Instituto de MATEMÁTICA E ESTATÍSTICA UFRGS

DEPARTAMENTO DE ESTATÍSTICA
E-mail: dest@mat.ufrgs.br

Trabalho de Conclusão de Curso

Aprendizado não-supervisionado para textos curtos

Gustavo Machado Utpott

18 de maio de 2022

Gustavo Machado Utpott

Aprendizado não-supervisionado para textos curtos

Este Trabalho foi julgado adequado para obtenção dos créditos da disciplina Trabalho de Conclusão de Curso em Estatística e aprovado em sua forma final pela Orientadora e pela Banca Examinadora.

Orientadora:_____
Profa. Dra. Márcia Helena Barbian, UFMG
Doutora pela Universidade Federal de Minas Gerais, Belo Horizonte, MG

Banca Examinadora:

Prof. Dr. Fabricio Murai Ferreira,
Ph.D. pela University of Massachusetts Amherst

Prof. Dr. João Henrique Ferreira Flores, UFRGS
Doutor pela Universidade Federal do Rio Grande do Sul, RS

Porto Alegre
Maio de 2022

"Life moves pretty fast. If you don't stop and look around once in a while, you could miss it.". (Ferris Bueler)

Agradecimentos

Aos meus pais Gilberto e Stela pelo apoio e carinho.

À minha irmã Nicole pela inspiração e parceria.

Aos meus amigos que me acolheram na faculdade Rafaela e Rafael.

Aos meus amigos do grupo "caótico" Guilherme, Gabriel, Frederico e Bruno.

À minha orientadora não só do TCC mas de toda vida acadêmica Márcia.

Aos professores do Departamento da Estatística que contribuíram na minha formação acadêmica e profissional.

Aos professores da banca pela disposição.

Aos demais amigos e familiares que me auxiliaram nessa jornada.

Resumo

Com a evolução da tecnologia na área da comunicação, quantidades enormes de textos têm sido escritas e compartilhadas em diversas plataformas ao longo da internet, levando a uma demanda crescente de algoritmos de Processamento de Linguagem Natural (NLP). Os objetivos das análises são diversos e buscam desde a identificação de *spams*, tradução ou classificação de textos a análise de sentimentos. Dentre esses temas, descobrir tópicos de documentos de textos que não possuem uma classificação prévia tornam-se cada dia mais comuns, tais métodos, denominados Modelos de Tópicos são definidos como uma classe de algoritmos de Aprendizado não Supervisionado. Especificamente, documentos que possuem uma quantidade limitada de caracteres, os textos curtos, necessitam de metodologias diferentes daquelas comumente aplicadas, como o conhecido algoritmo *Latent Dirichlet Allocation* (LDA). O presente trabalho visa aplicar uma dessas técnicas, o *Biterm Topic Modeling* (BTM), em uma base de dados composta por descrições de diferentes mercadorias para que, após o agrupamento, seja possível selecionar os tópicos com mais semelhança a um dado produto de interessse. Além da aplicação do BTM à base, será proposto um algoritmo para substuição de abreviações contidas nos documentos a serem analisados.

Palavras-Chave: Aprendizado não supervisionado, Modelagem de tópicos, Processamento de linguagem natural, Textos Curtos, Biterm Topic Modeling.

Abstract

With the evolution of technology in the field of communication, huge quantities of text are being written and shared in a lot of platforms across the internet, leading to an increasing demand for Natural Language Processing (NLP) techniques. The goals of the analysis are plenty and go from spam identification, text translation and classification to sentiment analysis. Among those themes, uncovering topics in text that doesn't have any kind of previous classification has become more common. Those methods are named Topic Modeling and are defined as an Unsupervised Learning class of algorithms. Specifically, documents that have a limited amount of characters, short texts, need different methods to those commonly applied, such as the famous Latent Dirichlet Allocation (LDA). This work aims to apply one of these techniques which is called Biterm Topic Modeling (BTM), in a database made of different merchandise to, after the clustering, be able to select the most similar topics to a given product of interest. Besides the application of BTM to the data, an algorithm will be proposed to replace the abbreviations contained on the document being analysed.

Keywords: Unsupervised Learning, Topic Modeling, Natural Language Processing, Short Texts, Biterm Topic Modeling.

Sumário

Lista de Figuras

Lista de Tabelas

1 Introdução

Com o advento da internet e das redes sociais a comunicação entre os seres humanos evoluiu substancialmente. Uma quantidade enorme de textos são escritos e compartilhados diariamente em diversas plataformas, tornando-se expressivo o interesse em buscar, analisar e compreender esses dados, a fim de oferecer uma melhor assistência na tomada de decisão em um ambiente *data-driven*.

Nesse contexto, observa-se uma necessidade crescente da utilização e desenvolvimento de técnicas de processamento de linguagem natural (NLP em inglês). Essas técnicas tem como foco tanto a análise da relação entre os diferentes usuários que fomentam as redes, como também a compreensão desse novo tipo de linguagem escrita, o que pode trazer diferentes *insights*, que buscam aprimorar a criação de soluções que antes não seriam possíveis. Além do que, as técnicas de NLP se beneficiaram do avanço da capacidade computacional, visto que com o desenvolvimento de novas tecnologias é possivel aplicar modelos mais complexos, com grande quantidade de parâmetros ajustados à extensas bases de dados, que segundo (Russell e Norvig, 2010), é um fator fundamental para o bom desempenho dessas análises.

