, 2019. Citado 3 vezes nas páginas 24, 25 e 48.

LOU, Y.; CARUANA, R.; GEHRKE, J. Intelligible models for classification and regression. In: *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. [S.l.: s.n.], 2012. p. 150–158. Citado 5 vezes nas páginas 34, 35, 36, 38 e 39.

LOU, Y. et al. Accurate intelligible models with pairwise interactions. In: *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*. [S.l.: s.n.], 2013. p. 623–631. Citado 5 vezes nas páginas 35, 38, 39, 40 e 41.

LUHN, H. P. The automatic creation of literature abstracts. *IBM Journal of research and development*, Ibm, v. 2, n. 2, p. 159–165, 1958. Citado 2 vezes nas páginas 19 e 20.

MAYBURY, M. T. Generating summaries from event data. *Information Processing & Management*, Elsevier, v. 31, n. 5, p. 735–751, 1995. Citado na página 19.

MIKOLOV, T. et al. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013. Citado na página 45.

MONTANI, I. et al. spaCy: industrial-strength natural language processing in Python. Zenodo, 2021. Disponível em: <https://doi.org/10.5281/zenodo.5648257>. Citado na página 45.

MORATANCH, N.; CHITRAKALA, S. A survey on extractive text summarization. In: IEEE. *2017 international conference on computer, communication and signal processing (ICCCSP)*. [S.l.], 2017. p. 1–6. Citado na página 19.

MUTLU, B.; SEZER, E. A.; AKCAYOL, M. A. Multi-document extractive text summarization: A comparative assessment on features. *Knowledge-Based Systems*, Elsevier, v. 183, p. 104848, 2019. Citado 3 vezes nas páginas 45, 46 e 47.

NALLAPATI, R.; ZHAI, F.; ZHOU, B. Summarunner: a recurrent neural network based sequence model for extractive summarization of documents. In: *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*. [S.l.]: AAAI Press, 2017. (AAAI'17), p. 3075–3081. Citado 8 vezes nas páginas 20, 24, 25, 33, 34, 44, 48 e 51.

NALLAPATI, R. et al. Abstractive text summarization using sequence-to-sequence rnns and beyond. In: *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*. Berlin, Germany: Association for Computational Linguistics, 2016. p. 280–290. Citado na página 47.

NARAYAN, S.; COHEN, S. B.; LAPATA, M. Ranking sentences for extractive summarization with reinforcement learning. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. [S.l.: s.n.], 2018. p. 1747–1759. Citado na página 24.

NENKOVA, A.; MCKEOWN, K. *Automatic summarization*. [S.l.]: Now Publishers Inc, 2011. Citado na página 20.

NENKOVA, A.; MCKEOWN, K. A survey of text summarization techniques. In: *Mining text data*. [S.l.]: Springer, 2012. p. 43–76. Citado na página 20.

NETO, J. L. et al. Document clustering and text summarization. Citeseer, 2000. Citado na página 21.

NORI, H. et al. Interpretml: A unified framework for machine learning interpretability. *arXiv preprint arXiv:1909.09223*, 2019. Citado 3 vezes nas páginas 35, 38 e 49.

OLIVEIRA, H. et al. Assessing shallow sentence scoring techniques and combinations for single and multi-document summarization. *Expert Systems with Applications*, Elsevier, v. 65, p. 68–86, 2016. Citado 2 vezes nas páginas 45 e 46.

OZSOY, M. G.; ALPASLAN, F. N.; CICEKLI, I. Text summarization using latent semantic analysis. *Journal of Information Science*, Sage Publications Sage UK: London, England, v. 37, n. 4, p. 405–417, 2011. Citado 2 vezes nas páginas 21 e 22.

QUINLAN, J. R. Induction of decision trees. *Machine learning*, Springer, v. 1, n. 1, p. 81–106, 1986. Citado na página 38.

RAMANATHAN, K. et al. Document summarization using wikipedia. In: SPRINGER. *Proceedings of the first international conference on intelligent human computer interaction*. [S.l.], 2009. p. 254–260. Citado na página 22.

RIBEIRO, M. T.; SINGH, S.; GUESTRIN, C. "why should i trust you?" explaining the predictions of any classifier. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. [S.l.: s.n.], 2016. p. 1135–1144. Citado 2 vezes nas páginas 31 e 32.

ROSENBLATT, F. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, American Psychological Association, v. 65, n. 6, p. 386, 1958. Citado na página 30.

