



UNIVERSIDADE FEDERAL DA FRONTEIRA SUL

CAMPUS DE CHAPECÓ

CURSO DE CIÊNCIAS DA COMPUTAÇÃO

ANDREW MALTA SILVA

UMA ANÁLISE EXPLORATÓRIA USANDO MODELAGEM DE TÓPICO

ACOMPANHANDO A EVOLUÇÃO E A LEALDADE DOS USUÁRIOS DO STACK OVERFLOW

JUROS

CHAPECÓ

2021

ANDREW MALTA SILVA

UMA ANÁLISE EXPLORATÓRIA USANDO MODELAGEM DE TÓPICO

ACOMPANHANDO A EVOLUÇÃO E A LEALDADE DOS USUÁRIOS DO STACK OVERFLOW

JUROS

Trabalho final de graduação apresentado como requisito para
obtenção do título de Bacharel em Ciência da Computação
pela Universidade Federal da Fronteira Sul.
Orientador: Denio Duarte

CHAPECÓ

2021

Silva, André Malta

Uma análise exploratória usando modelagem de tópicos: Acompanhamento da evolução e fidelidade dos interesses dos usuários do Stack Overflow / Andrew Malta Silva. – 2021. 67 pp.: il.

Orientador: Dênio Duarte.

Trabalho de Conclusão de Curso – Universidade Federal da Fronteira Sul, curso de Ciência da Computação, Chapecó, SC, 2021.

1. Estouro de pilha. 2. Modelagem de tópicos. 3. Interesses dos usuários. 4. Evolução do tópico. 5. Lealdade no tópico. I. Duarte, Denio, orientador. II. Universidade Federal da Fronteira Sul. III. Título.

ANDREW MALTA SILVA

UMA ANÁLISE EXPLORATÓRIA USANDO MODELAGEM DE TÓPICO

ACOMPANHANDO A EVOLUÇÃO E A LEALDADE DOS USUÁRIOS DO STACK OVERFLOW

JUROS

Trabalho final de graduação apresentado como requisito para obtenção do título de Bacharel em Ciência da Computação pela Universidade Federal da Fronteira Sul.

Orientador: Denio Duarte

Este trabalho final de graduação foi defendido e aprovado pela banca examinadora em:
19/05/2021.

COMISSÃO EXAME



Dênio Duarte – UFFS



Fernando Bevilacqua – UFFS



Guilherme dal Bianco – UFFS

AGRADECIMENTOS

Como este trabalho marca o fim de uma etapa da minha vida e o início de outra, há várias pessoas a agradecer. Em primeiro lugar, sou muito grato aos meus pais, Margarete e Vanderlei Silva, por me darem as condições e a educação necessárias para focar nos meus estudos.

Sem eles, tudo seria mais difícil e agradeço-lhes por todo o apoio.

Agradeço também aos meus professores e membros da banca examinadora, Fernando Bevilacqua e Guilherme dal Bianco. Suas percepções e conselhos claros e diretos me permitiram compreender melhor algumas facetas deste trabalho, tornando-o melhor.

Claro, agradeço também ao meu professor e orientador, Denio Duarte, que me apresentou a esta área de estudo e me ajudou neste processo de pesquisa. Além disso, ele me ensinou muitas coisas, desde conceitos sobre modelagem de tópicos até metodologia científica. Nossas reuniões semanais de aconselhamento foram fundamentais para que eu mantivesse o progresso e a motivação.

E por fim, mas não menos importante, agradeço a Estela Vilas Boas, minha linda e adorável namorada, que esteve lá para me apoiar o tempo todo. Ela me motivou quando eu estava desanimada e comemorou comigo quando as coisas deram certo. Não importa o desafio que eu precisava enfrentar, ela estava lá para me mostrar o caminho. Ela é a senhora da minha vida.

Por causa dessas pessoas mencionadas, estou finalmente fechando esta etapa e começando a voar voos mais altos. Você nunca será esquecido.

“Torture os dados e eles confessarão qualquer coisa.”

Ronald Coase

ABSTRATO

A web apresenta diversas plataformas de compartilhamento de conhecimento entre os usuários, como newsletters, blogs, redes sociais e comunidades. Entre eles, o Stack Overflow é uma comunidade popular de perguntas e respostas (Q&A) que permite aos usuários compartilhar e adquirir conhecimento sobre tópicos de programação de computadores. Se exploradas, as postagens do Stack Overflow apresentam informações que podem fornecer muitos insights sobre o conhecimento compartilhado. Alguns trabalhos exploraram essas postagens e geraram informações relevantes, mas as bases de dados que analisaram estão desatualizadas agora. Além disso, muitos deles não consideraram a autoria das postagens e as datas de publicação em suas análises, o que pode fornecer insights úteis temporais e centrados no usuário. Portanto, este trabalho fez vários experimentos para inferir tópicos do Stack Overflow e os analisou empregando métricas propostas para medir a popularidade relativa dos tópicos e sua deriva. Em seguida, essas métricas foram aplicadas para analisar a popularidade dos tópicos em informações temporais e de autoria, acompanhando a evolução da popularidade e a deriva desses tópicos globalmente e para cada usuário individual. A metodologia empregou modelagem de tópicos, processamento de linguagem natural, técnicas estatísticas e experimentos de validação para alcançar esses resultados promissores, que foram disponibilizados gratuitamente na web. Por fim, a metodologia abordada mostrou-se eficaz nas análises realizadas, que inferiram com sucesso os tópicos do Stack Overflow e acompanharam sua evolução de popularidade geral e centrada no usuário.

Palavras-chave: Stack Overflow. Modelagem de tópicos. Interesses dos usuários. Evolução do tópico. Lealdade do tópico.

LISTA DE FIGURAS

Figura 1 – Pergunta aleatória do Stack Overflow.	23
Figura 2 – Resposta aceita da questão da Figura 1	23
Figura 3 – Exemplo de aplicação de tokenização	26
Figura 4 – Exemplo de aplicação de remoção de stopword	26
Figura 5 – Exemplo de aplicação de lematização	27
Figura 6 – Exemplo de aplicação de lematização	27
Figura 7 – Exemplo de localização de bigramas	28
Figura 8 – Ilustração das intuições LDA	31
Figura 9 – Exemplo de distribuição por tópicos	32
Figura 10 – Modelo gráfico probabilístico LDA	35
Figura 11 – Perplexidade em função do número de tópicos no conjunto de dados	40
NIPS Figura 12 – Histograma de desenvolvedores por tópico plotado por Wand, Lo e Jiang (2013) .	41
Figura 13 – Metodologia aplicada por Barua, Thomas e Hassan (2014) em um fluxograma .	41
Figura 14 – As 5 principais tendências crescentes e decrescentes no Stack Overflow por Barua, Thomas, e Hassan (2014).	43
Figura 15 – Visão geral da metodologia abordada em fluxograma.	45
Figura 16 – Conteúdo de post aleatório submetido à subetapa de limpeza.	47
Figura 17 – Conteúdo do post aleatório submetido à subetapa de enriquecimento.	47
Figura 18 – Gráfico 3D do Stack Overflow $\gamma\gamma\gamma\gamma$ coerência por número de tópicos $\gamma\gamma$ e iterações $\gamma\gamma$	49
Figura 19 – Tópicos de tendências gerais no Stack Overflow.	53
Figura 20 – Evolução da popularidade do tópico geral por mês no Stack Overflow.	54
Figura 21 – Desvio geral de popularidade do tópico no Stack Overflow.	55
Figura 22 – Tópicos de tendências para o usuário 1.289.716	56
Figura 23 – Evolução da popularidade do tópico para o usuário 1.289.716	57
Figura 24 – Desvio de popularidade do tópico para o usuário 1.289.716	58

LISTA DE MESAS

Tabela 1 – Esquema simplificado dos Posts do Stack	21
Overflow Tabela 2 – Trend topics do Stack Overflow de Barua, Thomas e Hassan (2014) ⁴³	
Tabela 3 – Definição do corpus	48

CONTEÚDO

1	INTRODUÇÃO	15
1.1	ORGANIZAÇÃO DO DOCUMENTO	18
2	OVERFLOW DA PILHA.	21
2.1	ESQUEMA.	21
2.2	DISCUSSÃO	24
3	PROCESSAMENTO DE LINGUAGEM NATURAL	25
3.1	TOKENIZAÇÃO.	25
3.2	REMOÇÃO DE STOPWORD.	26
3.3	LEMATIZAÇÃO.	26
3.3.1	Derivação	27
3.4	N-GRAMAS.	28
3,5	DISCUSSÃO	28
4	MODELAGEM DO TÓPICO.	31
4.1	MODELO DE TÓPICO.	32
4.1.1	Processo generativo	33
4.1.2	Processo de inferência	33
4.1.3	Processo de rotulagem	33
4.2	ALOCAÇÃO DE DIRICHLET LATENTE	34
4.3	MÉTRICAS	36
4.3.1	Janelas deslizantes e de contexto	36
4.3.2	PMI e NPMI	37
4.3.3	yyy métrica de coerência	37
4.4	DISCUSSÃO	38
5	TRABALHO RELATADO	39
5.1	MODELOS DE TÓPICOS.	39
5.2	ANÁLISES EXPLORATÓRIAS	40
5.3	DISCUSSÃO	44
6	EXPERIMENTOS.	45
6.1	EXTRAÇÃO	45
6.2	PRÉ-PROCESSANDO	46
6.3	MODELAGEM DO TÓPICO.	48
6.3.1	Edifício Corpus	48
6.3.2	Inferência LDA	49
6.3.3	Rotulagem	50
6.4	DISCUSSÃO	50
7	RESULTADOS	51
7.1	MÉTRICAS	51

7.2	ANÁLISES GERAIS.	52
7.3	ANÁLISES CENTRADAS NO USUÁRIO	54
7.4	DISCUSSÃO	56
8	CONCLUSÕES	59
	REFERÊNCIAS	61
	APÊNDICE A – STACK OVERFLOW TÓPICOS E PALAVRAS PRINCIPAIS .	65
	APÊNDICE B - TÓPICOS DE EXCESSO DE PILHA E TÓPICOS DE TENDÊNCIA	
	MEDIDAS DE POPULARIDADE	67

1. INTRODUÇÃO

A Web é uma enorme fonte de informação e conhecimento para os mais diversos propósitos.

Qualquer pessoa pode acessar blogs, comunidades, newsletters, redes sociais, repositórios científicos, serviços de streaming e muitos outros sites. As pessoas podem se divertir, aprender e interagir com diversos conteúdos e pessoas ao redor do mundo.

Quanto mais o tempo passa, mais as pessoas se envolvem em interações na web e mais informações estão disponíveis.

Além disso, às vezes as pessoas usam a Web para buscar soluções para questões específicas que estão lidando. Existem muitas maneiras de buscar essas soluções, mas as comunidades de perguntas e respostas (Q&A) têm exatamente o objetivo de resolver os problemas dos usuários. As comunidades de perguntas e respostas permitem que os usuários peçam ajuda com seus problemas, criando perguntas que podem ser respondidas por outros usuários. Assim, quanto mais usuários perguntarem e responderem, mais conhecimento a comunidade de perguntas e respostas apresentará.

Algumas comunidades de perguntas e respostas apresentam postagens (termo geral para perguntas e respostas) sobre uma ampla variedade de tópicos. Por exemplo, Quora¹ e Yahoo Answers² permitem que os usuários contribuam em praticamente qualquer tópico. Diferentemente, há comunidades de perguntas e respostas que optam por delimitar os temas que abordam. Consequentemente, eles diminuem a variedade de tópicos, mas o conhecimento compartilhado torna-se mais especializado.

Seguindo o conceito de comunidades de perguntas e respostas delimitadas por tópicos, o Stack Overflow está focado em apresentar postagens sobre tópicos de programação de computadores. Com mais de 50 milhões de posts, o Stack Overflow é uma plataforma onde desenvolvedores de software profissionais e entusiastas podem compartilhar e adquirir conhecimento de forma concisa neste campo. Eles são capazes de buscar perguntas resolvidas, criar novas com base em seus próprios problemas e responder a outras perguntas.

O Stack Overflow permite que os usuários avaliem a qualidade das postagens, criando uma espécie de controle de qualidade do conhecimento compartilhado. Postagens com pontuações altas fornecem reputação e selos para os usuários que as criaram, motivando os usuários a fazer contribuições significativas (CAVUSOGLU; LI; HUANG, 2015). Além disso, os usuários também são motivados a editar outras postagens para corrigir erros de digitação, adicionar mais informações e melhorar a clareza do inglês. Esses recursos fazem com que os usuários melhorem continuamente a qualidade das postagens do Stack Overflow.

Como o Stack Overflow apresenta muitas informações úteis, diversos trabalhos utilizam as postagens para realizar análises exploratórias, identificar padrões e obter insights. Alguns deles empregam técnicas de modelagem de tópicos para descobrir os tópicos discutidos nas postagens, o que é útil para diversos tipos de análises. Por exemplo, Barua, Thomas e Hassan (2014) descobriram 40 tópicos de postagens do Stack Overflow de agosto de 2008 a agosto de 2010, analisando-os e classificando-os de acordo com sua popularidade relativa.

As postagens também apresentam outras informações analisáveis, como autores e datas de criação. Analisá-los pode fornecer resultados relevantes para fins centrados no usuário, como muitos tipos de sistemas de recomendação. Por exemplo, Wand, Lo e Jiang (2013) investigaram o comportamento dos usuários nos tópicos sobre os quais escrevem, identificando sua preferência como questionadores e respondentes.

¹ <https://www.quora.com/> <https://>

² answers.yahoo.com/

Além disso, Barua, Thomas e Hassan (2014) analisaram como os tópicos do Stack Overflow evoluem e como eles são afetados pelos interesses dos usuários.

Como os dados do Stack Overflow são uma fonte rica de informações de texto, eles podem ser explorados empregando essas técnicas de modelagem de tópicos. Portanto, este trabalho tem como objetivo fazer uma análise exploratória a partir de perguntas e respostas do Stack Overflow para investigar algumas facetas. No entanto, este tipo de estudo tem alguns desafios a enfrentar. Como as perguntas e respostas do Stack Overflow são criadas pelo usuário, seus tópicos são inevitavelmente baseados nos interesses dos usuários. Consequentemente, quando os usuários criam postagens relacionadas ao que estão usando, estudando ou trabalhando, eles fazem com que os tópicos discutidos do Stack Overflow evoluam ao longo do tempo. Portanto, as postagens são misturas de tópicos que representam indiretamente os usuários interesses.

Naturalmente, os usuários têm interesses diferentes que mudam dependendo de muitas circunstâncias, como iniciar um novo emprego, aprender uma nova tecnologia ou encontrar novos hobbies. Alguns usuários são leais a alguns tópicos nos quais estão muito interessados, enquanto outros desviam seus interesses constantemente. Identificar como os interesses dos usuários evoluem pode fornecer algumas informações sobre sua fidelidade aos tópicos de seu interesse.

No entanto, fazer essas análises é uma tarefa desafiadora devido a muitos fatores. A base de dados do Stack Overflow apresenta um grande volume de perguntas, respostas e muito mais informações, o que dificulta a organização e análise dos posts. Qualquer algoritmo que execute esta tarefa precisa ser construído otimizando a memória e o uso do disco.

Além disso, como os posts do Stack Overflow são documentos criados pelo usuário, analisá-los exige atenção especial. Várias informações semânticas, sintáticas e morfológicas estão escondidas entre as palavras, que é composta de linguagem natural (BIRD; KLEIN; LOPER, 2009). O ser humano pode ler facilmente a linguagem natural, sendo capaz de compreender significados, interpretar contextos e identificar tópicos discutidos. No entanto, fazer isso em larga escala a partir de uma coleção de milhares ou milhões de documentos é uma tarefa impraticável para humanos.

Construir algoritmos capazes de fazer análises em larga escala de forma otimizada é uma solução para este problema. No entanto, processar a linguagem natural presente em uma coleção de documentos é uma tarefa desafiadora para um algoritmo. Felizmente, ao invés de interpretar documentos apenas como arrays de caracteres, técnicas de processamento de linguagem natural podem ser empregadas para analisar documentos considerando gramática, morfologia e muitos outros fatores (BIRD; KLEIN; LOPER, 2009).

Embora a modelagem de tópicos apresente muitas técnicas para descobrir os tópicos de uma coleção, a maioria deles são definidos por métodos estatísticos e probabilísticos com hiperparâmetros que afetam diretamente os resultados, como o número de tópicos (BLEI; NG; JORDAN, 2003; BLEI, 2012; STEYVERS; GRIFFITHS, 2007). Portanto, empregar métricas para medir sua eficácia e interpretabilidade é importante para ter resultados satisfatórios nesse processo (FITELSON, 2003; WALLACH et al., 2009; CHANG et al., 2009; ALETRAS; STEVENSON, 2013; RÖDER; BOTH; HINNEBURG, 2015).

Entre outras, essas técnicas foram aplicadas pelos trabalhos citados (WANG;

LO; JIAN, 2013; BARUA; THOMAS; HASSAN, 2014), mas a disponibilidade de resultados é outro fator a ser destacado. Embora esses trabalhos tenham explorado postagens do Stack Overflow e obtido conclusões, seus resultados não estão disponíveis para serem explorados por outras pessoas. Por exemplo, outros pesquisadores podem analisar aspectos adicionais dos resultados, enquanto alguns recrutadores de empregos podem estar interessados em analisar os tópicos preferidos dos usuários para encontrar candidatos.

Portanto, a principal contribuição deste trabalho é fornecer uma análise exploratória a partir de perguntas e respostas do Stack Overflow, que visa acompanhar a evolução e a fidelidade dos interesses dos usuários nos tópicos discutidos no Stack Overflow. Assim, as contribuições específicas são definidas a seguir.

1. Definir o melhor número de tópicos para resumir os posts do Stack Overflow;
2. Descobrir e rotular os tópicos discutidos no Stack Overflow;
3. Propor uma métrica para medir a popularidade de um tópico em um conjunto de postagens;
4. Propor uma métrica para medir o desvio de um usuário em um tópico;
5. Identificando os tópicos de tendência no Stack Overflow analisando a popularidade geral do tópicos em todas as postagens;
6. Acompanhar a evolução da popularidade dos tópicos em todas as postagens;
7. Identificar os trend topics de cada usuário analisando a popularidade que apresenta nos tópicos de seu interesse;
8. Acompanhar a evolução da popularidade que cada usuário apresenta nos tópicos de seu interesse;
9. Analisar a fidelidade que cada usuário apresenta nos temas de seu interesse;
10. Tornar todos os resultados acessíveis na web para serem analisados por outros pesquisadores.

A modelagem de tópicos é uma técnica de aprendizado não supervisionado e, portanto, ajustar os hiperparâmetros é essencial para obter bons modelos (WALLACH et al., 2009; BLEI, 2012). A quantidade de tópicos é um fator determinante para a qualidade dos resultados; inferir alguns tópicos os torna gerais e de granulação mais grosseira, mas inferir muitos tópicos os torna detalhados e de granulação mais fina (BARUA; THOMAS; HASSAN, 2014).

Definir o melhor número de tópicos é essencial para garantir a qualidade dos tópicos descobertos. Após a realização de vários experimentos empregando a técnica LDA para atingir esses objetivos, a melhor distribuição por tópicos foi identificada e rotulada. Os resultados apresentaram 30 tópicos bem definidos, o que foi bastante promissor.

Além disso, como analisar a popularidade dos tópicos é uma parte importante deste trabalho, foi proposta a métrica Topic Popularity para realizar essa medição, necessária para atingir outros objetivos. Essa métrica proposta foi aplicada em todas as postagens do Stack Overflow,

fornecendo a popularidade geral do tópico de todos os tópicos e dando uma visão abrangente de seus comportamentos e tendências.