Dentre as técnicas de NLP, a área de aprendizado supervisionado aborda problemas como os de análise de sentimentos, filtro de *spams* e desenvolvimento de assistentes virtuais. Nesses casos, os dados são rotulados, ou seja, possuem uma variável resposta indicando, por exemplo, se a avaliação (texto do *review*) de um cliente é positiva ou negativa. A partir desse rótulo, os modelos aprendem a relacionar as variáveis preditoras (texto do *review*) com a variável resposta, obtendo-se um modelo treinado para a tarefa de predizer dados futuros.

Em particular, alguns estudos recentes da área têm analisado a classificação não supervisionada de documentos, desenvolvendo técnicas de clusterização que almejam descobrir tópicos latentes aos quais esses documentos pertenceriam. Tais técnicas são denominadas modelos de tópicos, dentre os quais pode-se destacar o *Probabilistic latent semantic analysis* (PLSA) (Hofmann, 1999) e o mais popularmente conhecido *Latent Dirichlet Analysis* (LDA) Blei et al. (2003).

Esses modelos de tópicos são muito utilizados na classificação de textos como artigos, livros e reportagens. Um exemplo de aplicação é dado por Feuerriegel et al. (2016), que analisa o impacto de notícias do mercado financeiro nas ações da bolsa de valores. Outra perspectiva é quando se trabalha com documentos de textos curtos, que têm se tornado uma tarefa em crescente relevância dado a sua popularidade na *web*, principalmente com o surgimento das redes sociais (Jelodar et al., 2019). Alguns dos problemas relacionados a textos curtos são a análise de manchetes de notícias, descrições de produtos e postagens em redes sociais como *tweets*.

Apesar do crescente desenvolvimento de metologias, a análise do cenário de textos curtos ainda é um desafio. Técnicas convencionais de modelagem de tópicos, como o LDA, não costumam retornar resultados tão interessantes (Yan et al., 2013), visto que muitos desses métodos representam cada documento como um vetor de tamanho n, onde cada elemento desse vetor indica o peso de uma palavra dentro daquele documento (representação *bag-of-words*). Isso, no contexto de textos curtos, gera uma matriz muito esparsa, dificultando a identificação dos diferentes tópicos, como também mencionado pelos autores Hong e Davison (2010) que buscam solucionar esse problema agregando os documentos no treinamento do modelo.

Considerando o desafio de identificar tópicos latentes em documentos com poucas palavras, ou seja, textos curtos, os autores Yan et al. (2013) criaram o método *Biterm Topic Model* (BTM). Tal metodologia identifica os tópicos através da modelagem da co-ocorrência global das palavras em todo o *corpus* (conjunto de documentos sobre determinado assunto). Portanto, busca superar o problema de esparsidade na representação dos documentos dos métodos referenciados anteriormente.

Existem outras formas de lidar com esse problema. Mais recentemente houve muitas inovações na área de representação de dados de texto. *Word embeddings* como o word2vec (Mikolov et al., 2013) foram um marco na evolução para essa área, transicionando do mais tradicional *bag of words* que representava os textos como uma matriz de frequência de termos por documento, para a representação de palavras como vetores de altas dimensões. Outra representação que tem se destacado é a feita por Devlin et al. (2018), que representa as palavras de formas diferentes dependendo do contexto delas. Tais métodos não fazem parte do escopo desse trabalho, e poderiam ser abordados em trabalhos futuros.

Para exemplificar o problema que será abordado no decorrer do trabalho, imagine que uma grande loja do varejo possua um *marketplace* (site de vendas online) em que diferentes comerciantes divulgam seus produtos. Essa empresa de varejo possui muitos vendedores cadastrados, com preços muito diferentes para produtos similares ou iguais, o que torna a busca de um comprador em potencial mais difícil e demorada. Logo, a empresa decide divulgar somente os produtos de lojas que tenham um valor compatível com o mercado, que não sejam muito acima de um valor médio (produtos superfaturados) nem muito abaixo (possíveis fraudes). Como fazer esse cálculo? Visto que há uma infinidade de possíveis produtos que apesar de serem muito similares, são descritos de forma diferente. Uma alternativa é utilizar a descrição do produto, agrupando mercadorias semelhantes e assim calcular um preço médio para aquele item. O agrupamento seria realizado por técnicas de NLP.

Observe que o objetivo final do problema é calcular o preço, mas para que isso seja possível, é necessário antes agrupar as mercadorias. Nesse trabalho o enfoque será: como agrupar esses produtos dado a sua descrição? Como, dentre milhões de transações que ocorrem no mercado, selecionar de forma não supervisionada, somente as descrições que representem um produto específico.

Os dados analisados na aplicação desse trabalho são referentes a notas fiscais de diferentes produtos e não possuem rótulos ou qualquer estrutura de classificação prévia. O *corpus* é composto pelas descrições textuais desses produtos, que não possuem uma padronização da escrita, a única formatação do documento é o limite de caracteres. É muito comum encontrar erros ortográficos ou de digitação no texto, além de diversas abreviações. Essas características tornam a etapa de pré processamento dos documentos uma tarefa ainda mais relevante na construção das

análises.

O objetivo principal deste trabalho é aplicar o BTM em uma base de dados com descrições de produtos para identificar os tópicos latentes ao *corpus* e selecionar, de forma automática, documentos que sejam mais semelhantes. Dessa forma, a base de dados final será mais consistente e específica. Outros objetivos específicos são:

- Pré-processar o corpus de modo automatizado para cada dado produto de interesse.
- Desenvolver um método automático de substituição de abreviações.
- Desenvolver um código que aplique o método BTM de forma automática, selecionando o número de tópicos de forma a otimizar a performance do modelo.
- Selecionar o(s) tópico(s) mais similar(es) com cada produto de interesse para se obter os documentos mais semelhantes àquele produto.