SAMEK, W.; MÜLLER, K.-R. Towards explainable artificial intelligence. In: *Explainable AI: interpreting, explaining and visualizing deep learning*. [S.l.]: Springer, 2019. p. 5–22. Citado 6 vezes nas páginas 16, 26, 27, 28, 29 e 31.

SANKARASUBRAMANIAM, Y.; RAMANATHAN, K.; GHOSH, S. Text summarization using wikipedia. *Information Processing & Management*, Elsevier, v. 50, n. 3, p. 443–461, 2014. Citado 2 vezes nas páginas 22 e 23.

SARKHEL, R. et al. Interpretable multi-headed attention for abstractive summarization at controllable lengths. In: *Proceedings of the 28th International Conference on Computational Linguistics*. [S.l.: s.n.], 2020. p. 6871–6882. Citado 3 vezes nas páginas 16, 34 e 43.

SEE, A.; LIU, P. J.; MANNING, C. D. Get to the point: Summarization with pointer-generator networks. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. [S.l.: s.n.], 2017. p. 1073–1083. Citado 2 vezes nas páginas 47 e 51.

SERRANO, S.; SMITH, N. A. Is attention interpretable? In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. [S.l.: s.n.], 2019. p. 2931–2951. Citado na página 33.

SIMONYAN, K.; VEDALDI, A.; ZISSERMAN, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. Iclr, 2014. Citado na página 29.

STROBELT, H. et al. S eq 2s eq-v is: A visual debugging tool for sequence-to-sequence models. *IEEE transactions on visualization and computer graphics*, IEEE, v. 25, n. 1, p. 353–363, 2018. Citado na página 33.

SUTSKEVER, I.; VINYALS, O.; LE, Q. V. Sequence to sequence learning with neural networks. In: *Advances in neural information processing systems*. [S.l.: s.n.], 2014. p. 3104–3112. Citado na página 33.

TAS, O.; KIYANI, F. A survey automatic text summarization. *PressAcademia Procedia*, v. 5, n. 1, p. 205–213, 2007. Citado 3 vezes nas páginas 19, 20 e 21.

VASWANI, A. et al. Attention is all you need. In: *Proceedings of the 31st International Conference on Neural Information Processing Systems*. Red Hook, NY, USA: Curran Associates Inc., 2017. p. 6000–6010. Citado na página 24.

WONG, K.-F.; WU, M.; LI, W. Extractive summarization using supervised and semi-supervised learning. In: *Proceedings of the 22nd international conference on computational linguistics (Coling 2008)*. [S.l.: s.n.], 2008. p. 985–992. Citado 3 vezes nas páginas 22, 23 e 43.

WU, L. et al. R-drop: regularized dropout for neural networks. *Advances in Neural Information Processing Systems*, v. 34, 2021. Citado 2 vezes nas páginas 33 e 51.

WU, T.-F.; LIN, C.-J.; WENG, R. C. Probability estimates for multi-class classification by pairwise coupling. *Journal of Machine Learning Research*, v. 5, n. Aug, p. 975–1005, 2004. Citado na página 23.

XIAO, W.; CARENINI, G. Extractive summarization of long documents by combining global and local context. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. [S.l.: s.n.], 2019. p. 3011–3021. Citado 7 vezes nas páginas 25, 44, 47, 48, 50, 51 e 52.

XIAO, W.; CARENINI, G. Systematically exploring redundancy reduction in summarizing long documents. In: *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*. [S.l.: s.n.], 2020. p. 516–528. Citado 2 vezes nas páginas 51 e 52.

YANG, Z.; ZHANG, A.; SUDJIANTO, A. Gami-net: An explainable neural network based on generalized additive models with structured interactions. *Pattern Recognition*, v. 120, p. 108192, 2021. Citado 5 vezes nas páginas 40, 41, 42, 49 e 53.

ZEILER, M. D.; FERGUS, R. Visualizing and understanding convolutional networks. In: SPRINGER. *European conference on computer vision*. [S.l.], 2014. p. 818–833. Citado na página 31.

ZHONG, M. et al. Extractive summarization as text matching. *arXiv preprint arXiv:2004.08795*, 2020. Citado 4 vezes nas páginas 24, 25, 50 e 51.

ZHU, J. et al. Explainable ai for designers: A human-centered perspective on mixed-initiative co-creation. In: IEEE. *2018 IEEE Conference on Computational Intelligence and Games (CIG)*. [S.l.], 2018. p. 1–8. Citado 3 vezes nas páginas 15, 27 e 34.