A evolução da popularidade dos tópicos foi rastreada calculando a métrica proposta para cada fatia de tempo mensal no histórico do Stack Overflow. Como analisar padrões de valores crescentes e decrescentes é um estudo frequente em aprendizado de máquina (BISHOP, 2006; PEDREGOSA et al., 2011), este estudo forneceu muitas informações para entender como os tópicos do Stack Overflow evoluem.

No entanto, como mencionado anteriormente, os interesses dos usuários têm muito impacto nas postagens do Stack Overflow. Assim, assumindo que as postagens apresentam esses interesses ocultos entre as palavras, analisar os tópicos relacionados às postagens criadas por um usuário pode definir os interesses que esse usuário possui. Portanto, a popularidade dos tópicos foi calculada separadamente para cada usuário, incluindo os tópicos de tendência e sua evolução de popularidade. Esses dados centrados no usuário permitem análises relevantes individuais para cada usuário, fornecendo informações perspicazes para análise.

Seguindo uma linha diferente, os interesses dos usuários evoluem de forma diferente para cada tópico em que estão interessados. Compreender como os usuários se mantêm fiéis ou se afastam de seus tópicos de interesse pode ser útil para muitos propósitos. Por exemplo, recrutadores de emprego que procuram desenvolvedores com talentos específicos podem analisar a fidelidade dos usuários do Stack Overflow em tópicos relacionados à experiência necessária. Portanto, com base no desvio padrão da estatística, a métrica Topic Popularity Drift foi proposta para medir a deriva da evolução da popularidade de um tópico.

A métrica Topic Popularity Drift foi aplicada globalmente e para cada usuário, gerando diversas medidas para análise. Embora essa métrica consiga calcular a variação de um conjunto de medidas de popularidade de um tópico e forneça resultados interessantes, ela se mostrou pouco útil para analisar a fidelidade em um tópico. Portanto, os requisitos para os traços de lealdade do tópico devem ser melhor analisados para outra proposta de métrica no futuro.

Por fim, como a análise desses dados seria inútil se os resultados não chegarem às pessoas que precisam deles, eles foram disponibilizados na web no site Internet Archive³. Assim, outros pesquisadores podem tomar os resultados gerados para validá-los e explorá-los com outras abordagens e para diversos fins.

1.1 ORGANIZAÇÃO DO DOCUMENTO

Como entender os dados é essencial para processá-los adequadamente, o capítulo 2 apresenta mais detalhes sobre o Stack Overflow, focando na apresentação dos atributos que compõem as perguntas e respostas. Em seguida, o capítulo 3 apresenta algumas técnicas de Processamento de Linguagem Natural, que visam interpretar e analisar textos.

A seguir, o capítulo 4 apresenta brevemente os principais conceitos sobre modelagem de tópicos e algumas métricas. Além disso, como o entendimento de trabalhos com objetivos semelhantes é essencial para orientar a metodologia abordada neste trabalho, o capítulo 5 apresenta alguns trabalhos relacionados divididos em abordagens de modelos de tópicos e análises exploratórias a partir do Stack Overflow.

³ <https://archive.org/details/staty>

Considerando todo o conteúdo apresentado, o capítulo 6 detalha a metodologia abordada neste trabalho, apresentando as técnicas, etapas e experimentos empregados para atingir os objetivos. Em seguida, o capítulo 7 apresenta os resultados obtidos com gráficos e tabelas, fornecendo informações suficientes para tirar algumas conclusões no capítulo 8.

2 EXCESSO DE PILHA

Lançado em 2008, o Stack Exchange¹ é uma rede de comunidades de perguntas e respostas, cada uma cobrindo um conjunto diferente de tópicos discutidos. O maior deles é o Stack Overflow², que apresenta perguntas e respostas abrangendo uma ampla gama de tópicos em programação de computadores desde 2008. O usuário base é composta por programadores profissionais e entusiastas que buscam soluções para os problemas que enfrentam e dar soluções a outros.

Com mais de 13 milhões de usuários (registrados e não registrados) e mais de 50 milhões posts, o Stack Overflow é a maior comunidade online para desenvolvedores de software. Permitindo eles para aprender, compartilhar conhecimento e construir carreiras em uma plataforma dinâmica, o Stack Overflow se tornou um grande fonte de conhecimento nesta área.

O conhecimento que o Stack Overflow fornece está contido nas postagens, ou seja, perguntas e respostas criadas. Os responsáveis por criar e avaliar essas postagens são os usuários, que ajudam entre si para encontrar, melhorar e avaliar soluções. Portanto, como postagens e usuários do Stack Overflow foram destacados como os principais objetos de estudo deste trabalho, as seções a seguir apresentar seu esquema e discutir sua importância.

2.1 ESQUEMA

O Stack Overflow armazena dados em um banco de dados relacional composto por algumas tabelas e relacionamentos. Postagens é a tabela que armazena todas as postagens, incluindo perguntas e respostas. Há 23 atributos nesta tabela, cada um com suas próprias finalidades. No entanto, apenas 8 deles apresentam informações significativas de acordo com o que este trabalho pretende fazer, que são apresentadas na tabela 1.

Tabela 1 – Esquema simplificado dos Postes Stack Overflow

Atributo	Modelo
<small>Id</small>	int
PostTypeId	minúsculo
ParentId	int
CreationDate	datetime
Corpo	nvarchar(max)
ProprietárioUserId	int
Título	nvarchar(250)
Fonte	nvarchar(250)

de tags – Stack Overflow Data Explorer (<https://bit.ly/3ep7Tdm>)

Além de perguntas e respostas, existem outros seis tipos de postagens no Stack Overflow. Embora sejam a minoria, filtrá-los é essencial para manter apenas as postagens criadas pelos usuários.

¹ <https://stackexchange.com/>

² <https://stackoverflow.com/>

O atributo `PostTypeId` armazena o tipo de postagem, onde 1 e 2 representam perguntas e respostas, respectivamente. Portanto, identificá-los e separá-los dos demais posts é uma tarefa fácil.

Sempre que um usuário cria uma postagem, esta postagem armazena automaticamente o identificador do autor no atributo `OwnerId`. Embora outros usuários possam editar qualquer postagem para corrigir erros gramaticais, erros de digitação ou fornecer mais informações, esse atributo armazena consistentemente apenas o autor da postagem. Portanto, considerando os objetivos deste trabalho, o atributo `OwnerId` é útil para identificar a autoria das postagens, que é representada por identificadores numéricos.

Outra informação também armazenada automaticamente sempre que um post é criado é a data de criação. Essas informações são essenciais para manter as postagens em ordem cronológica, possibilitando a análise das postagens de acordo com suas informações temporais. Por exemplo, o Stack Overflow apresenta pesquisas anuais³ analisando os dados gerados naquele ano e fornecendo alguns insights interessantes. Portanto, o atributo `CreationDate` é essencial para este trabalho.

Quando os usuários olham para uma pergunta, a primeira coisa que veem é o título da pergunta, que é uma frase curta que resume o assunto da pergunta. Títulos bem escritos fazem com que os usuários entendam rapidamente os tópicos abordados por uma pergunta, aumentando a probabilidade de serem respondidos. Portanto, o atributo `Título` pode ser útil para descobrir tópicos de perguntas, embora as respostas não têm títulos.

Além dos títulos, as perguntas também possuem tags que ajudam a identificar seus assuntos e melhorar o resultados do motor de busca. Cada tag é definida por uma palavra ou um pequeno conjunto de palavras divididas por hífens que definem brevemente um tópico abordado na pergunta. No entanto, embora as tags forneçam muitas dicas sobre os tópicos das perguntas, elas são definidas pelo usuário e podem apresentar algumas inconsistências (BARUA; THOMAS; HASSAN, 2014).

Por exemplo, um usuário pode criar uma pergunta atribuindo a tag *iphone-api* e outro usuário, atribuindo *iphone-sdk*. Mesmo que as questões sejam sobre temas semelhantes e apresentem contextos semelhantes, suas tags atribuídas são escritas de forma diferente e, portanto, uma busca baseada em apenas uma dessas tags resultará em apenas uma dessas questões (BARUA; THOMAS; HASSAN, 2014). Portanto, o atributo `Tags` não pode definir tópicos de perguntas por si só, mas ainda fornecer dicas úteis para isso.

Embora apenas as perguntas tenham título e tags, cada postagem possui um identificador exclusivo armazenado no atributo `Id`. A relação entre perguntas e respostas ocorre armazenando o identificador da pergunta em suas respectivas respostas. Portanto, cada resposta é associada a uma pergunta armazenando o ID da pergunta no atributo `ParentId`, facilitando a localização de uma pergunta com um
responda.

Por fim, enquanto os títulos as tags fornecem apenas breves ideias e dicas sobre os tópicos de um post, o atributo `Body` apresenta o conteúdo completo e detalhado. Neste atributo, o autor escreve tudo sobre o problema enfrentado ou solução proposta. Como o corpo é armazenado como HTML renderizado, uma postagem pode apresentar frases, parágrafos, estruturas de tópicos, imagens, referências, URLs e trechos de código. Esses recursos tornam as postagens suficientemente detalhadas e compreensíveis para outros programadores que querem contribuir e aprender com eles.

³ <https://insights.stackoverflow.com/survey/2019>

Figura 1 – Pergunta aleatória do Stack Overflow

Title: How to run Tensorflow on CPU

Asked 4 years ago Active today Viewed 134k times

Score: 122

Body: I have installed the GPU version of tensorflow on an Ubuntu 14.04. I am on a GPU server where tensorflow can access the available GPUs. I want to run tensorflow on the CPUs. Normally I can use `env CUDA_VISIBLE_DEVICES=0` to run on GPU no. 0. How can I pick between the CPUs instead? I am not interested in rewriting my code with `with tf.device("/cpu:0"):`

Tags: python tensorflow

share improve this question follow

edited Feb 7 '18 at 13:45 dsalaj 1,343 2 14 31

CreationDate: asked Jun 6 '16 at 14:41

Owner: Alexander R. Johansen 1,866 3 13 20

add a comment

Fonte – Site do Stack Overflow (<https://bit.ly/3d45hQG>)

A Figura 1 apresenta um instantâneo de uma pergunta aleatória escolhida no site do Stack Overflow, onde os atributos Title, Tags, CreationDate, Body e Score são destacados em retângulos azuis. Em vez de exibir o OwnerId como um número, o retângulo Owner na figura apresenta algumas informações sobre o proprietário: seu nome (Alexander R. Johansen), reputação (1.866) e distintivos (3 de ouro, 13 de prata e 20 de bronze). Por fim, o atributo Body tem seis frases e dois pequenos trechos de código que fornecem mais detalhes sobre o problema.

Figura 2 – Resposta aceita da pergunta da Figura 1

7 Answers

Body: You can apply `device_count` parameter per `tf.Session`:

```
config = tf.ConfigProto(
    device_count = {'GPU': 0}
)
sess = tf.Session(config=config)
```

See also protobuf config file:

[tensorflow/core/framework/config.proto](https://www.tensorflow.org/api_guides/python/config_protos)

Score: 112

share improve this answer follow

CreationDate: answered Jun 6 '16 at 15:11

Owner: Ivan Aksamentov - Drop 11.8k 3 27 59

Active Oldest Votes

Fonte – Site do Stack Overflow (<https://bit.ly/3d45hQG>)

A Figura 2 apresenta um instantâneo da resposta aceita da pergunta na figura 1.

Da mesma forma que na figura 1, os atributos CreationDate, Body, Score e as informações do proprietário são marcados em retângulos azuis. Observe que há um trecho de código maior apresentando a solução de código para o problema enfrentado.

2.2 DISCUSSÃO

Este capítulo apresentou brevemente algumas informações do Stack Overflow, com foco nos usuários, perguntas e respostas. Cada postagem do Stack Overflow é composta por vários atributos, mas especialmente 8 deles foram apresentados devido a sua importância para os objetivos deste trabalho. Portanto, entendê-los adequadamente é o primeiro passo para identificar o que é preciso analisá-los.

Além disso, o site Internet Archive⁴ fornece um despejo de dados completo de todo o conteúdo de contribuição do usuário na rede Stack Exchange, que é gerado e atualizado pelo Stack Exchange Community⁵. Portanto, as informações apresentadas neste capítulo podem ser baixadas como um arquivo XML⁶.

No entanto, as postagens do Stack Overflow são compostas de textos em inglês escritos livremente, o que significa que as postagens precisam ser processadas para serem compreensíveis por máquina. Portanto, o próximo capítulo apresenta brevemente algumas técnicas de Processamento de Linguagem Natural capazes de analisar postagens de várias formas, ajudando a processá-las algoritmicamente.

⁴ <https://archive.org> [https://](https://archive.org)

⁵ archive.org/details/stackexchange <https://archive.org/>

⁶ [download/stackexchange/stackoverflow.com-Posts.7z](https://archive.org/download/stackexchange/stackoverflow.com-Posts.7z)

3 PROCESSAMENTO DE LINGUAGEM NATURAL

A comunicação humana é especialmente baseada em linguagens naturais, permitindo que eles falem, escrevam e leiam diariamente. As linguagens naturais apresentam muitos conjuntos complexos e variáveis de regras que as definem e organizam, como gramática, sintaxe e morfologia. No entanto, diferentemente das linguagens artificiais, como linguagens de programação e notações matemáticas, as linguagens naturais evoluem naturalmente à medida que são passadas de geração em geração (BIRD; KLEIN; LOPER, 2009).

No entanto, as línguas naturais costumam apresentar conjuntos de regras que as definem e organizam, como gramática, sintaxe, morfologia e ideogramas (BIRD; KLEIN; LOPER, 2009). Essas regras acompanham a evolução da linguagem, adaptando-se ao longo do tempo para melhor representar a linguagem. Portanto, entender essas regras permite aprender uma linguagem natural.

No entanto, fazer um algoritmo capaz de compreender linguagens naturais é uma tarefa desafiadora. Um algoritmo geralmente interpreta frases apenas como matrizes de caracteres sem sentido. Por isso, o Processamento de Linguagem Natural (ou PNL para abreviar) vem com esse propósito: fornecer técnicas capazes de realizar qualquer tipo de manipulação computacional de linguagem natural. A PNL apresenta técnicas com muitos usos e para muitos propósitos, como reconhecimento de fala, compreensão de linguagem natural e geração de linguagem natural (BIRD; KLEIN; LOPER, 2009).

Como a manipulação algorítmica de textos sempre foi uma tarefa desafiadora, a PNL ganha cada vez mais relevância e aplicação tanto na indústria quanto na academia. Campos como interação humano-computador, análise de informações de negócios, inteligência artificial e modelagem de tópicos empregam a PNL como parte essencial de seus processos (BIRD; KLEIN; LOPER, 2009).

Portanto, como os posts do Stack Overflow são documentos escritos livremente em inglês, a PNL é necessária para manipulá-los adequadamente. Entre outras, técnicas como tokenização, remoção de palavras irrelevantes, lematização/lematização e *ŷŷ*-gramas são úteis para limpar, enriquecer e entender uma coleção de documentos, cada um explicado e discutido nas seções a seguir.

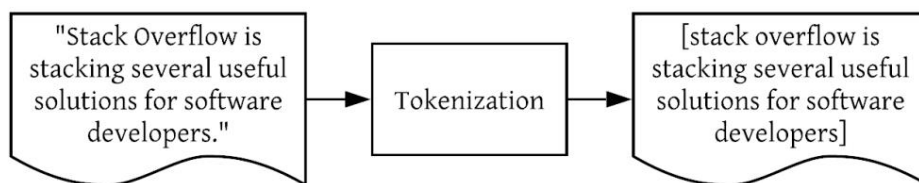
3.1 TOKENIZAÇÃO

Ao processar um texto com PNL, é essencial identificar os tokens que o compõem, que são unidades básicas de texto que representam palavras, dígitos e sinais de pontuação (MANNING; SCHÜTZE, 1999). Esse processo, denominado tokenização, divide os documentos em listas de tokens, permitindo identificar e analisar separadamente cada unidade básica de texto.

Dependendo do caso, cada token apresenta importância diferente. Os sinais de pontuação podem ser úteis na identificação da estrutura do texto e das formas gramaticais (MANNING; SCHÜTZE, 1999), como orações subordinadas ou formas afirmativas e interrogativas. Diferentemente, os sinais de pontuação podem ser ignorados quando apenas o significado das palavras importa.

A Figura 3 apresenta um exemplo de aplicação de tokenização a uma sentença usando a biblioteca *gensim* para Python (ŷEHŷŷEK; SOJKA, 2010). Observe que os sinais de pontuação foram ignorados e todos

Figura 3 – Exemplo de aplicação de tokenização



letras foram convertidas em minúsculas para evitar diferenciação entre maiúsculas e minúsculas (*por exemplo*, "o", "O" e "O").

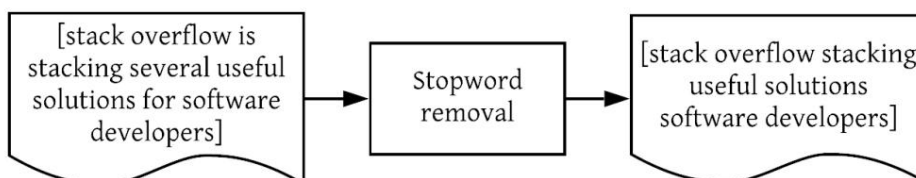
Portanto, qualquer processamento adicional pode analisar facilmente cada token separadamente.

3.2 REMOÇÃO DE STOPWORD

Todas as palavras apresentam importância sintática e semântica, contribuindo para melhorar a coerência e a coesão. No entanto, quando se busca identificar a semântica das palavras empregando a correspondência palavra por palavra, existem algumas palavras que raramente contribuem com informações úteis (MANNING; SCHÜTZE, 1999).

Essas palavras são chamadas de stopwords e, como "[...] geralmente possuem pouco conteúdo lexical, e sua presença em um texto não o distingue de outros textos" (BIRD; KLEIN; LOPER, 2009, p. 60), filtro reduz o tempo de processamento adicional. Por exemplo, palavras como "o", "é", "isto" e "um" são consideradas palavras irrelevantes.

Figura 4 – Exemplo de aplicação de remoção de stopwords



A Figura 4 apresenta um exemplo de remoção de stopwords da sentença já tokenizada na Figura 3, também utilizando a biblioteca gensim (YEHÿÿEK; SOJKA, 2010). Observe que o número de palavras foi reduzido de doze para sete, reduzindo o tempo de processamento posterior. Mesmo com menos palavras, ainda é possível entender o sujeito da frase.

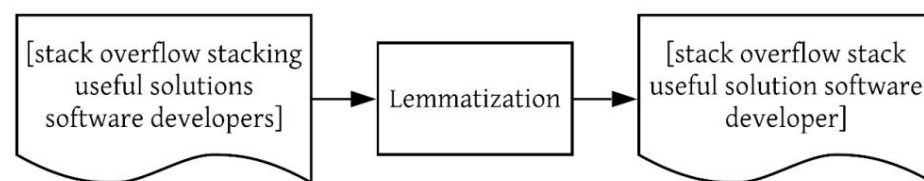
3.3 LEMATIZAÇÃO

A morfologia é uma área da linguística que estuda a formação e classificação de palavras (MANNING; SCHÜTZE, 1999). As palavras têm várias formas e inflexões que alteram parte de seus significados e as diferenciam umas das outras. No entanto, toda palavra possui um lema, ou seja, uma forma canônica, de dicionário ou de citação que resume o significado principal dessa palavra (BIRD; KLEIN; LOPER, 2009).

Por exemplo, as palavras “go”, “goes”, “gone”, “going” e “went” têm o mesmo lema: “go”. A palavra “ir” consegue resumir o significado de suas formas flexionadas. Portanto, encontrar lemas de palavras permite agrupar palavras por seus significados, reduzindo o número de palavras únicas e associando suas semânticas.