Para ler e manipular o banco de dados, assim como para modelar e avaliar os resultados será utilizado o *software* **R** (R Core Team, 2022).

2 Referencial Teórico

Nesse capítulo será feita uma breve introdução sobre métodos de aprendizado não-supervisionado e o tipo de problema que ele trata. Após, será apresentada a distribuição de Dirichlet, comumente utilizada na modelagem de textos, ao final da seção serão abordados os modelos LDA e BTM, representantes da análise de tópicos latentes em dados do tipo *string*.

2.1 Aprendizado não supervisionado

Dentro do campo de *Machine Learning* é comum que os problemas se encaixem em duas categorias (James et al., 2013): aprendizado supervisionado ou não supervisionado.

No aprendizado supervisionado, há uma medida de resposta associada a cada observação no banco de dados e busca-se modelar a relação das variáveis preditoras com a variável resposta. Na área da predição, o interesse é prever valores futuros com mais acurácia, no campo da inferência, é entender e interpretar a relação entre as variáveis preditoras e a resposta (James et al., 2013).

Já na aprendizagem não supervisionada, não há rótulos ou uma variável resposta para os dados. Em consequência disso, o que busca-se analisar nesses casos são associações ou padrões entre as observações, para assim identificar grupos de observações semelhantes (Johnson et al., 2014). Um exemplo de aplicação é a identificação de diferentes grupos de clientes de uma loja, a partir de informações como idade, gênero e renda. Com essa informação, é possível adotar diferentes estratégias de marketing para grupos específicos de consumidores, aumentando a eficiência da campanha e otimizando os recursos da empresa.

2.2 Distribuição de Dirichlet

A distribuição de Dirichlet $Dir(\boldsymbol{\alpha})$ é uma família de distribuições de probabilidade contínuas e multivariadas parametrizada pelo vetor $\boldsymbol{\alpha}$ real e não-negativo. É uma generalização da distribuição Beta (Kotz et al., 2004) e é conjugada da distribuição multinomial. Essa propriedade torna comum o uso da distribuição de Dirichlet como *priori* em inferência bayesiana e particularmente na modelagem de tópicos.

A função densidade de probabilidade da distribuição de Dirichlet($\boldsymbol{\alpha}$) de ordem $k \geq 2$ é dada por:

$$P(x_1, ..., x_k|\alpha_1, ..., \alpha_k) = P(\mathbf{x}|\boldsymbol{\alpha})\frac{1}{B(\boldsymbol{\alpha})}\prod_{i=1}^{k} x_i^{\alpha_i - 1}, \tag{2.1}$$

em que $x_i \geq 0$, $\sum_{i=1}^{k} x_i = 1$.

A constante de normalização $B(\boldsymbol{\alpha})$ pode ser expressa em termos da função gamma:

$$B(\boldsymbol{\alpha}) = \frac{\prod_{i=1}^{k} \Gamma(\alpha_i)}{\Gamma(\sum_{i=1}^{k} \alpha_i)}, \tag{2.2}$$

O suporte da distribuição 2.1 é definido pelo vetor $\mathbf{x}$ de dimensão k, composto de números reais inseridos no intervalo $(0, 1)$. Esse vetor de parâmetros pode ser interpretado como as probabilidades de uma variável categórica assumir cada uma de suas k categorias.

Um caso especial da distribuição de Dirichlet é observada quando o vetor $\boldsymbol{\alpha}$ possui todos os seus k valores iguais, resultando na chamada distribuição de Dirichlet simétrica. Nesse caso, nenhuma das k categorias possui maior probabilidade de ser observada, o que é uma característica comum quando não há nenhum conhecimento *a priori*. As magnitudes dos valores de $\boldsymbol{\alpha}$ irão direcionar a concentração do vetor $\mathbf{x}$, para $\boldsymbol{\alpha} < 1$ a distribuição é mais esparsa, com maiores probabilidades concentradas em valores mais próximos do vetor $\mathbf{0}$, enquanto que para $\boldsymbol{\alpha} > 1$ a distribuição fica mais densa nos valores mais similares entre si, tal característica pode ser vista na Figura (2.1).

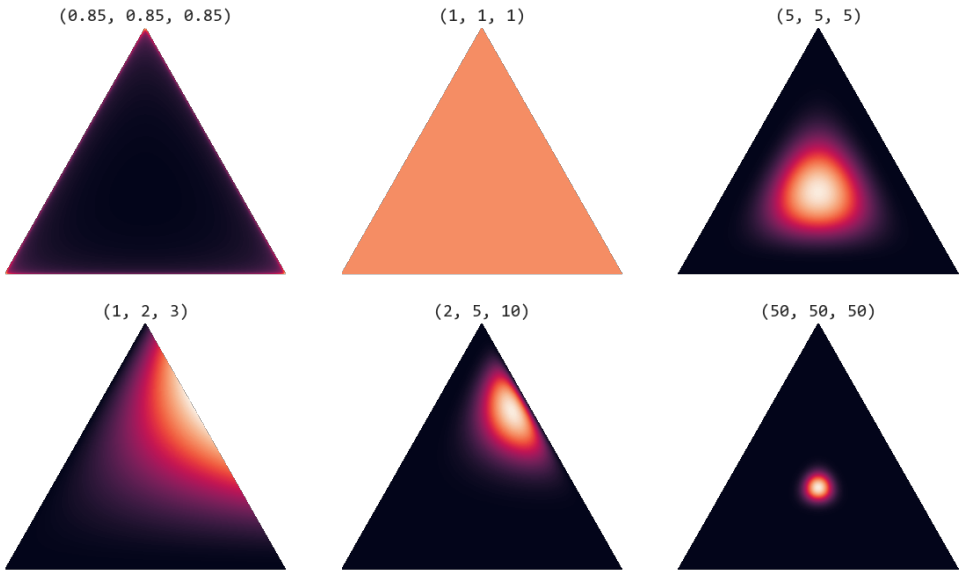


Figura 2.1: Distribuição de Dirichlet com $k = 3$, variando valores de $\boldsymbol{\alpha}$.
Fonte: https://github.com/yusueliu/medium/blob/master/scripts/plot_dirichlet.py

Em inferência bayesiana o objetivo da análise é estimar a distribuição de probabilidade de um parâmetro ou de um vetor de parâmetros. A estimativa é calculada através da combinação da informação *a priori* e da função de verossimilhança.

Pensando em um problema de modelagem de tópicos, cujo objetivo busca definir a qual tópico latente determinado documento pertence, pode-se representar a verossimilhança de vários documentos como a distribuição conjunta de uma Multinomial

de parâmetro ($\boldsymbol{\theta}$), onde $\boldsymbol{\theta}$ é descrito pela distribuição *a priori* $\boldsymbol{\theta} \sim Dir(\boldsymbol{\alpha})$, que representa um conhecimento prévio sobre as proporções dos tópicos nos documentos.

2.3 Modelagem de tópicos

2.3.1 Latent Dirichlet Allocation

O *Latent Dirichlet Allocation*(LDA) (Blei et al., 2003) foi publicado em 2003 e desde então é um algoritmo de referência para agrupamento de documentos de textos. LDA é um modelo probabilístico generativo, sua ideia principal é que um conjunto de M documentos são representados como uma mistura de k tópicos latentes, que, por sua vez, são caracterizados por uma distribuição de probabilidade de palavras. O LDA também pode ser visto como um modelo bayesiano hierárquico de 3 níveis, a palavra, o documento e o tópico. O modelo é formulado por meio dos seguintes termos:

- Uma *palavra* ou *token* w é uma unidade discreta, definida como um item de um vocabulário.
- Um documento é uma sequência de N_i palavras, denotadas por $\mathbf{w} = (w_1, w_2, \ldots, w_{N_i})$, onde w_n é a n-ésima palavra da sequência.
- Um *corpus* $\mathbf{D}$ é uma sequência de M documentos $D = (\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_M)$.
- z_i é a variável aleatória que indica o tópico ao qual o documento i pertence.
- $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_k)$ é o vetor das probabilidades de cada um dos tópicos observados nos M documentos do *corpus* $\mathbf{D}$, no modelo $z_i \sim$ Multinomial $(\boldsymbol{\theta})$.
- $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_k)$ é o vetor de hiperparâmetros da distribuição de probabilidade *a priori* dos k tópicos $\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_k)$ de um *corpus*, no modelo, essa distribuição será uma Dirichlet$(\boldsymbol{\alpha})$.
- w_j é a variável aleatória que indica a qual tópico pertence determinada palavra, no modelo $w_j \sim$ Multinomial $(\boldsymbol{\phi})$.
- $\boldsymbol{\beta}$ é o vetor de hiper-parâmetros da distribuição de probabilidade *a priori* das N palavras $\boldsymbol{\phi} = (\phi_1, \phi_2, \ldots, \phi_N)$ de um documento j do tópico z_i, no modelo, essa distribuição será uma Dirichlet$(\boldsymbol{\beta})$.

Na Figura (2.2) é possível ver um exemplo mais aplicado de como funcionam as diferentes probabilidades aplicadas aos tópicos e palavras. Além disso, o processo generativo para um corpus D com k tópicos e M documentos, cada um contendo N_i palavras é dado por:

1. Amostre $\boldsymbol{\theta}_i \sim Dir(\boldsymbol{\alpha})$, que é o parâmetro da multinomial que representa a distribuição de tópicos do documento i;
2. Amostre $\boldsymbol{\phi}_k \sim Dir(\beta)$, que é o parâmetro da multinomial representando a distribuição de tópico-palavras;

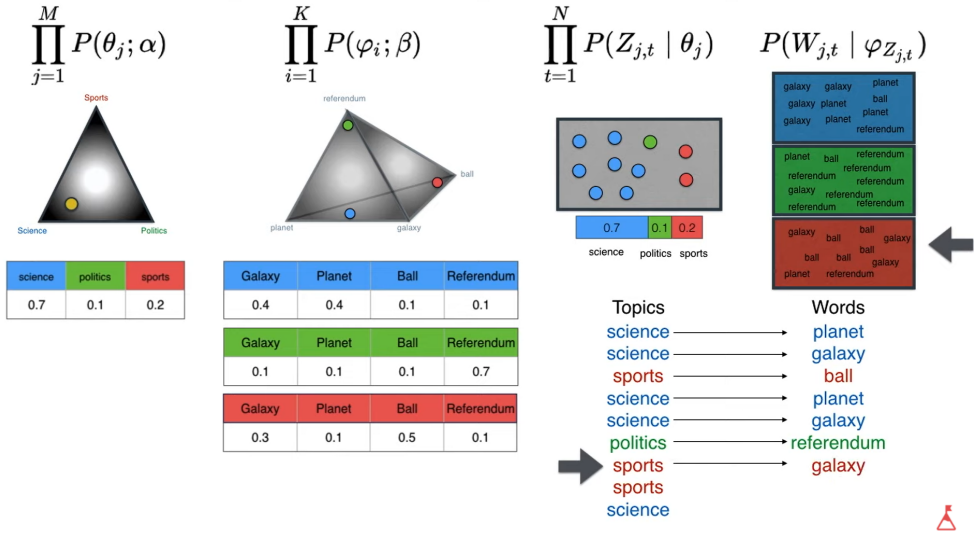


Figura 2.2: Representação gráfica de um modelo de tópico LDA

3. Para cada posição de palavra i, j, onde $i \in \{1, ..., M\}$ e $j \in \{1, ..., N_i\}$.
   - Amostre um tópico $z_{i,j} \sim Multinomial(\boldsymbol{\theta}_i)$.
   - Amostre uma palavra $w_{i,j} \sim Multinomial(\boldsymbol{\phi}_{z_{i,j}})$.