Além disso, a morfologia agrupa as palavras com comportamento sintático semelhante em classes, que são comumente chamadas de parte do discurso (MANNING; SCHÜTZE, 1999). Substantivos, verbos, adjetivos e advérbios são parte do discurso inglês comum, cada um apresentando diferentes comportamentos de flexão. Por exemplo, os verbos podem flexionar para indicar tempos diferentes, enquanto os substantivos podem flexionar para indicar formas singulares e plurais. Portanto, identificar a parte do discurso de uma palavra é essencial para entender esses comportamentos de flexão e, conseqüentemente, encontrar lemas.

Figura 5 – Exemplo de aplicação de lematização

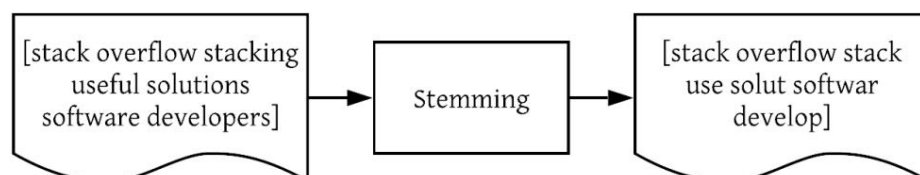


A Figura 5 apresenta um exemplo de aplicação do WordNet Lemmatizer para a frase limpa na Figura 3 e Figura 4 usando o Natural Language Tool Kit (ou NLTK para abreviar) para Python (BIRD; KLEIN; LOPER, 2009). Observe que todas as palavras flexionadas foram reduzidas com precisão aos seus respectivos lemas, pois o processo utiliza o reconhecimento de parte da fala.

3.3.1 Derivação

Quando o desempenho é um problema a enfrentar, as técnicas de stemming são bons substitutos para lematização. Stemming é um processo heurístico bruto que funciona removendo sufixos e prefixos anexados das palavras, letra por letra (JIVANI et al., 2011). Diferentemente da lematização, as técnicas de stemming não dependem da identificação adequada de parte do discurso e da associação de significados das palavras, o que lhes confere um melhor desempenho (JIVANI et al., 2011). No entanto, as técnicas de stemming encontram radicais (não lemas), dando resultados que nem sempre são palavras reais.

Figura 6 – Exemplo de aplicação de stemming



A Figura 6 apresenta um exemplo de aplicação de radicalização à sentença pré-processada nas figuras 3 e 4, empregando também NLTK (BIRD; KLEIN; LOPER, 2009). Observe que todas as palavras flexionadas foram reduzidas, mas não com precisão porque “útil”, “soluções” e “software”

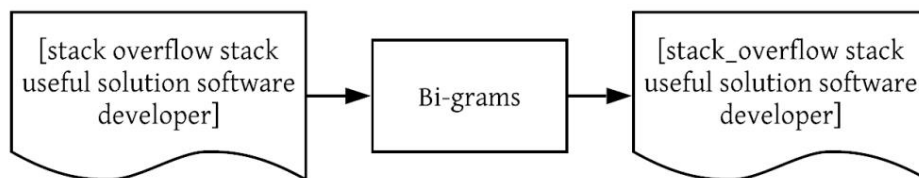
foram reduzidos a “use”, “solut” e “softwar”, respectivamente. Nesse caso, o significado correto deles foi parcialmente perdido, tornando-os difíceis de entender.

3.4 N-GRAMAS

Os possíveis significados de uma frase podem mudar de acordo com a sintaxe empregada, onde a ordem das palavras é um fator relevante. As palavras têm significados próprios, mas existem termos compostos por várias palavras que podem representar significados diferentes dos originais. Por exemplo, a palavra “novo” tem muitos significados relacionados a algo que não existia antes, mas “nova york” está relacionado a uma cidade ou estado dos EUA. Portanto, identificar esses termos multi-palavras é essencial para capturar seus reais significados.

A técnica de n-gramas consiste em aplicar métodos probabilísticos para entender padrões na ordem das palavras (MIKOLOV et al., 2013). Destina-se a encontrar sequências de palavras n que aparecem com frequência na coleção, onde sequências de duas palavras são chamadas de bigramas, sequências de três palavras são chamadas de trigramas e assim por diante.

Figura 7 – Exemplo de localização de bigramas



A Figura 7 apresenta um exemplo de localização de bigramas a partir da sentença lematizada na Figura 5. Apenas um bigrama pode ser observado neste exemplo: “stack_overflow”. A técnica aplicada verificou que as palavras “pilha” e “estouro” aparecem muitas vezes juntas de acordo com outros documentos, unindo as palavras com sublinhado. Portanto, o algoritmo entende “stack_overflow” como um conceito diferente de ocorrências separadas de “stack” e “overflow” palavras.

3.5 DISCUSSÃO

Este capítulo apresentou e discutiu algumas técnicas de PNL que normalmente são aplicadas em uma etapa de pré-processamento para limpar documentos e manter apenas informações úteis. A tokenização identifica todos os tokens, enquanto a remoção de palavras irrelevantes reduz o tamanho da coleção e evita o processamento de palavras sem sentido. Além disso, a lematização reduz todas as palavras aos seus respectivos lemas, associando os significados das palavras relacionadas. Finalmente, calcular n-gramas pode ajudar a identificar termos com várias palavras, melhorando as representações semânticas. A aplicação dessas técnicas prepara a coleção para qualquer processo subsequente.

A escolha entre lematização e lematização precisa estar de acordo com o campo de aplicação. “Quando há uma necessidade urgente na eficiência do agrupamento de documentos em inglês, Porter

lematizador é a melhor escolha, quando há necessidade de precisão ou na geração de rótulos de cluster, a lematização pode ser uma boa escolha” (HAN et al., 2012, p. 119). Portanto, este trabalho opta por empregar a lematização e focar na precisão e qualidade dos resultados, mesmo que seja mais demorado.

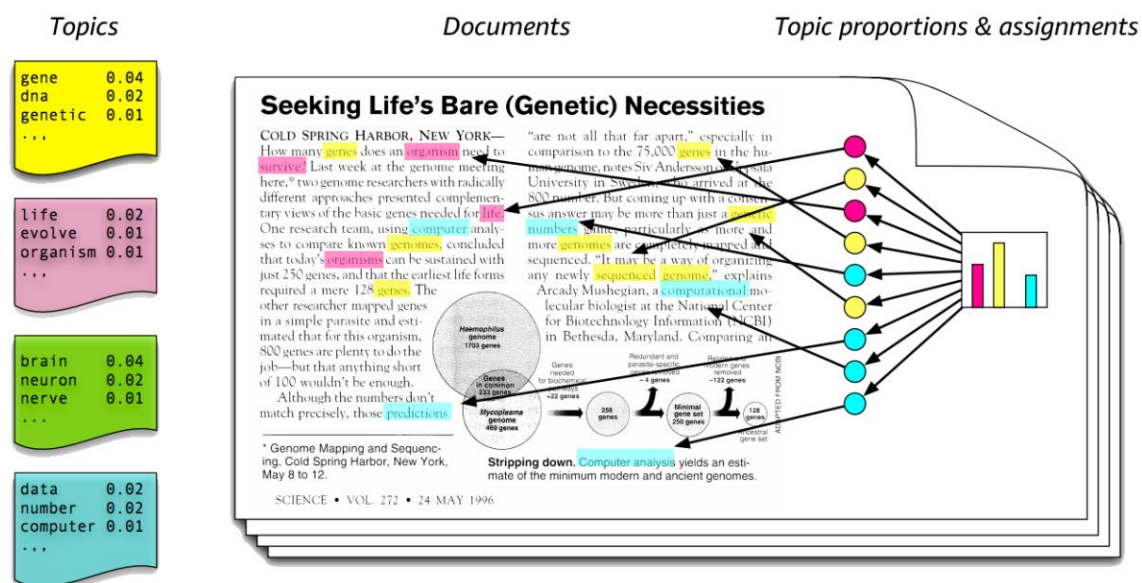
Considerando que muitos objetivos deste trabalho abordam o rastreamento dos tópicos presentes nas postagens criadas por usuários do Stack Overflow, entender como descobrir esses tópicos é essencial. Portanto, o próximo capítulo se concentra em apresentar brevemente alguns conceitos principais sobre modelagem de tópicos probabilísticos e métricas úteis.

4 MODELAGEM DE TÓPICOS

A Web é uma fonte crescente de documentos que podem ser encontrados na forma de notícias, blogs, artigos, livros, redes sociais, comunidades de perguntas e respostas, entre outros. Esses documentos são compostos por conteúdo criado pelo usuário com base em linguagens naturais, que fornecem contextos e informações para a cognição humana (STEYVERS; GRIFFITHS, 2007).

Ao ler um documento, um humano pode identificar alguns tópicos que o resumem. A intuição por trás desse processo é que os documentos são misturas de tópicos, cada tópico definido por um conjunto de palavras que parecem semanticamente relacionadas e frequentemente aparecem juntas (BLEI, 2012). Palavras comumente associadas a um tópico específico podem ser anotadas no documento, facilitando sua identificação. Naturalmente, algumas dessas palavras podem aparecer com mais frequência do que outras, mesmo que estejam relacionadas ao mesmo tema (BLEI, 2012).

Figura 8 – Ilustração das intuições LDA



Fonte – (BLEI, 2012)

Por exemplo, a Figura 8 apresenta um artigo intitulado "Seeking Life's Bare (Genetic) Necessities", que trata da aplicação de análise de dados para determinar o número de genes que um organismo precisa para sobreviver à evolução da espécie. Observação este artigo tem algumas palavras destacadas, onde cada cor representa um tópico diferente. Palavras destacadas em azul, como "computador" e "predição", são sobre análise de dados; palavras destacadas em rosa, como "vida" e "organismo", são sobre biologia evolutiva; e palavras destacadas em amarelo, como "genes" e "sequenciado", são sobre genética.

Os tópicos presentes no documento da Figura 8 são detalhados à esquerda, cada um com suas 3 principais palavras. Observe que cada palavra é seguida de um número real que representa seu peso, ou seja, a probabilidade da presença dessa palavra em seu tópico (BLEI, 2012). Além disso, a figura mostra

um histograma à direita, que apresenta uma breve visualização das proporções de cada tópico associado ao documento.

No entanto, como os dados de texto são encontrados em grandes coleções com milhares ou mesmo milhões de documentos não estruturados e não rotulados, ler uma coleção inteira e identificar seus tópicos pode ser impraticável se feito por humanos. Portanto, para fazer isso de forma eficiente, a modelagem de tópicos fornece “um conjunto de algoritmos que visam descobrir e anotar grandes arquivos de documentos com informações temáticas” (BLEI, 2012, p. 1).

4.1 MODELO DE TÓPICO

Para generalizar esse conceito, a modelagem de tópicos define um modelo de tópicos como uma distribuição sobre tópicos, onde cada tópico é uma distribuição de probabilidade sobre palavras (BLEI, 2012). Em outras palavras, um tópico é definido por um conjunto de palavras ponderadas, onde cada peso é a probabilidade de sua respectiva palavra ser encontrada em documentos (como na Figura 8). As palavras em um modelo de tópico são definidas por um vocabulário fixo e a soma das probabilidades de palavras em um tópico deve ser 1 (STEYVERS; GRIFFITHS, 2007).

Figura 9 – Exemplo de distribuição por tópicos

Topic 247	Topic 5	Topic 43	Topic 56
word	word	word	word
DRUGS .069	RED .202	MIND .081	DOCTOR .074
DRUG .060	BLUE .099	THOUGHT .066	DR. .063
MEDICINE .027	GREEN .096	REMEMBER .064	PATIENT .061
EFFECTS .026	YELLOW .073	MEMORY .037	HOSPITAL .049
BODY .023	WHITE .048	THINKING .030	CARE .046
MEDICINES .019	COLOR .048	PROFESSOR .028	MEDICAL .042
PAIN .016	BRIGHT .030	FELT .025	NURSE .031
PERSON .016	COLORS .029	REMEMBERED .022	PATIENTS .029
MARIJUANA .014	ORANGE .027	THOUGHTS .020	DOCTORS .028
LABEL .012	BROWN .027	FORGOTTEN .020	HEALTH .025
ALCOHOL .012	PINK .017	MOMENT .020	MEDICINE .017
DANGEROUS .011	LOOK .017	THINK .019	NURSING .017
ABUSE .009	BLACK .016	THING .016	DENTAL .015
EFFECT .009	PURPLE .015	WONDER .014	NURSES .013
KNOWN .008	CROSS .011	FORGET .012	PHYSICIAN .012
PILLS .008	COLORED .009	RECALL .012	HOSPITALS .011

Fonte – (STEYVERS; GRIFFITHS, 2007)

Por exemplo, a Figura 9 apresenta quatro tópicos obtidos da coleção TASA (LAN DAUER; FOLTZ; LAHAM, 1998), que possui mais de 37.000 passagens de texto de materiais educacionais (STEYVERS; GRIFFITHS, 2007). Observe que os tópicos são identificados por números e mostram apenas suas respectivas 16 principais palavras. Cada palavra é seguida de sua probabilidade, onde, por exemplo, a palavra “vermelho” tem uma probabilidade de 20,2% de ser encontrada em documentos associados ao tópico 5.

4.1.1 Processo generativo

Dado um modelo de tópicos, é possível utilizar um processo generativo para selecionar palavras de sua distribuição sobre tópicos e criar novos documentos (BLEI, 2012; STEYVERS; GRIFFITHS, 2007). Esse processo gerativo pode ser generalizado por um procedimento probabilístico simples que opera no processo de duas etapas abaixo:

- a) Escolher aleatoriamente um conjunto de um ou mais tópicos;
- b) Para cada palavra no documento:
 - Escolha aleatoriamente um tópico entre os tópicos escolhidos;
 - Escolha aleatoriamente uma palavra desse tópico.

Por exemplo, se estiver criando um documento de 100 palavras a partir de alguns tópicos da Figura 9, as etapas a seguir são: (a) escolher os novos tópicos do documento (por exemplo, tópicos 247 e 5) e (b) selecionar palavras desses tópicos com base em as probabilidades das palavras até que o tamanho do documento seja 100 (*por exemplo*, “drogas vermelhas brilhantes efeitos de cor vermelha perigosa...”). Observe que “drogas” e “vermelho” tendem a ser as palavras mais frequentes no novo documento devido às suas probabilidades.

4.1.2 Processo de inferência

Em vez de gerar novos documentos de acordo com um determinado modelo de tópicos, alguns trabalhos precisam para descobrir automaticamente a distribuição sobre tópicos de uma determinada coleção. Nesse caso, a modelagem de tópicos também apresenta técnicas que utilizam uma determinada coleção de documentos para fazer o processo inverso: inferir o modelo de tópicos que provavelmente gerou a coleção (BLEI, 2012; STEYVERS; GRIFFITHS, 2007).

Essa inferência visa descobrir os padrões ocultos em documentos e palavras para construir um modelo de tópicos (BLEI, 2012). Existem várias técnicas de inferência, que geralmente funcionam fazendo uso de métodos probabilísticos e estatísticos (BLEI, 2012). Como identificar a estrutura de tópicos de uma grande coleção pode fornecer muitos insights para a cognição humana (STEYVERS; GRIFFITHS, 2007), esse tipo de inferência é abordado em vários trabalhos.

4.1.3 Processo de rotulagem

A inferência de modelagem de tópicos geralmente resulta em tópicos identificados por números que diferem entitativos, mas desprovidos de significado útil sobre seu conteúdo. Por exemplo, os tópicos apresentados nas Figuras 8 e 9 não possuem rótulos, mas cores e números que os identificam. Existem alguns estudos abordando a rotulagem automática de tópicos (ALLAHYARI; KOCHUT, 2015), mas é possível rotular tópicos analisando manualmente suas distribuições sobre as palavras.

Por exemplo, os quatro tópicos apresentados na Figura 9 podem ser rotulados como “uso de drogas” (tópico 247), “cores” (tópico 5), “memória e mente” (tópico 43) e “consultas médicas” (tópico

56) (STEYVERS; GRIFFITHS, 2007). Assim, os tópicos parecem mais compreensíveis para a cognição humana e dão uma melhor noção sobre o conteúdo apresentado nos documentos.

4.2 ALOCAÇÃO DE DIRICHLET LATENTE

Latent Dirichlet Allocation (LDA) é um modelo de tópico popular que é considerado o mais simples (BLEI, 2012). LDA é uma abordagem de aprendizado não supervisionado para descobrir e associar variáveis latentes escondidas em variáveis observadas, independente do que as variáveis representam (BLEI; NG; JORDAN, 2003). No caso de tópicos e documentos, as variáveis latentes são os tópicos e as variáveis observadas são os documentos (BLEI, 2012).

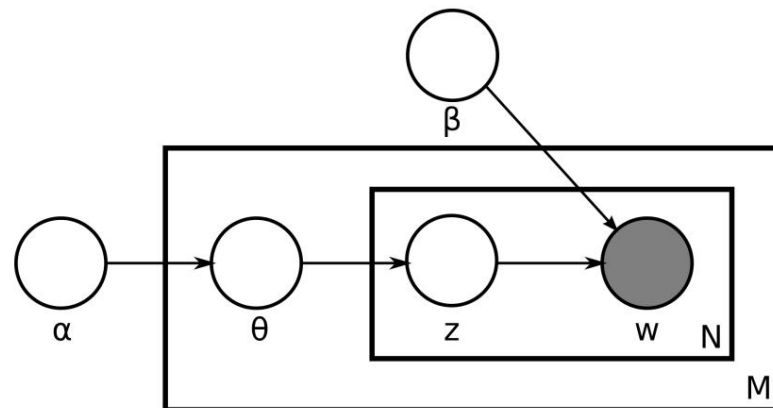
O LDA é descrito como um modelo probabilístico generativo que gera documentos usando a distribuição de Dirichlet para escolher aleatoriamente distribuições de tópicos por documento e distribuições de palavras por tópico (BLEI; NG; JORDAN, 2003). Como a distribuição de Dirichlet é uma família de distribuições de probabilidade contínuas e multivariadas que são parametrizadas por um vetor de números reais positivos (MINKA, 2000), suas propriedades facilitam a inferência LDA (BLEI; NG; JORDAN, 2003).

As definições e representações LDA são baseadas em notação formal e terminologia, que são apresentadas a seguir.

- \mathcal{D} é o corpus, *ou seja*, a coleção de documentos, onde d é um documento
- N é o número de documentos no corpus
- n_d é o número de palavras em um determinado documento, onde um documento d tem n_d palavras
- K é o número de tópicos
- θ_d é a distribuição de tópicos para o documento d
- $\phi_{k,v}$ é o tópico para a v -ésima palavra no documento d
- $w_{d,v}$ é uma palavra específica.
- α é o parâmetro do Dirichlet prior nas distribuições de tópicos por documento
- β é o parâmetro do Dirichlet prior na distribuição de palavras por tópico

A Figura 10 apresenta o modelo gráfico probabilístico LDA em notação de placa usando notação formal e terminologia LDA (BLEI; NG; JORDAN, 2003). As caixas representam iterações de elementos específicos, enquanto os círculos (ou placas) representam variáveis LDA. A caixa externa itera os documentos d e aplica o parâmetro α à distribuição de Dirichlet para calcular o θ_d para cada documento d . Em seguida, a caixa interna itera as n_d palavras de cada documento iterado d e aplica o parâmetro β à distribuição de Dirichlet para usar a atribuição de tópico θ_d à palavra $w_{d,v}$.

Figura 10 – Modelo gráfico probabilístico LDA



Fonte – (BLEI; NG; JORDÂNIA, 2003; BLEI, 2012)

Essas definições formais de LDA servem como um guia para definir como funciona o processo de inferência de LDA. Dirichlet é uma distribuição conveniente porque “[...] está na família exponencial, possui estatísticas suficientes de dimensão finita, e é conjugada à distribuição multinomial” (BLEI; NG; JORDAN, 2003, p. 996), o que facilita a processo de inferência (BLEI; NG; JORDÂNIA, 2003).

A inferência LDA analisa documentos para identificar padrões de ocorrência de palavras e usá-los para construir tópicos (BLEI, 2012). Embora essa inferência seja considerada intratável devido à sua complexidade, o LDA faz isso aplicando métodos de aproximação, como a aproximação de Laplace, a aproximação variacional de Bayes e a cadeia de Markov Monte Carlo (BLEI; NG; JORDAN, 2003).

Ao inferir a distribuição sobre tópicos que provavelmente geraram uma determinada coleção de documentos, o LDA requer um corpus onde cada documento está em um formato bag-of-words, ou seja, uma representação de texto que descreve a ocorrência de palavras dentro de um documento (BLEI; NG; JORDÂNIA, 2003). Essa representação é simples: cada documento é representado por uma lista com o número de vezes que cada palavra única aparece no documento. Assim, este formato assume que os documentos possuem um dicionário fixo definindo todas as palavras únicas que compõem a coleção (BLEI; NG; JORDAN, 2003), sendo necessário computar o dicionário antes de realizar a inferência.

Além disso, alguns trabalhos computam o termo frequência inversa do documento (TF IDF para abreviar), uma estatística numérica que calcula a importância de uma palavra para um documento em uma coleção (RAMOS et al., 2003). O TF-IDF é definido como o produto de duas estatísticas: frequência de termos e frequência de documentos inversa, que ponderam com sucesso a importância das palavras (RAMOS et al., 2003). Ao invés de apresentar o número de ocorrências para cada palavra (bag of-words), o TF-IDF calcula um peso que representa a importância de cada palavra em um documento (RAMOS et al., 2003).

Como os resultados do TF-IDF podem ser representados no formato bag-of-words, é possível usar diretamente o TF-IDF como entrada para inferência LDA. De fato, o TF-IDF é considerado o método de ponderação mais aplicado em sistemas de recomendação e trabalhos de modelagem de tópicos (BEEL et al.,

2016). Portanto, ao fazer uma inferência LDA, é necessário computar o dicionário e os sacos de palavras (ou TF-IDF) da coleção dada (BLEI; NG; JORDAN, 2003).

No entanto, a inferência não garante um resultado de alta qualidade. A aplicação de técnicas de TF-IDF e NLP pode melhorar o modelo inferido, mas para detectar essas melhorias é necessário aplicar métricas. Portanto, a próxima seção apresenta algumas métricas capazes de avaliar a qualidade de um modelo e ajudar a melhorá-lo.

4.3 MÉTRICAS

As técnicas de inferência de modelagem de tópicos não possuem conhecimento prévio sobre os tópicos da coleção antes de inferi-los e, portanto, são consideradas técnicas não supervisionadas (LAU; NEWMAN; BALDWIN, 2014). Por isso, avaliar a qualidade de um modelo inferido é uma tarefa desafiadora, pois não existem rótulos conhecidos para orientar uma análise de acurácia ou precisão (CHANG et al., 2009; LAU; NEWMAN; BALDWIN, 2014).

Nesse caso, é possível utilizar a avaliação humana analisando alguns aspectos dos resultados, mas realizá-la em larga escala pode se tornar caro e inviável, além de suscetível a potenciais erros humanos. Portanto, métricas de avaliação automatizadas podem ser muito úteis para esse fim (RÖDER; BOTH; HINNEBURG, 2015).

As métricas de coerência são consideradas relativamente próximas da percepção humana (CHANG et al., 2009; LAU; NOVO HOMEM; BALDWIN, 2014), o que os torna boas escolhas. Eles trabalham analisando a similaridade semântica e associatividade de palavras e tópicos de uma determinada coleção e seus tópicos inferidos (CHANG et al., 2009). Existem várias métricas de coerência e vários métodos em que se baseiam, alguns deles são apresentados a seguir.

4.3.1 Janelas deslizantes e de contexto

Existem muitas métricas de coerência que fazem uso de *janela deslizante* e *janela de contexto*. Assumindo que palavras consecutivas podem ter algum tipo de associação, tanto as técnicas de janelas deslizantes quanto as de janelas de contexto consistem em dividir um determinado conjunto de palavras em subconjuntos que podem fornecer informações úteis para o cálculo da métrica de coerência (RÖDER; BOTH; HINNEBURG, 2015).

Uma *janela deslizante* é um subconjunto de \tilde{y} palavras consecutivas que “deslizam” sobre um determinado conjunto de palavras, palavra por palavra (RÖDER; BOTH; HINNEBURG, 2015; CHANG et al., 2009). Por exemplo, se criar uma *janela deslizante* de tamanho 2 a partir do conjunto de palavras $\tilde{y} = \{\tilde{y}_1, \tilde{y}_2, \tilde{y}_3, \tilde{y}_4\}$, os resultados seriam $\tilde{y}\tilde{y} = \{\tilde{y}_1, \tilde{y}_2\} \tilde{y} \{\tilde{y}_2, \tilde{y}_3\} \tilde{y} \{\tilde{y}_3, \tilde{y}_4\}$.

Uma *janela de contexto* é um subconjunto de \tilde{y} palavras consecutivas antes e depois de uma determinada palavra de um conjunto de palavras (RÖDER; BOTH; HINNEBURG, 2015; CHANG et al., 2009). Por exemplo, se criar uma *janela de contexto* de tamanho 1 dada a palavra \tilde{y}_3 do conjunto de palavras $\tilde{y} = \{\tilde{y}_1, \tilde{y}_2, \tilde{y}_3, \tilde{y}_4\}$, os resultados seriam $\tilde{y}\tilde{y} = \{\tilde{y}_2, \tilde{y}_4\}$.

4.3.2 PMI e NPMI

Pointwise Mutual Information (PMI para abreviar) é um método capaz de medir a associatividade entre duas palavras (RÖDER; BOTH; HINNEBURG, 2015; ALETRAS; STEVENSON, 2013; CHANG et al., 2009). Alguns trabalhos revelam que métricas de coerência usando métodos baseados em PMI tendem a dar resultados relativamente semelhantes à classificação humana (RÖDER; BOTH; HINNEBURG, 2015; CHANG et al., 2009). Formalmente, o PMI é definido como a equação a seguir (RÖDER; BOTH; HINNEBURG, 2015):

$$PMI(w_1, w_2) = \log \frac{f(w_1, w_2)}{f(w_1)f(w_2)} \quad (4.1)$$

Assumindo que w_1 e w_2 são palavras dadas, $f(w_1, w_2)$ é a frequência em que ambas as palavras são encontradas na mesma janela (seja deslizante ou contexto), e $f(w_1)$ e $f(w_2)$ são a frequência em que cada palavra é encontrada separadamente (RÖDER; AMBOS; HINNEBURG, 2015; ALETRAS; STEVENSON, 2013).

Os resultados do PMI fornecem informações úteis sobre a semântica das palavras e existem vários métodos baseados no PMI. Normalized Pointwise Mutual Information (NPMI para abreviar) é uma variação que normaliza os resultados do PMI para intervalos $[-1, 1]$ (RÖDER; BOTH; HINNEBURG, 2015; ALETRAS; STEVENSON, 2013). O NPMI é formalmente definido como a equação a seguir (RÖDER; BOTH; HINNEBURG, 2015):

$$NPMI(w_1, w_2) = \frac{PMI(w_1, w_2)}{\max_{w'} PMI(w_1, w') \text{ ou } \max_{w'} PMI(w_2, w')} \quad (4.2)$$

Assumindo que w_1 e w_2 são palavras dadas, um resultado próximo a -1 representa nenhuma coocorrência das duas palavras dadas, enquanto um resultado próximo a 1 representa coocorrência completa. Um resultado 0 representa que as palavras dadas são independentes umas das outras. Essa propriedade do NPMI facilita a interpretabilidade dos resultados tanto por humanos quanto pelas métricas de coerência que aplica o NPMI (RÖDER; BOTH; HINNEBURG, 2015).

4.3.3 w_1 métrica de coerência

w_1 é uma métrica de coerência que, considerando uma determinada palavra de um determinado tópico, calcula a coocorrência dessa palavra com todas as palavras de seu tópico (RÖDER; BOTH; HINNEBURG, 2015). Esse cálculo aplica uma variação do NPMI sobre uma *janela deslizante* de tamanho 110 e resulta em um conjunto de vetores, um para cada palavra. Formalmente, w_1 é definido como a equação a seguir (SYED; SPRUIT, 2017; RÖDER; BOTH; HINNEBURG, 2015):

$$w_1(w_0) = \left(\frac{PMI(w_0, w_1)}{\max_{w'} PMI(w_0, w')} \right)_{w_1=1, \dots, |V|} \quad (4.3)$$

Experimentos mostram que $\gamma\gamma\gamma\gamma$ tem um desempenho melhor em comparação com outras métricas de coerência, como $\gamma\gamma\gamma\gamma\gamma\gamma\gamma\gamma$, $\gamma\gamma\gamma\gamma\gamma\gamma\gamma\gamma\gamma\gamma\gamma\gamma$, $\gamma\gamma\gamma\gamma$, $\gamma\gamma\gamma\gamma$ (ver (RÖDER; BOTH; HINNEBURG, 2015) para mais detalhes). Portanto, $\gamma\gamma\gamma\gamma$ é uma boa escolha ao verificar a representabilidade de uma determinada distribuição sobre tópicos de acordo com sua coleção original.

4.4 DISCUSSÃO

Este capítulo apresentou brevemente conceitos e técnicas gerais de modelagem de tópicos. Como descobrir tópicos de postagens do Stack Overflow é uma parte essencial deste trabalho, o LDA pode ser usado para isso. Portanto, também é necessário computar o dicionário, sacos de palavras e TF-IDF da coleção. A biblioteca *gensim* ¹ para Python oferece muitas classes e métodos para trabalhar com modelagem de tópicos, incluindo os citados anteriormente.

Além disso, a qualidade da inferência LDA depende da qualidade da coleta e dos resultados de outras técnicas aplicadas. Por exemplo, os métodos de PNL melhoram a qualidade da coleta em muitos aspectos, enquanto várias das técnicas mencionadas possuem hiperparâmetros que também afetam o resultado final. Portanto, a métrica de coerência $\gamma\gamma\gamma\gamma$ é essencial para ajudar a melhorar a representatividade dos tópicos.

Por fim, é necessário entender como outros trabalhos aplicam os conceitos e técnicas supracitados. Portanto, o próximo capítulo apresenta um conjunto de trabalhos apresentando objetivos relacionados a este trabalho em diferentes aspectos, que podem fundamentar a metodologia abordada.

¹ <https://radimrehurek.com/gensim/>

5 TRABALHOS RELACIONADOS

A identificação de tópicos latentes de uma determinada coleção de documentos pode fornecer informações perspicazes. Vários trabalhos têm aplicado modelagem de tópicos a diversos tipos de aplicações, como filtragem colaborativa, recuperação de informações, identificação de autoria, sistemas de recomendação e extração de opinião (YANG et al., 2019).

No entanto, muitas abordagens convencionais geralmente não consideram recursos extras ocultos nos metadados dos documentos, como autoria e data de criação/publicação (XU; SHI; QIAO; ZHU; JUNG, et al., 2014; XU; SHI; QIAO; ZHU; ZHANG, e outros, 2014). A análise desses recursos extras pode ajudar no desenvolvimento de aplicativos personalizados e centrados no usuário (YANG et al., 2019). Portanto, as seções a seguir apresentam algumas abordagens de modelos de tópicos e análises exploratórias que abordam esse tipo de estudo.

5.1 MODELOS DE TÓPICO

A análise da autoria em processos de modelagem de tópicos pode fornecer informações úteis sobre os interesses dos autores. Rosen-Zvi *et al.* (2004) abordam o problema da autoria propondo o modelo Author-Topic (AT), um modelo generativo no estilo LDA que inclui informações de autoria. O modelo AT funciona associando autores a distribuições multinomiais sobre tópicos que definem proporcionalmente os tópicos aos quais cada autor está relacionado.

No entanto, os interesses dos autores não são estáticos. O tempo é um fator significativo, pois os interesses dos autores podem derivar e evoluir ao longo do tempo. O modelo Topics over Time (TOT), proposto por Wang e McCallum (2006), é um modelo do estilo LDA capaz de acompanhar a evolução do tópico ao longo do tempo, semelhante ao Dynamic Topic Model (DTM) proposto por Blei e Lafferty (2006).

No entanto, embora ambos os modelos abordem o rastreamento da evolução do tópico, eles não consideram a autoria em formação.

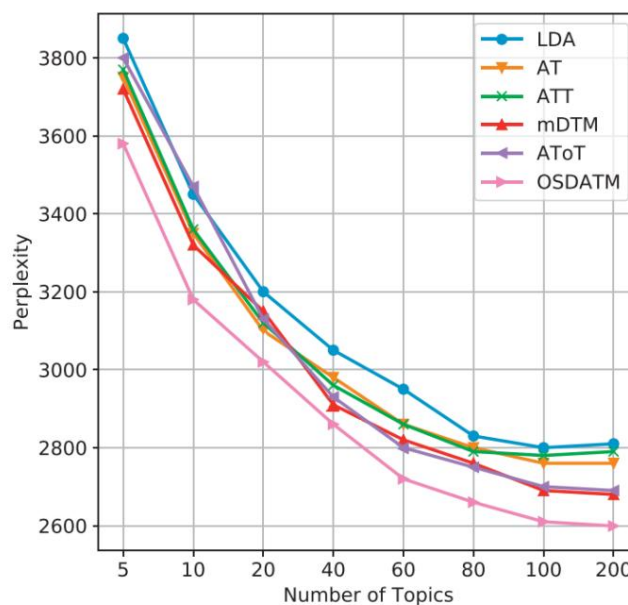
Existem alguns trabalhos que abordam o acompanhamento da evolução dos interesses dos autores. Tang e Zhang (2010) propõem o modelo Autor-Tempo-Tópico (ATT), que combina os modelos AT e TOT para estimar a distribuição de tópicos sobre palavras e timestamps e suas associações com autores. Além disso, o modelo Temporal-Author-Topic (TAT), proposto por Ali Daud (2012), faz a mesma combinação, mas utilizando timestamps anuais ao invés de timestamps dinâmicos.

Além disso, Xu *et al.* (2014) propõem o modelo Author-Topic over Time (AToT), que também combina os modelos AT e TOT, mas aplicando diferentes métodos de inferência. O modelo AToT também possui uma variação distribuída chamada Distributed Author-Topic over Time (D-AToT), que faz uso de múltiplos processadores e melhora o desempenho do modelo (XU; SHI; QIAO; ZHU; ZHANG, et al., 2014).

Recentemente, Yang *e col.* (2019) propõem o Ordering-sensitive and Semantic-aware Dynamic Author Topic Model (OSDATM), um novo modelo para acompanhamento da evolução dos interesses dos autores. Diferente dos outros modelos mencionados, o OSDATM não é do tipo LDA. Ao invés de usar

Na suposição do saco de palavras, o OSDATM é sensível à ordenação das palavras e afirma aprender tópicos melhores do que os modelos de tópicos de última geração (YANG et al., 2019).

Figura 11 – Perplexidade em função do número de tópicos no conjunto de dados NIPS



Fonte - Adaptado de (YANG et al., 2019)

A Figura 11 apresenta o desempenho das abordagens apresentadas anteriormente, ou seja, LDA, AT, ATT, AToT e OSDATM. O desempenho foi medido usando a coleção Neural Information Processing Systems (NIPS), que representa os procedimentos das principais conferências de aprendizado de máquina do mundo; e a métrica de perplexidade, que mede como um modelo lida com documentos não vistos, onde quanto menor o resultado, melhor o modelo (WALLACH et al., 2009; NEWMAN et al., 2010).

Além disso, observe que o OSDATM apresenta as melhores avaliações independentemente do número de tópicos. Por outro lado, LDA é o pior. Além disso, observe que os desempenhos de todos os modelos da Figura 11 tendem a se estabilizar quando atingem 100 tópicos.

5.2 ANÁLISES EXPLORATÓRIAS

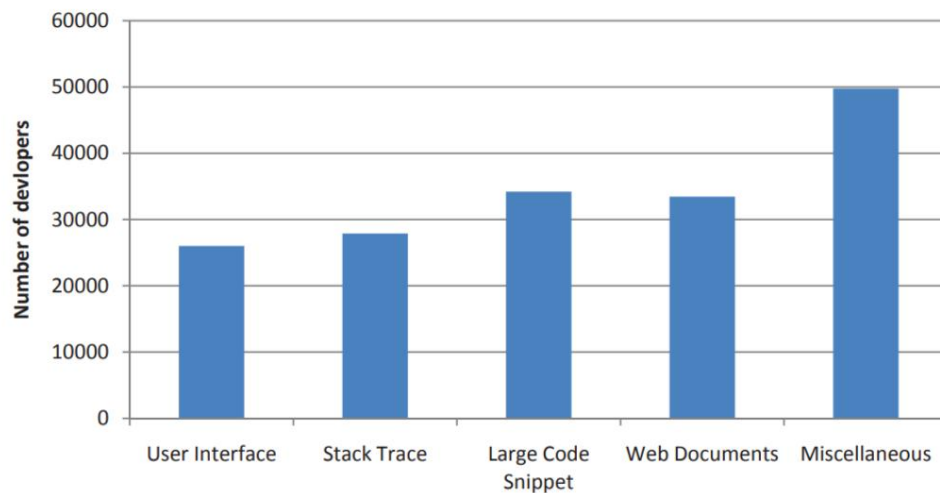
Modelos de tópicos são propostos para serem empregados em outros estudos. Entre outros, um estudo exploratório analisa uma coleção de dados com o objetivo de descobrir informações perspicazes a partir dele. Na modelagem de tópicos, as informações descobertas são fornecidas como distribuições sobre tópicos, que podem ser analisadas para vários propósitos. Portanto, como este trabalho é uma análise exploratória, compreender outras que apresentem objetivos relacionados é essencial para nortear a metodologia abordada neste trabalho.

Wand, Lo e Jiang (2013) apresentam uma análise exploratória das interações do usuário no Stack Overflow. Os autores investigam os comportamentos dos usuários, analisando seu viés como questionadores e respondentes, e emprega modelagem de tópicos para atribuir tópicos a perguntas do Stack Overflow. Elas

extraíram 63.863 perguntas do Stack Overflow, incluindo seus conteúdos, autores e respostas.

A partir daí, a coleção extraída foi submetida a uma etapa de processamento que, entre outros resultados, separou todas as questões de seus trechos de código em uma subcoleção.

Figura 12 – Histograma de desenvolvedores por tópico plotado por Wand, Lo e Jiang (2013)

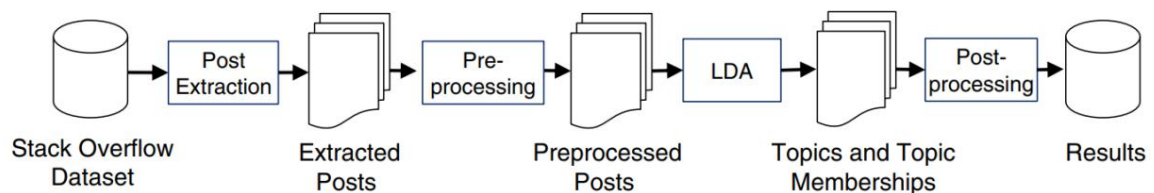


Fonte – (WANG; LO; JIANG, 2013)

Em seguida, foi aplicada uma etapa de mineração de texto, onde uma etapa de pré-processamento realizou tokenização, remoção de stopwords e stemming, e uma etapa de modelagem de tópicos inferiu 5 tópicos tanto de textos normais quanto de trechos de código usando LDA (WANG; LO; JIANG, 2013). A Figura 12 mostra um histograma apresentando os 5 tópicos inferidos e o número de desenvolvedores relacionados a cada tópico. Observe que os tópicos possuem rótulos amplos e o número de desenvolvedores para cada tópico é muito semelhante, exceto para o tópico *Miscelânea*.