Dado os parâmetros $\boldsymbol{\alpha}$ e $\boldsymbol{\beta}$, a probabilidade conjunta da mistura de tópicos θ, um conjunto de K tópicos $\mathbf{z}$ e um conjunto de N palavras $\mathbf{w}$ são dadas por:

$$\begin{aligned} P(\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta}) &= P(\boldsymbol{\theta}|\boldsymbol{\alpha}) \prod_{n=1}^{N} P(z_n|\boldsymbol{\theta}) P(w_n|z_n, \boldsymbol{\beta}) \\ &= \prod_{i=1}^{M} P(\theta_i|\boldsymbol{\alpha}) \prod_{n=1}^{K} P(\phi_n|\boldsymbol{\beta}) \prod_{j=1}^{N_i} P(z_{i,j}|\theta_i) P(w_{i,j}|\phi_{z_{i,j}}) \end{aligned} \tag{2.3}$$

Para inferir os parâmetros $\boldsymbol{\theta}$ e $\boldsymbol{\phi}$ é utilizado frequentemente o algoritmo de Gibbs Sampling (Blei et al., 2003). Observe que o LDA pode ser visto primeiramente como uma mistura de tópicos, similarmente, um tópico é uma mistura de palavras. Se uma palavra tem alta probabilidade de pertencer a um tópico, todos os documentos com a palavra serão mais fortemente associados com esse tópico. Ao contrário, se a probabilidade da palavra w_i for baixa para o tópico, os documentos que contêm essa palavra dificilmente terão sido gerados por esse tópico.

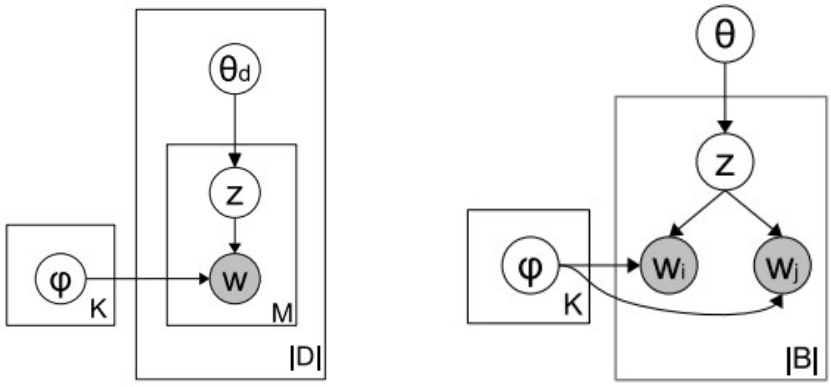
Um grande desafio na estimação do LDA é decidir em quantas categorias k os documentos podem ser divididos. Observe, também, que o LDA possui uma grande limitação, visto que não leva em consideração a ordem em que as palavras são citadas no documento.

2.3.2 Biterm Topic Model

O método desenvolvido por Yan et al. (2013) é frequentemente classificado como um STTM (*Short Text Topic Modeling*), além dele há outros métodos que se propõem a fazer uma modelagem específica para tópicos de textos curtos, uma compilação e comparação entre eles foi feita por (Qiang et al., 2020). O artigo em questão

separa os métodos em 3 categorias, são elas: mistura multinomial de Dirichlet; auto agregação e co-ocorrências globais; Sendo o BTM pertencente à categoria de co-ocorrências globais. Além disso, o *biterm* no nome do método é referente a dois termos ocorrendo de forma conjunta em uma janela de texto.

Em contraposição ao LDA, que modela diretamente a probabilidade de cada documento i pertencer a um conjunto de tópicos k, sem levar em consideração a ordem em que as palavras são mencionadas em cada um dos documentos, o BTM busca inferir a probabilidade de um documento pertencer à determinado tópico de forma indireta. Para isso o método calcula a probabilidade das co-ocorrências globais das palavras pertencentes àquele documento, como pode ser visto na Figura (2.3).



(a) LDA (b) BTM

Figura 2.3: Representação gráfica dos modelos de tópicos.

Essa diferença entre os modelos é dada pelo parâmetro $\boldsymbol{\theta}$, que no LDA é específico para cada documento i, no BTM essa característica não está presente. Além disso, o BTM leva em consideração a ordem em que as palavras são observadas no documento, pois ele calcula o que é chamado de co-ocorrências. O processo generativo do BTM pode ser definido como:

1. Amostre $\boldsymbol{\phi}_k \sim Dir(\boldsymbol{\beta})$, que é o parâmetro da multinomial representando a distribuição de tópico-palavras;
2. Amostre da distribuição de tópicos $\boldsymbol{\theta} \sim Dir(\boldsymbol{\alpha})$ para todos os documentos conjuntamente.
3. Para cada *biterm* b no conjunto B.
   - Amostrar um tópico $z \sim Multinomial(\boldsymbol{\theta})$.
   - Amostrar duas palavras: $w_i, w_j \sim Multinomial(\boldsymbol{\phi}_z)$.

Seguindo os passos acima, a probabilidade conjunta de um *biterm* $b = (w_i, w_j)$ pode ser escrita como:

$$\begin{aligned} P(b) &= \sum_{z=1}^{k} P(z)P(w_i|z)P(w_j|z) \\ &= \sum_{z=1}^{k} \theta_z \phi_{i_z} \phi_{j|z} \end{aligned} \tag{2.4}$$

A inferência sobre a probabilidade dos tópicos nos documentos é feita de forma indireta, já que em seu processo generativo o BTM não modela a geração dos documentos. Para isso, assume-se que as probabilidades dos tópicos de um documento são iguais à esperança das probabilidades de tópicos dos *biterms* pertencentes ao documento:

$$P(z|d) = \sum_b P(z|b)P(b|d). \tag{2.5}$$

O primeiro termo da equação 2.5 $P(z|b)$ pode ser obtido através dos parâmetros estimados pelo BTM:

$$P(z|b) = \frac{P(z)P(w_i|z)P(w_j|z)}{\sum_z P(z)P(w_i|z)P(w_j|z)}, \tag{2.6}$$

em que $P(z) = \theta_z$ e $P(w_i|z) = \phi_{i|z}$. Já o outro termo da equação 2.5, $P(b|d)$, é obtido simplesmente calculando a proporção dos *biterms* b no documento d:

$$P(b|d) = \frac{n_d(b)}{\prod_b n_d(b)}. \tag{2.7}$$

3 Metodologia

Neste capítulo serão abordadas as etapas para a implementação do BTM. Inicialmente haverá uma introdução ao banco de dados ao qual o método será aplicado, após serão descritas as técnicas de pré-processamento complementares ao BTM, também será apresentado o critério de ajuste do modelo e por último o *tuning* do parâmetro k, que indica o número de tópicos.

3.1 Banco de dados

Para exemplicar a aplicação do BTM foi utilizado um banco de dados composto por 20 milhões de notas fiscais eletrônicas de diferentes produtos. Os dados são confidenciais e foram disponibilizados pelo Tesouro, órgão da Secretaria da Fazenda do Estado do Rio Grande do Sul. Das informações relativas a essas transações utilizaremos apenas a variável descrição do produto, que é delimitada por um número máximo de caracteres. A união dessas 20 milhões de notas compõem o corpus completo. Na aplicação do modelo BTM, utilizou-se uma amostra reduzida, por questões de poder computacional, com 20% das notas, chegando a 4 milhões de mercadorias.

Apesar da metodologia ser aplicada a uma base de Notas Fiscais, ela também é generalizável e, portanto, aplicável a outros dados de texto e produtos como um *marketplace*. Os principais ajustes para adaptação da metodologia são referentes ao pré-processamento das descrições, onde são aplicadas técnicas mais específicas, dado as características como o tipo de preenchimento, tipos de produtos e *corpus*.

Os produtos que exemplificam a aplicação do método são relacionados ao ramo alimentício, serão abordados os resultados para dois itens específicos com descrições escolhidas pela Secretaria da Fazenda:

- Leite de Vaca Integral em Pó;
- Carne Bovina Resfriada Bifes Patinho,

as interpretações e análises de outros produtos seguem de forma similar.

3.2 Pré-processamento

O pré-processamento é uma parte essencial quando se trabalha com NLP (Kannan et al., 2014), tais técnicas almejam, de modo geral, reduzir o vocabulário de

palavras únicas do *corpus*. Isso é feito tanto pela unificação de palavras que representam o mesmo significado quanto pela remoção de palavras que não são relevantes, levando-se em conta o tipo de problema e modelo (Kannan et al., 2014). Os principais pacotes do *software* **R** utilizados nessa etapa são o *tm* (Feinerer et al. (2008)), o *stringdist* (van der Loo (2014)) e por final o *stringr* (Wickham (2019)). As etapas empregadas estão listadas a seguir:

- Transformação dos caracteres em minúsculos;
- Remoção de acentos, pontuações, números e símbolos;
- Remoção das *stopwords*[1] do português;
- Filtragem de palavras minimamente frequentes;
- Correção de palavras abreviadas e com erros de digitação.

Após o refinamento dos documentos , executa-se a tokenização do *corpus* através do pacote *udpipe* (Wijffels, 2021b). Nesse processo, cada documento é separado em palavras, os *tokens*. Um exemplo de como um documento de texto fica representado após a aplicação dos tratamentos é dado abaixo:

- Documento original

  Carne Moída Dianteiro/Bov.1KG Cong

- Documento após o pré-processamento

  carne; **moida**; **dianteiro**; **bovina**; **1kg**; **congelada**

Como o objetivo principal da análise é agrupar produtos semelhantes e o banco de dados de notas fiscais possui itens genéricos, com produtos que se estendem do ramo alimentício até eletrônicos, decidiu-se por acrescentar mais uma etapa ao pré-processamento das notas. Porque, ao buscar-se documentos referentes ao produto leite em pó, não faz sentido considerar mercadorias que não façam menção a palavra leite em sua descrição.

Adicionou-se um filtro aos documentos do conjunto total das 20 milhões de notas, esse filtro busca aquelas mercadorias que contenham pelo menos uma das palavras pertencentes às descrições dos produtos: Leite de Vaca Integral em Pó e Carne Bovina Resfriada Bifes Patinho. Com essa seleção, obtem-se um *corpus* específico para cada produto, o que também produzirá um modelo particular para cada item de interesse.

3.2.1 Abreviações

A metodologia de substituição das abreviações será destacada nessa subseção, pois foi uma das etapas mais relevantes e impactantes no pré-processamento do texto.