No entanto, embora este trabalho supracitado tenha analisado a relação entre tópicos e desenvolvedores, ele não considera o fator tempo. Portanto, Barua, Thomas e Hassan (2014) apresentam uma análise exploratória abordando questões de pesquisa fortemente relacionadas a este trabalho. Entre outros, este trabalho visa descobrir os principais tópicos do Stack Overflow e como os interesses dos desenvolvedores mudam ao longo do tempo.

Figura 13 – Metodologia aplicada por Barua, Thomas e Hassan (2014) em um fluxograma



Fonte – (BARUA; THOMAS; HASSAN, 2014)

A Figura 13 mostra um fluxograma apresentando uma visão geral da metodologia aplicada por Barua, Thomas e Hassan (2014). Eles extraíram 3.474.987 perguntas e respostas de

Stack Overflow, cada um separado em seu próprio documento. Além disso, cada postagem inclui o corpo, carimbo de data/hora, tipo (pergunta ou resposta), tags especificadas pelo usuário e as relações pergunta-resposta.

A etapa de pré-processamento, diferentemente de Wand, Lo e Jiang (2013), descartou todos os trechos de código, pois a maioria do código-fonte é semelhante em sintaxe e palavras-chave, que representam ruídos no processamento posterior (BARUA; THOMAS; HASSAN, 2014). Em seguida, todas as tags HTML foram removidas, mantendo-se apenas o texto. Finalmente, tokenização, remoção de palavras irrelevantes e lematização também foram aplicadas.

O processo de modelagem de tópicos concatenou bigramas na coleção pré-processada e empregou a implementação LDA fornecida por Mallet (MCCALLUM, 2002). Após a experimentação empírica, o autor definiu o número de tópicos para 40, cada um rotulado manualmente com base nas 4 primeiras palavras (BARUA; THOMAS; HASSAN, 2014).

Além disso, como todo documento está associado a todos os tópicos em proporções distintas, $\gamma_j = 0,1$ foi definido como um limite para determinar se um tópico está ou não relacionado a um documento (BARUA; THOMAS; HASSAN, 2014). Em outras palavras, um documento está associado a um tópico se essa associação for maior ou igual a 10%.

Por fim, a etapa de pós-processamento aplicou algumas métricas para responder às questões de pesquisa propostas. Dentre elas, é possível destacar duas métricas: as métricas γ_j e $\gamma_j(\gamma_j)$. A métrica γ_j mede a popularidade relativa de um tópico em todos os documentos. Considerando γ_j a coleção e $\gamma_j(\gamma_j)$ a associação ponderada do tópico para o documento γ_j e um determinado tópico γ_j , métrica é definida pela equação 5.1.





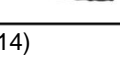
$$\gamma_j(\gamma_j) = \frac{1}{\sum_{j=1}^n \gamma_j(\gamma_j)} \gamma_j(\gamma_j) \quad (5.1)$$

A métrica $\gamma_j(\gamma_j)$ é uma variação da métrica γ_j que “[...] mede a relativa proporção de postagens relacionadas a esse tópico em comparação com os outros tópicos naquele mês específico” (BARUA; THOMAS; HASSAN, 2014, p. 9). Embora o trabalho original tenha considerado fatias de tempo definidas como meses, é possível aplicar a métrica $\gamma_j(\gamma_j)$ a qualquer fatia de tempo. De fato, as métricas γ_j e $\gamma_j(\gamma_j)$ são quase as mesmas, mas $\gamma_j(\gamma_j)$ divide os documentos de acordo com as fatias de tempo a que pertencem. Considerando γ_j um determinado mês (ou fatia de tempo), a métrica $\gamma_j(\gamma_j)$ é definida pela Equação 5.2.

$$\gamma_j(\gamma_j) = \frac{1}{\sum_{j=1}^n \gamma_j(\gamma_j)} \gamma_j(\gamma_j) \quad (5.2)$$

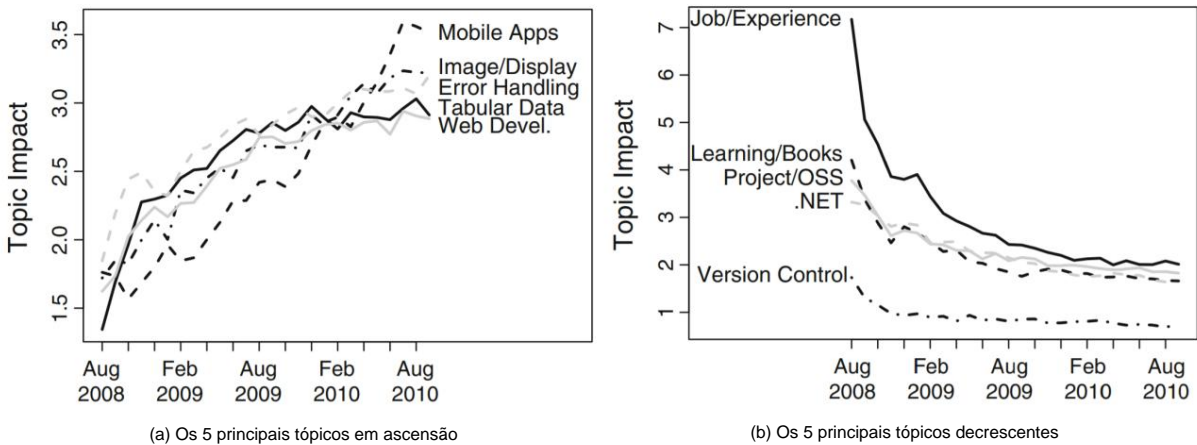
Ao aplicar a métrica $\gamma_j(\gamma_j)$ a todos os 40 tópicos e ordenando-os em ordem decrescente, Barua, Thomas e Hassan (2014) descobriram os principais tópicos de tendência no Stack Overflow de agosto de 2008 a agosto de 2010. A Tabela 2 apresenta os 5 principais tópicos de tendência e seus respectivos $\gamma_j(\gamma_j)$ s e padrões de tendência. Observe que a métrica $\gamma_j(\gamma_j)$ gera medidas semelhantes a proporções, onde a soma de todos os $\gamma_j(\gamma_j)$ s deve ser 1 (ou 100%).

Tabela 2 – Tópicos de tendências do Stack Overflow de Barua, Thomas e Hassan (2014)

Rótulo do tópico	Tendência	Linha de tendência
Estilo/prática de codificação	4,5% \ddot{y}	
Solução do problema	4,5% \ddot{y}	
Controle de qualidade e links	3,8% \ddot{y}	
Função	3,4% \ddot{y}	
Experiência de trabalho	3,3% \ddot{y}	

Fonte – (BARUA; THOMAS; HASSAN, 2014)

Figura 14 – As 5 principais tendências crescentes e decrescentes no Stack Overflow por Barua, Thomas e Hassan (2014)



Fonte – (BARUA; THOMAS; HASSAN, 2014)

Além disso, eles também aplicaram a métrica \ddot{y} $\ddot{y}\ddot{y}\ddot{y}\ddot{y}\ddot{y}\ddot{y}$ para cada fatia de tempo (ou seja, mensal), computando cada evolução do tópico e padrões de tendência. A Figura 14 apresenta dois gráficos, onde 14a mostra a Top 5 tópicos crescentes e 14b mostra os 5 principais tópicos decrescentes de agosto de 2008 a 2010 Agosto.

De acordo com esses e outros resultados, Barua, Thomas e Hassan (2014) afirmam que as discussões relacionadas à web são as mais populares no Stack Overflow, como desenvolvimento web, web design e webservice. Além disso, as discussões de desenvolvimento móvel também são muito comuns e tende a aumentar.

5.3 DISCUSSÃO

Este capítulo apresentou alguns trabalhos que possuem objetivos relacionados aos nossos. Vários deles são modelos de tópicos propostos para lidar com autoria e informações temporais, enquanto outros trabalhos são aplicativos que empregam, entre outros, modelagem de tópicos e técnicas de PNL para rastrear os interesses dos usuários e a evolução dos tópicos a partir do Stack Overflow.

A maioria dos modelos propostos aplicou perplexidade para avaliar e comparar seus resultados com outros modelos. No entanto, embora os tópicos inferidos devam ser significativos para humanos, a perplexidade e outras métricas tradicionais não estão correlacionadas à interpretação humana (CHANG et al., 2009). Em vez disso, as métricas de coerência estão fortemente correlacionadas à interpretação humana e são mais recomendadas do que a perplexidade para avaliar os resultados da modelagem de tópicos (CHANG et al., 2009).

Embora tenham sido apresentadas algumas abordagens de modelos de tópicos que podem ser úteis para os objetivos deste trabalho, as análises exploratórias discutidas aplicaram o LDA como algoritmo de modelo de tópicos. Uma possível razão é que o LDA está disponível em vários frameworks e várias linguagens de programação, enquanto a maioria das abordagens de modelos de tópicos apresentadas são difíceis de encontrar e complexas de implementar. Além disso, considerando que a saída LDA fornece tópicos e suas associações para cada documento, é fácil computar algoritmicamente autoria e tempo relações se a coleção tiver essa informação.

Além disso, as análises exploratórias discutidas também fornecem importantes informações metodológicas. Wand, Lo e Jiang (2013) submeteram apenas questões ao processo de modelagem de tópicos, incluindo texto normal e trechos de código. Diferentemente, Barua, Thomas e Hassan (2014) extraíram tanto perguntas quanto respostas, além de descartar todos os trechos de código alegando que eles representam ruídos para processos posteriores.

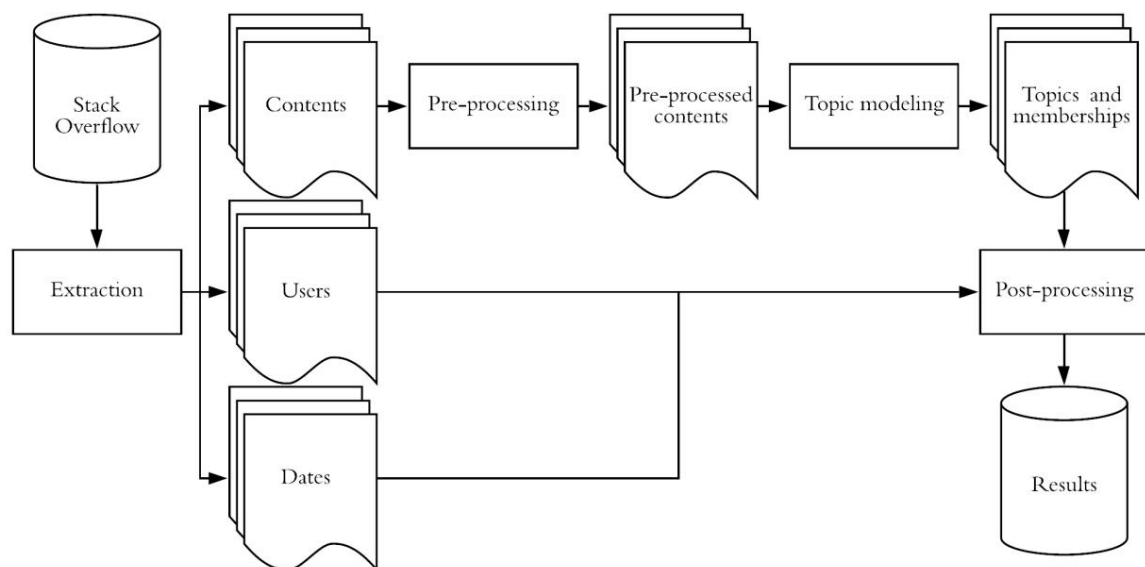
Além disso, ambos os trabalhos empregaram uma etapa de pré-processamento antes da etapa de modelagem de tópicos, que aplicou tokenização, remoção de palavras irrelevantes e lematização. No entanto, Barua, Thomas e Hassan (2014) concatenaram bigramas para cada documento. Além disso, eles inferiram 40 tópicos muito mais detalhados e perspicazes do que os 5 tópicos inferidos por Wand, Lo e Jiang (2013).

Além disso, as métricas γ e γ_{temp} foram notáveis ao calcular os tópicos de tendência e as mudanças ao longo do tempo para cada tópico e fatia de tempo. É possível adaptar ambas as métricas para que considerem informações de autoria, o que seria importante para atingir os objetivos deste trabalho.

6 EXPERIMENTOS

Conforme apresentado no capítulo 1, este trabalho tem como objetivo realizar uma análise exploratória a partir das postagens do Stack Overflow. O objetivo deste estudo é analisar os tópicos discutidos entre os autores e as datas de publicação para acompanhar a evolução da popularidade do tópico, entender suas tendências e identificar como a popularidade do tópico se deslocou ao longo do tempo. Com esses dados, é possível obter insights gerais sobre tópicos de estouro de pilha e insights centrados no usuário para cada usuário.

Figura 15 – Visão geral da metodologia abordada em fluxograma



Fonte – dos autores

Considerando esses propósitos, a Figura 15 apresenta uma visão geral do método empregado em um fluxograma. Em primeiro lugar, extrair postagens do banco de dados do Stack Overflow é uma etapa essencial. A partir daí, a coleção extraída precisa ser submetida a uma etapa de pré-processamento, que limpa e enriquece os pinos extraídos. Em seguida, a etapa de modelagem de tópicos realiza experimentos aplicando várias técnicas para descobrir a distribuição sobre tópicos e associações de tópicos para cada postagem. Por fim, os tópicos descobertos e as informações de autoria temporal são analisados na etapa de pós-processamento para computar métricas e gerar gráficos.

6.1 EXTRAÇÃO

Primeiro, o arquivo Posts.xml¹ foi baixado do despejo de dados do Stack Overflow. Este dump de dados possui outros arquivos XML, um para cada entidade de dados do Stack Overflow (*por exemplo*, Users.xml, Tags.xml, Comments.xml, etc), mas Posts.xml já possui todos os dados necessários: todos os posts não excluídos desde 2008, incluindo autoria e informações temporais. As postagens baixadas foram atualizadas pela última vez em setembro de 2020, incluindo **50.337.841** postagens.

¹ <https://archive.org/download/stackexchange/stackoverflow.com-Posts.7z>

Após o download das postagens, elas foram filtradas de acordo com dois requisitos. Primeiro, cada postagem extraída deve ser uma pergunta ou resposta, que pode ser verificada verificando o atributo `PostTypeId`, onde 1 é para perguntas e 2 para respostas. Em segundo lugar, como as postagens podem apresentar alguma informação ausente, toda postagem extraída deve ter valores não nulos nos atributos `OwnerId`, `CreationDate` e `Body`, pois são essenciais para etapas posteriores. As postagens que violaram qualquer um desses requisitos foram removidas da coleção, resultando em **739.023** postagens removidas e em uma coleção de **49.598.818** postagens.

Após filtrar as postagens, as informações necessárias foram extraídas e armazenadas em arquivos `.txt`. Para cada postagem, o usuário que a criou, a data de publicação e o próprio conteúdo foram armazenados. O autor do post foi extraído do atributo `OwnerId`, que é um número de identificação. Em seguida, a data de criação foi extraída acessando o atributo `CreationDate`, mas horas, minutos e segundos são descartados aplicando um formato `aaaa-mm-dd`.

O conteúdo de uma postagem é composto principalmente pelo atributo `Body`. No entanto, como os atributos `Title` e `Tags` também contêm informações úteis sobre os tópicos relacionados a um post, eles foram concatenados ao conteúdo. No caso de respostas, que não possuem títulos ou tags, o atributo `ParentId` pode ser utilizado para identificar a pergunta pai por seu `Id` e concatenar suas `Tags` ao conteúdo de suas respectivas respostas.

Todos esses dados foram armazenados em três arquivos diferentes: `users.txt`, `datas.txt` e `contents.txt`, onde a i -ésima linha em cada arquivo representa o usuário, a data e o conteúdo pertencentes à postagem i . Como a autoria e as informações temporais são empregadas apenas na etapa de pós-processamento, armazenar o conteúdo separado delas melhora o desempenho ao analisar apenas o conteúdo da postagem.

6.2 PRÉ-PROCESSAMENTO

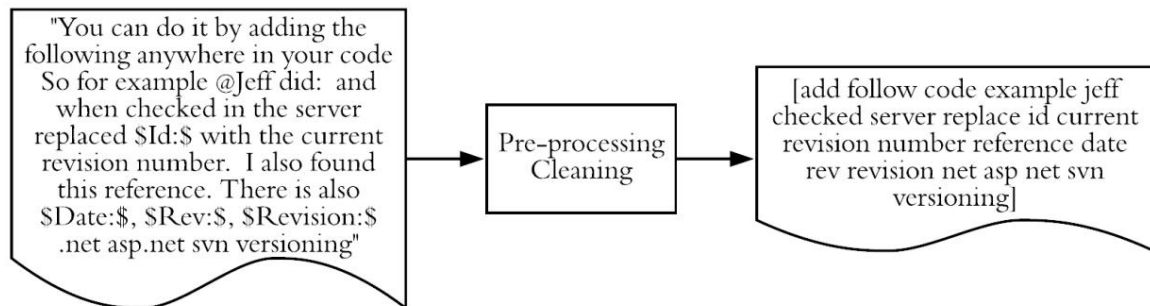
Após a extração dos postes Stack Overflow, seus conteúdos foram submetidos a uma etapa de pré-processamento dividida em duas subetapas: limpeza e enriquecimento. Primeiro, as postagens foram limpas descartando trechos de código (cercados por tags `<code>`), removendo tags HTML (como `<p>` e ``) e realizando tokenização, remoção de palavras irrelevantes e lematização. Para manipulação do HTML, foi empregada a biblioteca Beautiful Soup2 para Python, onde as técnicas de NLP foram obtidas da biblioteca NLTK3.

Embora a lematização apresente bons resultados com melhor desempenho do que a lematização, empregar a lematização pode maximizar a qualidade dos resultados, mesmo impactando no desempenho. Essa perda de desempenho ocorre porque a captura do trecho de fala de cada palavra da coleção é uma tarefa dispendiosa, mas fazer isso ajuda a pré-processar os dados adequadamente e fornecer melhores resultados em etapas posteriores.

² <https://pypi.org/project/beautifulsoup4/> [https://](https://www.nltk.org/)

³ www.nltk.org/

Figura 16 – Conteúdo do post aleatório submetido à subetapa de limpeza



Fonte – dos autores

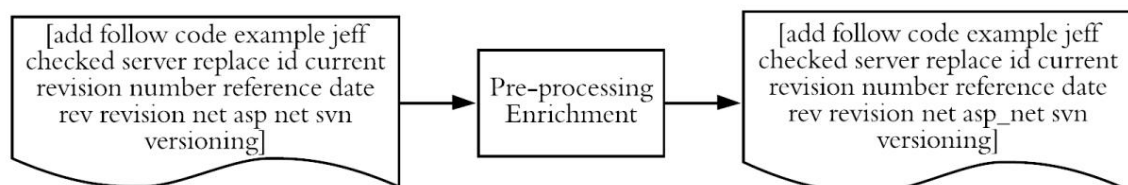
A Figura 16 apresenta um exemplo de aplicação da subetapa de limpeza em um poste aleatório.