A partir de análises descritivas percebeu-se que em alguns casos as abreviações são mais frequentes que as palavras originais que foram abreviadas, como pode ser

[1] *stopwords* são palavras que são normalmente irrelevantes para a análise do texto, alguns exemplos são **as, os, de, para, com**

visto na Tabela 3.1. Isso gera problemas na modelagem, pois o modelo identifica essas palavras como *tokens* diferentes, por exemplo, por mais que os *biterms* mais associados a **cong** e **congelado** sejam parecidos, a co-ocorrência entre esses dois termos é muito baixa, o que pode levar o modelo a identificar essas palavras como associadas à tópicos diferentes.

Tabela 3.1: Tabela com frequências de palavras e abreviações no banco de dados.

Token	Frequência
congelado	146.525
congelada	21.520
cong	390.535
resfriado	788.392
resfriada	142.103
resf	302.959
bovina	111.817
bovino	62.637
bov	141.132

Nas abreviações apresentadas na Tabela 3.1 é bem intuitiva a identificação da palavra original a qual a abreviação está se referindo, porém há muitos casos em que a abreviação é ambígua e não se sabe ao certo a qual palavra ela se refere. Um exemplo é a abreviação **int** que pode se referir tanto ao *token* **integral** quanto ao **inteiro**.

À vista disso, decidiu-se analisar o contexto ao qual as abreviações se inseriam, mensurada por meio de uma medida de coerência comumente utilizada para analisar o desempenho de modelos de tópicos, como descrito por Röder et al. (2015). Apesar dessa métrica ser proposta para outro contexto, ela se encaixa muito bem para o problema das abreviações, pois deseja-se avaliar o quão bem uma palavra se encaixa no contexto do documento e a coerência avalia justamente isso.

A ideia principal é selecionar a palavra com a maior coerência para cada documento e avaliar se esse valor é suficiente para efetuar a troca. É importante destacar que o algoritmo não busca substituir todas as abreviações encontradas, pois quando nenhuma palavra candidata a substituição é minimamente coerente com as outras palavras do documento, a abreviação não deve ser substituída.

Inicialmente, a regra de decisão proposta para efetuar a troca ou não da abreviação foi a soma das coerências com a palavra candidata. Caso fosse positiva, era efetuada a troca da abreviação, caso contrário, o documento permanecia o mesmo. Entretanto, observou-se que esse critério é bastante conservador, pois poucas abreviações eram efetivamente substituídas. Pensando em alternativas, surgiu a ideia de explorar como os máximos das coerências se comportariam. Estudando a teoria de valores extremos em De Haan et al. (2006), resolveu-se ajustar os dados das coerências de palavras candidatas a substituição das abreviações com uma distribuição *Generalized Extreme Value* (GEV).

O valor de coerência geral para uma palavra candidata é muito dependente do número de *tokens* que o documento possui, pois é uma soma de valores que variam de -1 a $+1$, por isso decidiu-se agrupar a análise por tamanho do documento.

Para cada tamanho de descrição ajustou-se uma distribuição GEV através do pacote *extRemes* (Gilleland e Katz, 2016). Considerando a notação desse artigo, a distribuição GEV estimada pode ser de três tipos, sendo definida a partir do valor do parâmetro de forma ξ, onde para $\xi < 0$ se tem uma distribuição Weibull Reversa que é superiormente limitada, para ξ tendendo a 0, se tem uma distribuição Gumbell, e para $\xi > 0$ a distribuição de Fréchet. Essas diferenças podem ser vistas na Figura 3.1.

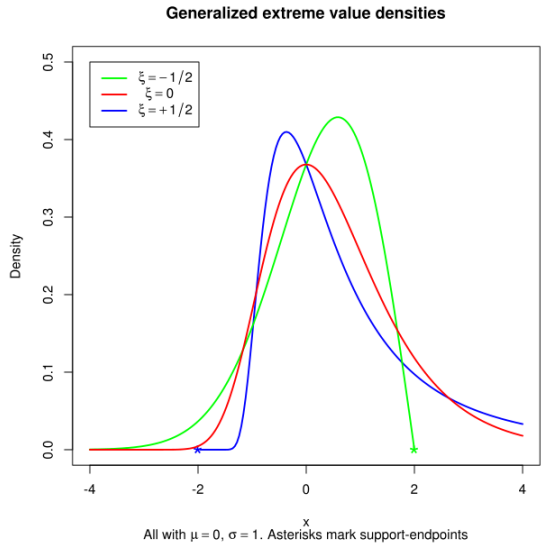
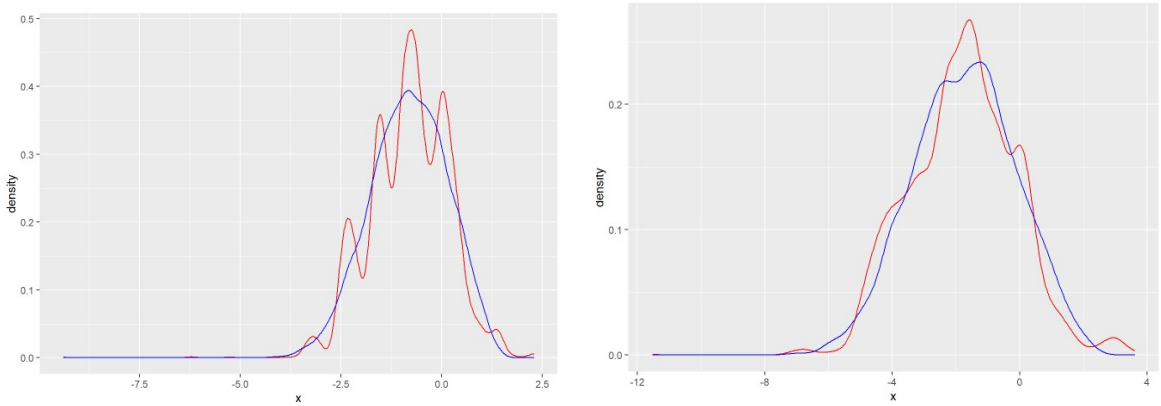


Figura 3.1: Distribuição GEV variando o parâmetro de forma ξ.