Observe que todas as palavras foram convertidas em minúsculas e todas as palavras irrelevantes e sinais de pontuação foram removidos. Além disso, as palavras flexionadas foram reduzidas aos seus lemas, como “adicionar”, “seguir” e “substituir”. Além disso, como a lematização considera parte do discurso, “versioning” não foi equivocadamente reduzido a “version” porque esta palavra é um substantivo.

Após a limpeza dos pinos, os mesmos foram submetidos a uma subetapa de enriquecimento acrescentando informações para refinar os resultados. De fato, calcular *n*-gramas pode melhorar os processos de categorização de texto, mas também pode ser uma tarefa muito cara. Felizmente, bigramas computacionais são suficientes para elevar substancialmente a qualidade de uma coleção (TAN; WANG; LEE, 2002).

Portanto, a subetapa de enriquecimento consiste em computar e concatenar os bigramas de cada post ao seu conteúdo. Em outras palavras, essa subetapa resultou em uma coleta usando tanto unigramas quanto bigramas. O resultado final da etapa de pré-processamento foi armazenado no arquivo `pre-processed-contents.txt`, que segue as mesmas regras apresentadas na etapa de extração: a *n*-ésima linha neste arquivo representa o conteúdo pré-processado pertencente ao *n* postar.

Figura 17 – Conteúdo do post aleatório submetido à subetapa de enriquecimento



Fonte – dos autores

A Figura 17 apresenta um exemplo de aplicação da subetapa de enriquecimento ao mesmo post aleatório da Figura 16. Embora não haja muitas alterações, observe que a subetapa gerou com sucesso o bigrama “asp_net”. Separadas, as palavras “asp” e “net” podem apresentar muitas

significados diferentes de acordo com a frase, mas como um bigrama, eles estão claramente relacionados ao ASP.NET, um framework web de código aberto.

6.3 MODELAGEM DO TÓPICO

Após o pré-processamento da coleção, foi executada a etapa de modelagem do tópico. Esta etapa consiste em realizar experimentos aplicando os conteúdos pré-processados a técnicas de modelagem de tópicos para inferir a distribuição sobre os tópicos e gerar as associações de tópicos para cada postagem. Essa etapa também possui três subetapas: construção do corpus, inferência LDA e rotulagem. Essas subetapas foram implementadas empregando a biblioteca tomotopy⁴ para Python.

6.3.1 Construção do Corpus

Primeiramente, deve-se definir o corpus para realizar as inferências LDA. Embora a etapa de pré-processamento remova e resuma o conteúdo do post para melhorar os resultados da modelagem do tópico, existe a possibilidade de que posts de conteúdo curto tenham todas as palavras removidas. Portanto, esses posts vazios foram ignorados na construção do corpus, resultando em **25.214** posts ignorados e **49.573.604** posts usados.

Além disso, o dicionário de corpus define quais palavras a inferência do modelo de tópicos considera ao inferir tópicos. Filtrar as palavras no dicionário pode ajudar a melhorar a qualidade dos tópicos inferidos. Descartar palavras que aparecem em poucas postagens ou que aparecem em muitas postagens pode ajudar a manter apenas as informações relevantes. No entanto, não existem parâmetros de filtragem adequados a todos os casos, pois produzem resultados diferentes dependendo da coleção.

Portanto, a coleção foi filtrada por muitos parâmetros diferentes para definir a filtragem adequada. Palavras com uma frequência de documentos menor que 200 foram removidas do dicionário, enquanto as 20 palavras mais importantes também foram removidas. A Tabela 3 apresenta as estatísticas sobre as palavras e vocabulários do corpus, incluindo os removidos.

Tabela 3 – Definição do corpus

Palavras	Vocabes Vocais Usados
1.523.460.528	5.836.775 59.478

Por fim, a coleção foi transformada em um corpus apto a ser aplicado em uma inferência de modelo de tópicos. Como o LDA foi escolhido para realizar a inferência do tópico, o corpus deve estar no formato bag of word. Portanto, o TF-IDF foi empregado para transformar e preparar o corpus para Inferência LDA.

⁴ <https://bab2min.github.io/tomotopy/>

6.3.2 Inferência LDA

A subetapa de inferência consiste em realizar vários experimentos com inferência LDA empregando diferentes hiperparâmetros e medindo a coerência de cada modelo de tópico inferido. Os hiperparâmetros de interesse são o número de tópicos γ e o número de iterações β .

Considerando que quanto maior a coerência $\gamma\beta$ melhor o modelo, esses experimentos são úteis para encontrar o melhor conjunto de γ e β .

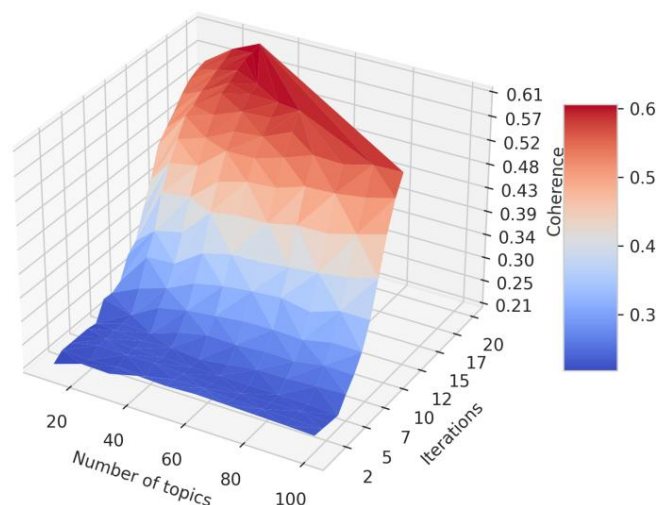
O número de iterações β define quantas vezes o modelo itera a coleção inteira para inferir os tópicos. Como mais iterações não necessariamente resultam em modelos melhores, é importante considerá-las nos experimentos. No entanto, à medida que mais iterações resultam em modelos LDA maiores, o número de iterações β também afeta o desempenho, pois os recursos computacionais são finitos.

Além disso, γ é muito determinante para definir o comportamento geral dos tópicos inferidos. Valores maiores de γ produzem tópicos mais detalhados e de granulação mais fina, enquanto valores menores de γ produzem tópicos mais gerais e de granulação mais grosseira (BARUA; THOMAS; HASSAN, 2014). Além disso, o tamanho da coleção também é determinante da granularidade dos tópicos inferidos γ . Consequentemente, não há valor de γ que seja mais apropriado para todas as coleções, pois cada caso pode apresentar comportamentos diferentes. Portanto, os experimentos são essenciais para definir o melhor número de tópicos γ para resumir os dados do Stack overflow.

Por fim, vários experimentos foram realizados, cada um com uma combinação de hiperparâmetros provenientes de $\gamma = \{10, 20, 30, \dots, 100\}$ e $\beta = \{1, 2, 3, \dots, 20\}$. Após realizar cada experimento, a coerência $\gamma\beta$ foi calculada para avaliar a qualidade do modelo de tópico. Todos os hiperparâmetros configurados e suas respectivas coerências foram armazenados no arquivo experiment.csv, permitindo análises posteriores.

Figura 18 – Gráfico 3D do Stack Overflow $\gamma\beta$ coerência por número de tópicos γ e iterações β

Best experiment: iterations=20 topics=30 coherence=0.6140



Fonte – dos autores

A Figura 18 apresenta um gráfico 3D triangulado com os resultados desses experimentos. Aparentemente, à medida que γ e β aumentaram, a coerência de $\gamma\beta\gamma\beta$ também aumentou. Além disso, todos os tópicos apresentaram baixa coerência com menos de 12 iterações e o melhor experimento apresentou aproximadamente 0,6140 de coerência para 30 tópicos e 20 iterações. Esse resultado significa que o melhor número de tópicos para esta coleção e esses experimentos é de 30 tópicos. O Apêndice A mostra os tópicos inferidos e suas respectivas palavras principais.

6.3.3 Rotulagem

A última subetapa para modelagem de tópicos é rotular os tópicos inferidos. Não há método que pode ser considerado o mais apropriado para a rotulagem de tópicos. Vários estudos de modelagem de tópicos rotulam os tópicos analisando as 10 principais palavras de cada tópico manualmente e empiricamente, enquanto outros estudos preferem perguntar rotulagem humana.

Neste trabalho, o processo de rotulagem foi realizado através da busca das 10 palavras mais importantes de cada tópico no Google e no Google Acadêmico. Como o Stack Overflow apresenta perguntas e respostas sobre tópicos de programação de computadores, esse método de rotulagem foi eficiente porque muitas palavras principais foram facilmente encontradas em documentações. Os resultados da pesquisa foram analisados empiricamente para entender seus fatores comuns e definir um rótulo.

No entanto, como alguns resultados de pesquisa apresentaram ambiguidades ou nenhum fator comum, os resultados foram aprimorados removendo as palavras menos relevantes da consulta de pesquisa. O Apêndice A mostra os rótulos atribuídos a cada tópico de acordo com suas respectivas 10 principais palavras. Finalizando esta etapa, os tópicos rotulados e suas respectivas top-10 palavras foram armazenados no arquivo rotulado-topics.csv.

6.4 DISCUSSÃO

As etapas realizadas neste capítulo tiveram como objetivo preparar o conteúdo das postagens e inferir a distribuição sobre os tópicos delas. Vários experimentos foram realizados para definir os tópicos, onde a coerência $\gamma\beta\gamma\beta$ foi o principal fator para isso. Portanto, considerando as medidas $\gamma\beta\gamma\beta$ e os recursos computacionais que este trabalho dispunha, o melhor número de tópicos é 30.

Como o número de iterações chegou a apenas 20 devido ao poder computacional limitado, realizar esses experimentos com mais iterações poderia melhorar ainda mais a qualidade. No entanto, a distribuição dos resultados por tópicos e as atribuições de rótulos geraram tópicos muito promissores. Eles são distintos um do outro e parecem ser mais específicos do que amplos. Além disso, como a coerência $\gamma\beta\gamma\beta$ é relativamente próxima da interpretação humana dos tópicos (CHANG et al., 2009; LAU; NEWMAN; BALDWIN, 2014), esses resultados são sólidos e bem fundamentados mesmo com 20 iterações.

7 RESULTADOS

Após concluir todos os experimentos e encontrar os tópicos do Stack Overflow, foi realizada a etapa de pós-processamento para analisar os resultados e computar as métricas. Esta etapa consiste em usar a autoria e informações temporais, os tópicos inferidos e as associações de tópicos para fazer análises exploratórias. Essas análises são divididas em duas categorias: análises gerais e análises centradas no usuário. Para cada um, várias métricas foram computadas e disponibilizadas no site Internet Archive¹, permitindo que qualquer pessoa na Web analise os resultados gerados.

7.1 MÉTRICAS

Primeiramente, é necessário entender as métricas empregadas para fazer essas análises. Propostas por Barua, Thomas e Hassan (2014), as métricas $\gamma_{t,d}$ (ver Equação 5.1) e $\gamma_{t,u}$ (ver Equação 5.2) podem medir a popularidade relativa de um tópico em todos os documentos \mathcal{D} e nos documentos \mathcal{D}_u pertencentes para um determinado mês m , respectivamente. De fato, ambas as métricas são muito semelhantes; a diferença entre eles é que a métrica $\gamma_{t,d}$ analisa toda a coleção, enquanto $\gamma_{t,u}$ analisa uma subcoleção composta pelas postagens de um determinado mês.

No entanto, como essas métricas não consideram informações de autoria, elas foram adaptadas para se adequarem a este trabalho. Em vez de calcular a popularidade de um tópico em um conjunto predefinido de documentos, é possível calcular a mesma métrica em qualquer conjunto de documentos. Portanto, considerando \mathcal{D} um determinado conjunto de documentos, \mathcal{T} um determinado tópico e $\gamma(\mathcal{T}, \mathcal{D})$ a associação de tópicos ponderada para o documento \mathcal{D} e o tópico \mathcal{T} , a popularidade do tópico \mathcal{T} é a métrica proposta definida pela equação 7.1.

$$\gamma(\mathcal{T}, \mathcal{D}) = \frac{1}{|\mathcal{D}|} \sum_{d \in \mathcal{D}} \gamma(\mathcal{T}, d) \quad (7.1)$$

Dado um tópico \mathcal{T} e um conjunto de documentos \mathcal{D} obtidos da coleção original \mathcal{D} , a popularidade do tópico \mathcal{T} $\gamma(\mathcal{T}, \mathcal{D})$ é a média das associações de tópicos $\gamma(\mathcal{T}, d)$ soma para cada documento d pertencente ao conjunto de documentos \mathcal{D} . No entanto, observe que esses $\gamma(\mathcal{T}, d)$ são filtrados de acordo com um limite θ , que foi definido como 10%. Esse recurso significa que tópicos com menos de 10% de $\gamma(\mathcal{T}, d)$ não estão associados a um documento d .

Essa filtragem é necessária porque os modelos LDA associam todos os tópicos a todos os documentos, mesmo que o valor de $\gamma(\mathcal{T}, d)$ seja muito baixo. No caso de um documento d não apresentar tópicos acima de θ , o $\gamma(\mathcal{T}, d)$ mais alto é manter e qualquer outro $\gamma(\mathcal{T}, d)$ igual ao mais alto também é manter.

Como forma de variação das métricas $\gamma_{t,d}$ e $\gamma_{t,u}$, a Popularidade do Tópico é uma métrica mais genérica que pode calcular a popularidade de um tópico em relação a qualquer conjunto de documentos. Esse recurso permite analisar subcoleções de acordo com qualquer tipo de condição, como uma subcoleção que inclua apenas postagens criadas por um determinado usuário em um determinado mês.

¹ <https://archive.org/details/staty>

Além disso, como um dos propósitos deste trabalho é identificar como os usuários são leais ou derivam os tópicos de seu interesse, é necessário calcular uma métrica capaz de medir a deriva ou variância de um tópico. Portanto, o desvio padrão estatístico pode ser usado para essa tarefa, pois essa métrica pode medir a variância e a dispersão de qualquer conjunto de valores. Assim, considerando \bar{y} um determinado conjunto de medidas de popularidade de tópicos, \bar{y} seu valor médio e \bar{y}^2 a \bar{y} -ésima popularidade de tópicos dada, o desvio de popularidade de tópicos \bar{y} \bar{y}^2 é definido pela equação 7.2.

$$\bar{y} \bar{y}^2 (\bar{y}) = \frac{1}{|\bar{y}|} \sqrt{\sum_{i=1}^n (\bar{y}_i - \bar{y})^2} \quad (7.2)$$

Considerando \bar{y} um determinado conjunto de medidas de popularidade de tópico previamente calculadas \bar{y}^2 , para o tópico \bar{y} $\bar{y}^2 (\bar{y})$ é o desvio padrão de \bar{y} , que é definido como a raiz quadrada da variância de \bar{y} . Esta variância calcula a média da distância ao quadrado cada popularidade de tópico \bar{y}^2 é da média \bar{y} . Como quanto mais próxima uma medida computada estiver de zero, menor será a variação do tópico \bar{y}^2 , essa métrica pode ser útil para analisar a fidelidade do tópico.

Em vez de propor uma métrica específica para usuários e fatias de tempo, o Topic Popularity Drift é proposto como uma métrica genérica capaz de ser aplicada a qualquer conjunto de medidas de Topic Popularity. Por exemplo, se \bar{y} é calculado a partir de postagens em uma fatia de tempo específica, $\bar{y} \bar{y}^2 (\bar{y})$ é a Deriva de popularidade do tópico do tópico \bar{y}^2 somente nesta fatia de tempo. Além disso, como a popularidade do tópico resulta em valores entre 0 e 1, os resultados de $\bar{y} \bar{y}^2 (\bar{y})$ também seguem essa regra.

7.2 ANÁLISES GERAIS

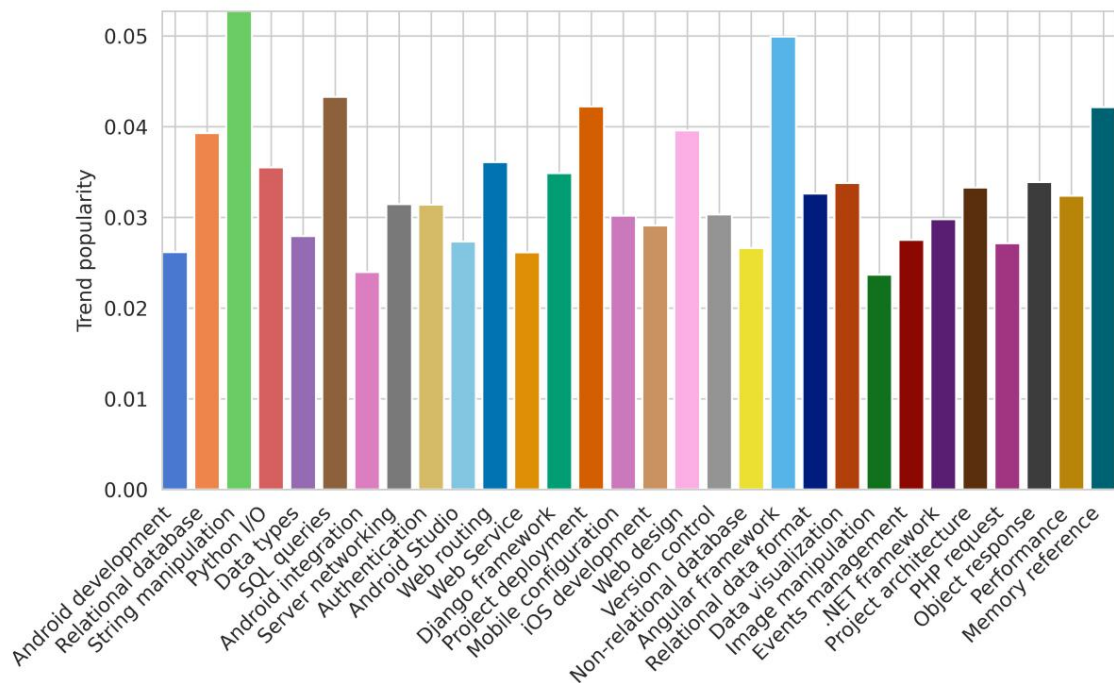
As análises gerais não consideram informações de autoria e, conseqüentemente, visam fornecer insights relacionados a todas as postagens do Stack Overflow. As três análises empregadas computam métricas para entender os tópicos de tendência, sua evolução e sua deriva por medidas de popularidade do tópico. Essas análises gerais resultaram em **4.421** cálculos de métricas, que foram armazenados em arquivos CSV com sufixo geral em seus nomes.

A primeira análise geral consiste em, para cada tópico, computar a popularidade do tópico em toda a coleção, sem considerar informações temporais ou de autoria. O resultado dessa análise é a popularidade geral de cada tópico em todo o histórico do Stack Overflow, identificando os tópicos de tendência, que são armazenados no arquivo general-trend.csv.

A Figura 19 apresenta um gráfico de barras contendo as medidas computadas de popularidade do tópico em todas as postagens. Observe que, embora essas medidas não variem muito, os tópicos *Manipulação de String* (0,052), *Angular Framework* (0,049), *Consultas SQL* (0,043), *Implantação de projeto* (0,042) e *Referência de memória* (0,042) são os 5 principais globais tópicos de tendência no Stack Overflow.

Depois disso, foi analisada a evolução mensal da popularidade de cada tópico em todas as postagens. Essa análise forneceu informações úteis sobre como os tópicos discutidos no Stack Overflow mudam ao longo do tempo, como suas tendências de aumento e diminuição. Nesse caso, o arquivo date.txt foi necessário para dividir as postagens em subcoleções de acordo com os intervalos de tempo mensais. Então, para cada

Figura 19 – Tópicos de tendências gerais no Stack Overflow



Fonte – dos autores

topic, a popularidade do tópico foi calculada em todas as subcoleções, gerando um conjunto de medidas de popularidade do tópico ao longo de fatias de tempo, que são armazenadas no arquivo general-popularity.csv.

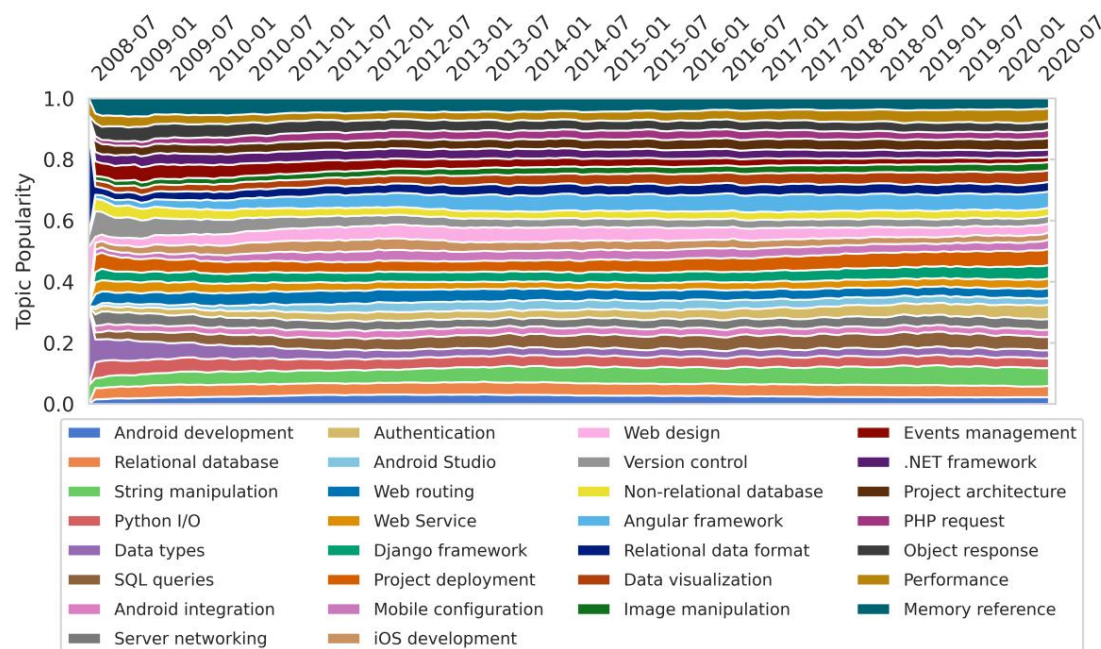
O Apêndice B apresenta a evolução da popularidade de cada tópico. Observe que cada tópico apresentou uma linha de evolução distinta, dando uma ideia geral de suas tendências de aumentar, diminuir ou manter-se constante. Além disso, como o número de postagens nos primeiros meses do Stack Overflow foi muito baixo, um comportamento extremo de aumento ou diminuição é claramente visível nesses primeiros meses. Por exemplo, o tópico Formato de dados relacionais apresentou uma queda de popularidade muito repentina, enquanto o gerenciamento de Eventos apresentou uma queda de popularidade muito repentina.

Para entender melhor a evolução do tópico do Stack Overflow, a Figura 20 apresenta um gráfico de área empilhada com a evolução mensal da popularidade de cada tópico em todas as postagens, onde os rótulos de data são agrupados em semestres para evitar sobreposição de rótulos. Este gráfico funciona como um gráfico de pizza, mas com informações temporais, o que significa que a soma da popularidade de todos os tópicos em um mês é 1. Este gráfico é muito útil para entender como a popularidade dos tópicos muda ao longo do tempo, permitindo comparar suas tendências de evolução.

Outro dado interessante para analisar é como esses tópicos derivaram em suas próprias linhas de evolução. Observe que as medidas de Popularidade do Tópico apresentadas na Figura 20 não variam muito, o que é uma característica também presente na Figura 19. No entanto, o Apêndice B mostra que vários tópicos apresentam variação repentina nos primeiros meses.

Portanto, a última análise geral emprega a métrica Topic Popularity Drift para medir a variabilidade que cada tópico apresenta ao longo das fatias de tempo mensais, que são armazenadas no

Figura 20 – Evolução da popularidade do tópico geral por mês no Stack Overflow



Fonte – dos autores

arquivo general-drift.csv. O cálculo dessa métrica requer o conjunto de medidas de popularidade do tópico de cada tópico, resultando em medidas de desvio padrão, onde quanto menor a medida, mais constante o tópico é. Como esta medida é estatística, tem muitas possibilidades de uso em diferentes estudos de variabilidade.

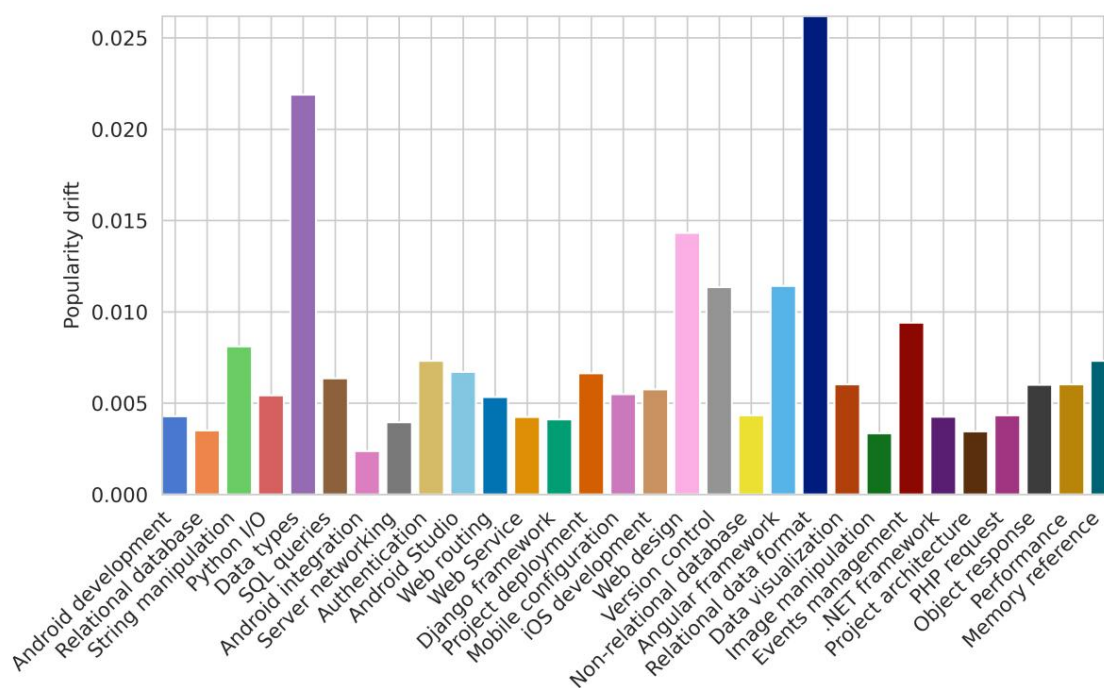
A Figura 21 apresenta um gráfico de barras contendo as medidas Topic Popularity Drift para cada tópico do Stack Overflow. Observe que os tópicos mais variáveis são “Formato de dados relacionais” (0,0261) e “Tipos de dados” (0,0218), que, conforme a Figura 20, iniciam seu histórico no Stack overflow com alta popularidade, mas diminuem essa popularidade ao longo do tempo. Por outro lado, “Integração Android” (0,0023) e “Manipulação de imagem” (0,0033) são os tópicos mais constantes, embora apresentem menor tendência de popularidade.

7.3 ANÁLISES CENTRADAS NO USUÁRIO

Após as análises gerais, foram realizadas análises centradas no usuário. Elas consistem em computar as mesmas métricas empregadas nas análises gerais, mas, ao invés de aplicá-las a todos os posts, essas métricas foram computadas separadamente para cada usuário armazenado no arquivo user.txt. Para isso, a coleção foi dividida em subcoleções por seus autores, cada uma com seu próprio conjunto de tópicos de tendência, evolução de popularidade de tópicos e deriva de popularidade, permitindo obter vários tipos de insights centrados no usuário.

Considerando todos os métodos de filtragem empregados, o número final de usuários na coleção

Figura 21 – Desvio geral de popularidade do tópico no Stack Overflow



Fonte – dos autores

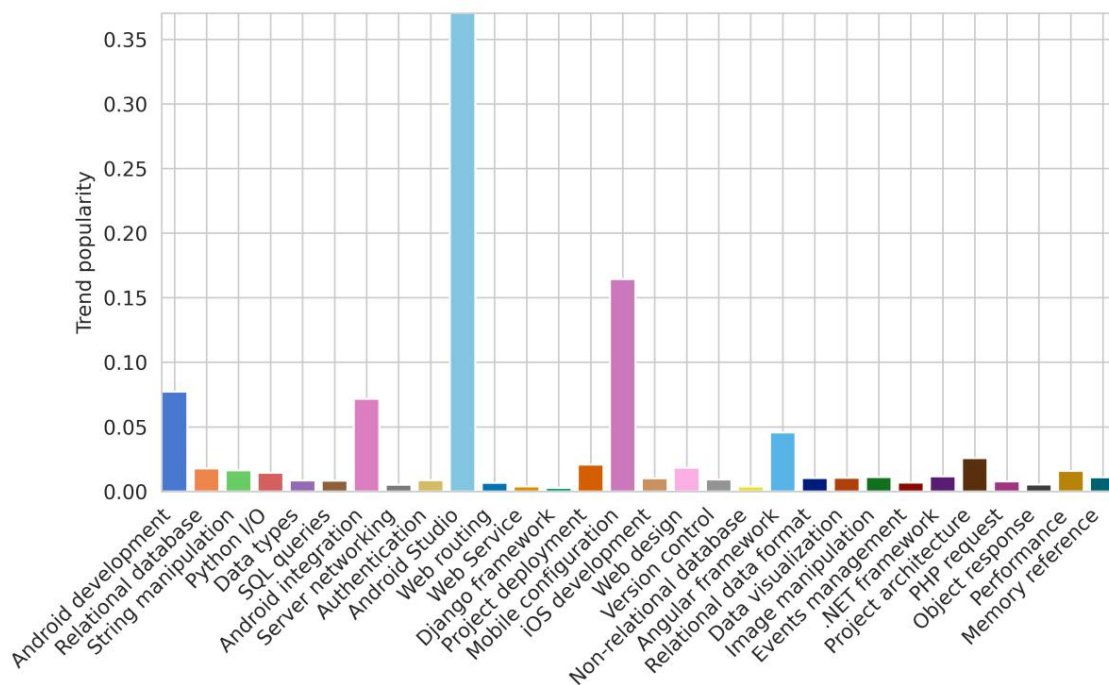
foi de 4.943.206. Como cada análise foi feita individualmente para cada usuário, o número de métricas computadas nas análises centradas no usuário foi de 135.841.897. Todas essas métricas computadas foram armazenadas da mesma forma que as análises gerais foram: nos arquivos user-trends.csv, user-popularity.csv e user-drift.csv.

A Figura 22 apresenta os tópicos de tendência para o ID de usuário 1.289.716, que foi escolhido aleatoriamente entre os usuários com pelo menos um ano de contribuição no Stack Overflow. Observe que os 5 principais tópicos de tendência desse usuário são claramente visíveis: *Android Studio* (0,36), *configuração móvel* (0,16), *desenvolvimento Android* (0,08), *integração Android* (0,07) e *Angular Framework* (0,04). Com essas informações, é possível considerar esse usuário como um grande colaborador em tópicos relacionados ao desenvolvimento Android em diferentes frameworks, como Android Studio e Angular. No entanto, embora as preferências do usuário estejam claramente definidas, esse usuário ainda se interessa por outros tópicos, como mostra a figura.

A Figura 23 apresenta a evolução da popularidade do tópico do mesmo usuário. O gráfico mostra claramente a preferência desse usuário pelo tema *Android Studio*, que foi abordado por vários anos. Além disso, os demais tópicos de tendência também são abordados por muitos meses, mesmo apresentando menor popularidade. Considerando os dados apresentados deste usuário, é possível sugerir que este usuário possui sólidos conhecimentos em frameworks de desenvolvimento Android, principalmente o Android Studio, e esteve muito empenhado em contribuir para esta área no Stack Overflow de março de 2012 a julho de 2015.

Por fim, a Figura 24 apresenta o desvio de popularidade do tópico. Embora esse usuário tenha apresentado preferência por tópicos relacionados ao Android, o gráfico mostra que os tópicos de tendência para esse usuário têm a

Figura 22 – Tópicos de tendências para o usuário 1.289.716



Fonte – dos autores

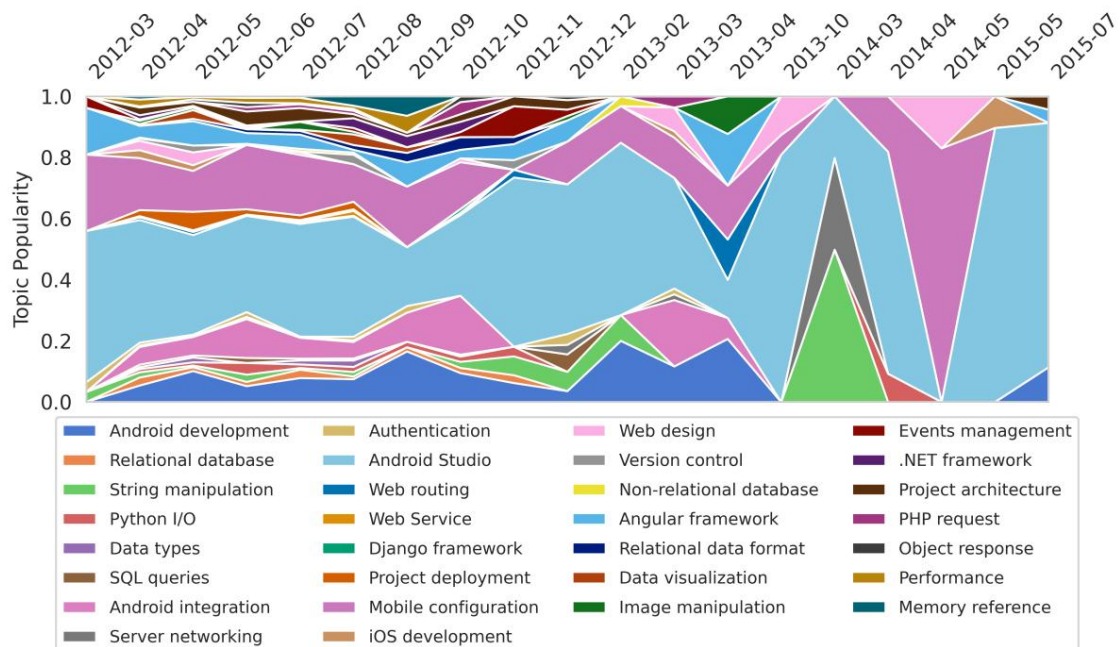
maiores desvios entre todos os outros tópicos. Além disso, os tópicos menos populares para este usuário apresentam menor desvio, pois suas variações absolutas são menores do que os tópicos de tendência.

7.4 DISCUSSÃO

Este capítulo apresentou as métricas propostas e empregadas para analisar perguntas e respostas do Stack Overflow nos tópicos inferidos. A métrica Topic Popularity Drift foi proposta para medir como um conjunto de medidas de popularidade varia ao longo do tempo, o que permite entender a fidelidade de um usuário neste tópico. No entanto, os resultados centrados no usuário mostraram que basear essa métrica no desvio padrão estatístico geralmente faz com que os tópicos menos populares apresentem menor desvio do que os tópicos de tendência. A razão para esse comportamento é que a variação absoluta mensal de tópicos com alta popularidade tende a ser maior à medida que a dominância de popularidade do tópico é maior.

Por exemplo, suponha que o usuário U escreveu alguns posts onde o tópico A tem 0,6 de popularidade, o tópico B 0,3 e o tópico C 0,1. Então o tópico A aumenta sua popularidade para 0,8, o tópico B diminui para 0,2 e o tópico C para 0. Como o tópico A é o dominante e aumentou sua popularidade, este usuário é fiel ao tópico A. Porém, como o Topic Popularity Drift é maior para o tópico A, sugere-se que este tópico é o que mais deriva. Portanto, embora a métrica Topic Popularity Drift possa fornecer informações úteis para analisar a variação da popularidade do tópico, considerar a dominância do tópico nesta análise de lealdade pode melhorar os resultados.

Figura 23 – Evolução da popularidade do tópico para o usuário 1.289.716



Fonte – dos autores

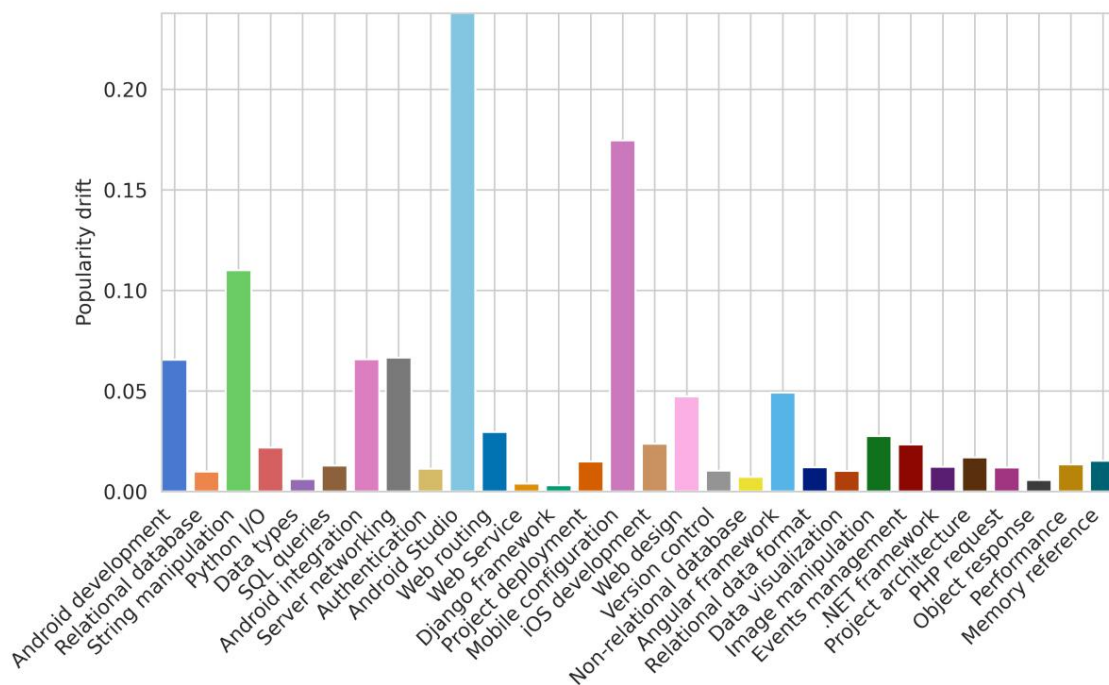
Por outro lado, a métrica Topic Popularity foi muito versátil nas análises feitas.

Essa métrica calculou com sucesso a popularidade relativa do tópico em várias subcoleções diferentes de documentos, permitindo analisar as postagens escritas por usuários individuais em muitas fatias de tempo diferentes. Esse resultado sugere que a métrica Topic Popularity é capaz de considerar qualquer outra informação nas análises; apenas divida a coleção por essas informações e o Topic Popularity pode lidar com o trabalho.

Além disso, a compreensão dos gráficos apresentados mostra que há muito conhecimento oculto nesses cálculos de métricas. As análises gerais podem ser exploradas em muitos outros aspectos, como especificar alguns intervalos de fatia de tempo para analisar as tendências de seus tópicos ou tentar prever a popularidade dos tópicos no futuro. Além disso, as métricas centradas no usuário permitem análises profundas e individuais dos tópicos de interesse de cada usuário, o que pode ser útil para muitas finalidades. Por isso, a contribuição mais importante deste trabalho é tornar todos esses resultados acessíveis a qualquer pessoa com conexão à internet, para que análises muito mais exploratórias possam ser realizadas.

Embora este trabalho não explore profundamente os resultados computados, ele fornece dados atualizados sobre o estado atual dos tópicos do Stack Overflow. Como o Apêndice B apresenta a popularidade do tópico e suas tendências, os 3 principais tópicos *Manipulação de String*, *Estrutura Angular* e *Consultas SQL* ainda estão em uma tendência crescente, enquanto a *Arquitetura do projeto* está estagnada e a *referência de memória* continua diminuindo ao longo do tempo. No entanto, outros tópicos apresentaram tendências crescentes interessantes, como *visualização de dados* e *manipulação de imagens*, o que sugere que as áreas de visão computacional e ciência de dados ainda estão se tornando mais populares.

Figura 24 – Desvio de popularidade do tópico para o usuário 1.289.716



Fonte – dos autores

No entanto, esses resultados não se limitam ao que este trabalho apresentou, pois podem ser muito mais explorados. Por exemplo, a equipe do Stack Overflow pode usar esses dados para entender melhor o conteúdo gerado pelo usuário nos últimos meses para analisar a tendência atual.

Se algum tópico se tornar muito dominante sobre os outros, a moderação do Stack Overflow pode observar esse comportamento analisando a evolução da popularidade do tópico, o que pode ser útil em algumas tomadas de decisão.

Além disso, este trabalho forneceu dados exclusivos para cada usuário do Stack Overflow, o que nunca havia sido feito antes. Cada usuário contém dados suficientes para análises completas e profundas sobre seus tópicos de interesse, permitindo análises pessoais e centradas no usuário sobre tendências de popularidade de tópicos, deriva e evolução. O potencial desses dados para rastrear os interesses dos usuários se mostrou grande e as possibilidades de uso são muito numerosas. Por exemplo, a equipe do Stack Overflow pode usar esses dados para entender melhor os grupos de usuários de acordo com seus interesses, a fim de fornecer ferramentas específicas para eles.

8 CONCLUSÕES

Este trabalho propôs uma metodologia para descobrir e analisar os tópicos de tendência no Stack Overflow, uma comunidade popular de perguntas e respostas sobre programação de computadores. Experimentos foram feitos para definir a melhor distribuição de tópicos para posts do Stack Overflow empregando LDA. Além disso, foram propostas métricas para medir a popularidade e a deriva de cada tópico em informações temporais de postagem e autoria, gerando diversos dados que foram disponibilizados no site Internet Archive¹ para quem quiser validar e explorar os resultados.

Como o número de tópicos é um fator importante nos estudos de modelagem de tópicos, este trabalho contribuiu para encontrar o melhor número de tópicos que resumem as perguntas e respostas do Stack Overflow. Como a coerência $\gamma\gamma\gamma\gamma$ foi empregada para avaliar os modelos de tópicos experimentados, os tópicos resultantes são relativamente próximos da interpretação humana dos tópicos. Além disso, os rótulos atribuídos abordam vários tópicos de programação de computadores e são mais específicos do que amplos, sugerindo que os tópicos descobertos resumem com sucesso o conhecimento compartilhado do Stack Overflow.

Desde que os tópicos foram descobertos, eles se tornam essenciais para computar as métricas que tornam esse trabalho significativo. Cada métrica computada foi importante para construir os resultados, fornecendo dados ricos para serem explorados. Além disso, como esses dados foram disponibilizados gratuitamente na web, este trabalho também contribuiu para a democratização do conhecimento. Portanto, qualquer pesquisador pode empregar esses dados para fornecer insights do Stack Overflow para melhorar os recursos e a ferramenta disponíveis.

No entanto, os usos desses dados também estão além do site do Stack Overflow. Por exemplo, um painel interativo com esses dados pode ajudar os recrutadores de vagas a encontrar usuários interessados em tópicos específicos e verificar quais deles possuem a experiência necessária para uma demanda específica. Portanto, outro trabalho futuro possível é construir um site onde esses resultados possam ser analisados interativamente por qualquer pessoa.

As contribuições fornecidas podem ser exploradas significativamente em trabalhos futuros e há uma muitas aplicações potenciais para os resultados gerados. Este trabalho pode ser a base para estudos mais aprofundados sobre tópicos do Stack Overflow, pois os resultados permitem análises gerais e centradas no usuário. Além disso, incluir outros metadados do Stack Overflow nas postagens pode fornecer mais dados para análise, como analisar a quais tópicos cada tag de postagem está relacionada.

Além disso, validar as métricas propostas e propor outras métricas podem ajudar a melhorar as análises exploratórias no campo de modelagem de tópicos. Embora o Topic Popularity Drift meça com sucesso a variação e a dispersão da popularidade de um conjunto de tópicos, ele não é consistente para a análise de lealdade do tópico. No entanto, muitas métricas podem ser empregadas para essa tarefa, como

² como $\gamma\gamma$ métrica.

Por fim, embora este trabalho tenha analisado perguntas e respostas do Stack Overflow, a metodologia abordada pode ser empregada em outros recursos textuais, como portais web, blogs, fóruns, newsletters, repositórios científicos e muito mais. Comparar o uso dessa metodologia em outros recursos pode validá-la e fornecer resultados perspicazes em outros contextos.

¹ <https://archive.org/details/staty>

REFERÊNCIAS

- ALETRAS, Nikolaos; STEVENSON, Marcos. Avaliando a coerência do tópico usando semântica distributiva. In: PROCEDIMENTOS da 10ª Conferência Internacional de Semântica Computacional (IWCS 2013) – Long Papers. [Sl: sn], 2013. p. 13-22.
- ALLAHYARI, Mehdi; KOCHUT, Krys. Rotulagem automática de tópicos usando modelos de tópicos baseados em ontologia. Em: IEEE. 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA). [Sl: sn], 2015. p. 259-264.
- BARUA, Anton; THOMAS, Stephen W; HASSAN, Ahmed E. Do que os desenvolvedores estão falando? uma análise de tópicos e tendências em estouro de pilha. **Empirical Software Engineering**, Springer, v. 19, n. 3, pág. 619-654, 2014.
- BEEL, Joeran et al. sistemas de recomendação de papel: um levantamento da literatura. **Revista Internacional de Bibliotecas Digitais**, Springer, v. 17, n. 4, pág. 305-338, 2016.
- BIRD, Steven; KLEIN, Ewan; LOPER, Eduardo. **Processamento de linguagem natural com Python: analisando texto com o kit de ferramentas de linguagem natural**. [Sl]: "O'Reilly Media, Inc.", 2009.
- BISHOP, Christopher M. **Reconhecimento de padrões e aprendizado de máquina**. [Sl]: Springer, 2006.
- BLEI, David M. Modelos de tópicos probabilísticos. **Comunicações da ACM**, ACM New York, NY, USA, v. 55, n. 4, pág. 77-84, 2012.
- BLEI, David M; LAFFERTY, John D. Modelos de tópicos dinâmicos. In: PROCEDIMENTOS da 23ª Conferência Internacional de Aprendizado de Máquina. [Sl: sn], 2006. p. 113-120.
- BLEI, David M; NG, André Y; JORDAN, Michael I. Alocação de dirichlet latente. **Journal of Machine Learning Research**, v. 3, Jan, p. 993-1022, 2003.
- CAVUSOGLU, Huseyin; LI, Zhuolun; HUANG, Ke-Wei. A gamificação pode motivar contribuições voluntárias? O caso da comunidade de perguntas e respostas do StackOverflow. In: PROCEDIMENTOS da 18ª Conferência ACM sobre trabalho cooperativo suportado por computador e computação social. [Sl: sn], 2015. p. 171-174.
- CHANG, Jonathan et al. Lendo folhas de chá: como os humanos interpretam os modelos de tópicos. In: PROCEDIMENTOS do Vigésimo Terceiro Avanços nos sistemas de processamento de informação neural. [Sl: sn], 2009. p. 288-296.
- DAUD, Ali. Usando modelagem de tópicos de tempo para descoberta de interesse de pesquisa dinâmica baseada em semântica. **Sistemas Baseados em Conhecimento**, Elsevier, v. 26, p. 154-163, 2012.
- FITELSON, Branden. Uma teoria probabilística da coerência. **Análise**, JSTOR, v. 63, n. 3, pág. 194-199, 2003.
- HAN, Pu et al. A influência da normalização de palavras no agrupamento de documentos em inglês. Em: IEEE. 2012 IEEE Conferência Internacional de Ciência da Computação e Engenharia de Automação (CSAE). [Sl: sn], 2012. v. 2, p. 116-120.

JIVANI, Anjali Ganesh et al. Um estudo comparativo de algoritmos de stemming. **Int. J. Comp. Tecnologia Appl**, v. 2, n. 6, pág. 1930-1938, 2011.

LANDAUER, Thomas K; FOLTZ, Peter W; LAHAM, Darrel. Uma introdução à análise semântica latente. **Processos discursivos**, Taylor & Francis, v. 25, n. 2-3, pág. 259-284, 1998.

LAU, Jey Han; NEWMAN, David; BALDWIN, Timóteo. Folhas de chá de leitura automática: avaliando automaticamente a coerência do tópico e a qualidade do modelo de tópico. Em: PROCESSOS DO 14ª Conferência do Capítulo Europeu da Association for Computational Linguistics. [Sl: sn], 2014. p. 530-539.

MANNING, Christopher D; SCHÜTZE, Hinrich. **Fundamentos do processamento estatístico de linguagem natural**. [Sl]: MIT Press, 1999.

MCCALLUM, Andrew Kachites. Mallet: Um kit de ferramentas de aprendizado de máquina para idiomas. <http://mallet.cs.umass.edu>, 2002.

MIKOLOV, Tomas et al. Representações distribuídas de palavras e frases e sua composicionalidade. In: AVANÇOS em sistemas de processamento de informação neural. [Sl: sn], 2013. p. 3111-3119.

MINKA, Thomas. **Estimando uma distribuição de Dirichlet**. [Sl]: Relatório técnico, MIT, 2000.

NEWMAN, David et al. Avaliação automática da coerência do tópico. In: HUMAN language technologies: A conferência anual de 2010 do capítulo norte-americano da associação de linguística computacional. [Sl: sn], 2010. p. 100-108.

PEDREGOSA, Fabian et al. Scikit-learn: Aprendizado de máquina em Python. **o Journal of Machine Learning Research**, JMLR. org, v. 12, p. 2825-2830, 2011.

RAMOS, Juan et al. Usando tf-idf para determinar a relevância da palavra em consultas de documentos. In: NOVA JERSEY, EUA. PROCEDIMENTOS da primeira conferência instrucional sobre aprendizado de máquina. [Sl: sn], 2003. v. 242, p. 133-142.

ŸEHŸŸEK, Radim; SOJKA, Petr. Framework de Software para Modelagem de Tópicos com Grandes Corpora. Inglês. In: ANAIS DO WORKSHOP LREC 2010 SOBRE NOVOS DESAFIOS PARA FRAMEWORKS DE PNL. Valletta, Malta: ELRA, maio de 2010. p. 45-50. <http://is.muni.cz/publication/884893/en>.

RODER, Michael; AMBOS, Andréa; HINNEBURG, Alexandre. Explorando o Espaço das Medidas de Coerência de Tópicos. In: Anais da Oitava ACM International Conference on Web Search and Data Mining. Xangai, China: Association for Computing Machinery, 2015. p. 399-408. ISBN 9781450333177. DOI: 10.1145/2684822.2685324.

ROSEN-ZVI, Michal et al. O modelo autor-tópico para autores e documentos. **arXiv pré-impressão arXiv:1207.4169**, 2004.

STEYVERS, Mark; GRIFFITHS, Tom. Modelos de tópicos probabilísticos. **Manual de análise semântica latente**, v. 427, n. 7, pág. 424-440, 2007.

SYED, Shaheen; SPRUIT, Marco. Texto completo ou resumo? Examinando pontuações de coerência de tópicos usando alocação de dirichlet latente. Em: IEEE. 2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA). [SI: sn], 2017. p. 165-174.

TAN, Chade-Meng; WANG, Yuan-Fang; LEE, Chan-Do. O uso de bigramas para melhorar a categorização de texto. **Processamento e gestão da informação**, Elsevier, v. 38, n. 4, pág. 529-546, 2002.

TANG, Jie; ZHANG, Jing. Modelagem da evolução dos dados associados. **Engenharia de Dados e Conhecimento**, Elsevier, v. 69, n. 9, pág. 965-978, 2010.

WALLACH, Hanna M et al. Métodos de avaliação para modelos de tópicos. In: PROCEDIMENTOS da 26ª Conferência Internacional Anual de Aprendizado de Máquina. [SI: sn], 2009. p. 1105-1112.

WANG, Shaowei; LO, Davi; JIANG, Lingxiao. Um estudo empírico sobre interações de desenvolvedores em stackoverflow. In: PROCEDIMENTOS do 28º Simpósio Anual ACM de Computação Aplicada. [SI: sn], 2013. p. 1019-1024.

WANG, Xuerui; MCCALUM, André. Tópicos ao longo do tempo: um modelo de tempo contínuo não-Markov de tendências tópicas. In: PROCEDIMENTOS da 12ª Conferência Internacional ACM SIGKDD sobre Descoberta de Conhecimento e Mineração de Dados. [SI: sn], 2006. p. 424-433.

XU, Shuo; SHI, Qingwei; QIAO, Xiaodong; ZHU, Lÿun; JUNG, Hanmin, et ai. Author-Topic over Time (AToT): um modelo dinâmico de interesse dos usuários. In: Computação MÓVEL, ubíqua e inteligente. [SI]: Springer, 2014. p. 239-245.

XU, Shuo; SHI, Qingwei; QIAO, Xiaodong; ZHU, Lÿun; ZHANG, Han, et ai. Um modelo dinâmico de descoberta de interesses de usuários com algoritmo de inferência distribuído. **International Journal of Distributed Sensor Networks**, Publicações SAGE Sage UK: Londres, Inglaterra, v. 10, n. 4, pág. 280892, 2014.

YANG, Min et al. Descobrindo a evolução do interesse do autor na modelagem de tópicos sensível à ordem e ciente da semântica. **Ciências da Informação**, Elsevier, v. 486, p. 271-286, 2019.

APÊNDICE A – TÓPICOS E PALAVRAS PRINCIPAIS DO ESTÁGIO DE PILHA

Etiqueta de identificação	Principais palavras LDA
0 Desenvolvimento Android	<i>botão clique na página de visualização do android jquery selecione a imagem do item javascript</i>
1 banco de dados relacional	<i>tabela sql consulta coluna banco de dados linha mysql server selecione id</i>
2 Manipulação de strings	<i>string python array list number caractere corresponde à linha de saída regex</i>
3 E/S Python	<i>comando de string de linha python script caractere executar gravação do programa de saída</i>
4 Tipos de dados	<i>tipo de tempo número modelo de caso exemplo desempenho do objeto ponto de interrogação</i>
5 consultas SQL	<i>tabela de matrizes consulta sql coluna string lista de retorno linha php</i>
6 Integração Android	<i>aplicativo de mensagem android enviar tempo do aplicativo do servidor executar processo de encadeamento</i>
7 Rede de servidores	<i>servidor executar teste thread serviço aplicativo cliente mensagem de conexão de mola</i>
8 Autenticação	<i>user api request google http token post url de acesso ao facebook</i>
9 Estúdio Android	<i>android xml view lista de layout item de atividade fragmento de texto json</i>
10 Roteamento da Web	<i>javascript html jquery página http navegador carregar evento de script chrome</i>
11 Webservice	<i>servidor web http net serviço usuário aplicativo pedido página com</i>
12 Estrutura do Django	<i>python test run script django ruby command linha do módulo ruby_rail</i>
13 Implantação do projeto	<i>projeto executar versão do servidor de compilação serviço da web da janela http do aplicativo</i>
14 Configuração móvel	<i>aplicativo android google application device com ios window http user</i>
15 Desenvolvimento iOS	<i>ios image swift xcode iphone app objetivo janela do google map</i>
16 Webdesign	<i>imagem css html conjunto de elementos de texto jquery javascript color div</i>
17 Controle de versão	<i>git test object type branch version commit novo repositório de casos</i>
18 Tipo de chave de objeto de banco de dados não relacional banco de dados mongodb instância de modelo de caso de exceção de retorno	
19 Estrutura angular	<i>jquery html javascript css botão componente angular clique no elemento da página</i>
20 Formato de dados relacionais	<i>data php hora consulta sql formato do servidor de tabela de banco de dados mysql</i>
21 Visualização de dados	<i>array número ponto loop elemento de linha excel exemplo de plotagem python</i>
22 Manipulação de imagem	<i>elemento de texto de linha de imagem python definir tamanho de exemplo html css</i>
23 Gestão de eventos	<i>conjunto de javascript de thread de controlador de objeto de propriedade de controle de exibição de rede de evento</i>
24 estrutura .NET	<i>net user form página asp mvc controller view model session</i>
25 Arquitetura do projeto	<i>pasta php do projeto execute o diretório http versão com caminho do servidor</i>
26 solicitação PHP	<i>php user io post página do produto api json http form</i>
27 Resposta do objeto	<i>net type objeto json asp parâmetro de propriedade xml template return</i>
28 Desempenho	<i>tempo número python memória coluna resultado loop linha lista tamanho</i>
29 Referência de memória	<i>objeto variável tipo ponteiro retornar matriz referência memória programa caso</i>

APÊNDICE B - TÓPICOS DE EXCESSO DE PILHA E POPULARIDADE DE TÓPICOS DE TENDÊNCIA MEDIDAS

Etiqueta de identificação	Linha de evolução da tendência de popularidade do tópico	
0 Desenvolvimento Android	0,0261 / 2,61% -	
1 Banco de dados relacional	0,0392 / 3,92% -	
2 Manipulação de strings	0,0527 / 5,27% ÿ	
3 E/S Python	0,0354 / 3,54% -	
4 Tipos de dados	0,0279 / 2,79% ÿ	
5 consultas SQL	0,0432 / 4,32% ÿ	
6 Integração Android	0,0239 / 2,39% -	
7 Rede de servidores	0,0314 / 3,14% -	
8 Autenticação	0,0313 / 3,13% ÿ	
9 Estúdio Android	0,0273 / 2,73% ÿ	
10 Roteamento da Web	0,0360 / 3,60% ÿ	
11 WebService	0,0261 / 2,61% -	
12 Estrutura do Django	0,0348 / 3,48% -	
13 Implantação do projeto	0,0422 / 4,22% -	
14 Configuração móvel	0,0301 / 3,01% ÿ	
15 Desenvolvimento iOS	0,0290 / 2,90% ÿ	
16 Webdesign	0,0395 / 3,95% -	
17 Controle de versão	0,0303 / 3,03% ÿ	
18 Banco de dados não relacional	0,0266 / 2,66% ÿ	
19 Estrutura angular	0,0498 / 4,98% ÿ	
20 Formato de dados relacionais	0,0326 / 3,26% ÿ	
21 Visualização de dados	0,0337 / 3,37% ÿ	
22 Manipulação de imagem	0,0236 / 2,36% ÿ	
23 Gestão de eventos	0,0275 / 2,75% ÿ	
24 estrutura .NET	0,0297 / 2,97% ÿ	
25 Arquitetura do projeto	0,0332 / 3,32% -	
26 solicitação PHP	0,0271 / 2,71% ÿ	
27 Resposta do objeto	0,0338 / 3,38% ÿ	
28 Desempenho	0,0323 / 3,23% ÿ	
29 Referência de memória	0,0421 / 4,21% ÿ	