Alguns exemplos de ajustes das distribuições do máximo das coerências são dadas nas Figuras 3.2 abaixo:



(a) Descrição com 4 palavras

(b) Descrição com 8 palavras

Figura 3.2: Curva em vermelho: densidade estimada do máximo das coerências de notas fiscais com 4 e 8 palavras. Curva em azul: distribuição GEV estimada para o conjunto das coerências de notas fiscais com 4 e 8 palavras.

Ao estimar as distribuições desses valores de coerência pode-se criar estatísticas e mensurar melhor o comportamento desses dados para que, no final, se possa criar uma regra de decisão mais rebuscada e baseada na teoria dos valores extremos e nos valores estimados de ξ.

3.3 Ajuste do modelo

Após o tratamento dos dados e filtragem das notas fiscais serão ajustados os modelos BTMs específicos para cada produto através do pacote homônimo *BTM*(Wijffels (2021a)). Abaixo os argumentos mais relevantes da função que ajusta o modelo:

- *data*: Banco de dados tokenizado obtido ao final da etapa de pré-processamento, é formado por um *data.frame*.
- $\boldsymbol{\alpha}$: Hiper-parâmetro da distribuição *a priori* Dirichlet($\boldsymbol{\alpha}$) do vetor $\boldsymbol{\theta}$, que representa a probabilidade de um tópico $P(\boldsymbol{\theta}|\boldsymbol{\alpha})$. Os valores do vetor recomendados por (Yan et al., 2013) e aplicados nas análises são $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_k) = (50/k, \ldots, 50/k)$.
- $\boldsymbol{\beta}$ Hiper-parâmetro $\boldsymbol{\beta}$ da distribuição *a priori* de Dirichlet($\boldsymbol{\beta}$) para a probabilidade de uma palavra dado um tópico $P(w|z)$, os valores, também recomendados por (Yan et al., 2013) e utilizado nas análises foram $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_N) = (0.01, \ldots, 0.01)$.
- *window*: Número inteiro do tamanho da janela de contexto, os *biterms* são extraídos a partir de duas palavras que estejam dentro de uma mesma janela. Na análise, utilizou-se o *default* da função, 15 palavras. Considerando que o tipo de dado abordado nesse trabalho é composto por notas fiscais com poucos caracteres, toda a informação contida na descrição torna-se relevante, por isso decidiu-se por uma janela ampla.
- *iterations*: Número de iterações do algoritmo de Gibbs *sampling*. Nas análises é igual à 1000.
- k: Quantidade de tópicos, a escolha do valor aplicado nas análises será abordada na próxima seção.

Ao final das iterações teremos acesso às estimativas das probabilidade dos documentos pertencerem aos tópicos $P(\boldsymbol{\theta}|\boldsymbol{\alpha})$ e a probabilidade de uma palavra específica pertencer à um tópico $P(z|w)$. Para alocar os documentos aos tópicos é necessário selecionar aquele com a maior $P(\theta)$ para cada documento, formando-se assim os *clusters* do BTM.

3.4 *Tuning* do parâmetro k

Em aprendizado não supervisionado é comum que para aplicação de determinado modelo seja necessário definir algumas quantidades, por exemplo, o número de *clusters* no método *k-means* (Johnson et al., 2014). No BTM, definir a quantidade de tópicos latentes k que geraram os documentos observados no *corpus* também é um argumento necessário e assim como em qualquer análise de agrupamento, não é uma tarefa trivial. Essa medida influencia fortemente os resultados das análises e é crucial nas estimativas do BTM. Se o número de tópicos escolhido for muito alto, o modelo acabará por identificar documentos semelhantes como sendo de grupos diferentes. Se o valor for baixo, alguns documentos, com características extremamente diferentes poderão ser considerados como gerados pelo mesmo processo latente.

Uma proposta de estimativa para o valor de k é calcular o BTM para diversos modelos de forma independente, em que cada uma dessas análises considera um valor diferente para k. Para cada modelo é registrado o valor da coerência das principais palavras de cada tópico.

Esse tipo de medida é costumeiramente utilizada para analisar o desempenho de modelos de tópicos, como descrito por Röder et al. (2015). Os autores fazem uma comparação de diferentes medidas de coerência, correlacionando essas quantidades e o senso humano de interpretabilidade. A medida utilizada nesse trabalho foi a NPMI (*Normalized pointwise mutual information*), que mostrou alta correlação com *ratings* humanos (Röder et al., 2015). No trabalho de Tran e Truong (2019) há uma análise detalhada de uma aplicação do BTM. O NPMI de duas palavras é dado por:

$$\text{NPMI}(w_i, w_j) = \frac{\log \frac{P(w_i,w_j)+\epsilon}{P(w_i)*P(w_j)}}{-\log(P(w_i, w_j) + \epsilon)} \tag{3.1}$$

O valor da Equação 3.1 varia entre -1 e 1. Os valores próximos de 1 são observados quando as palavras ocorrem muito conjuntamente, enquanto que palavras que são frequentes mas raramente coocorrem tem valores mais próximos de -1. Para o cálculo da co-ocorrência o *corpus* das notas fiscais foi ampliado, com outros *corpus* em português disponíveis na web, entretanto, os resultados foram desanimadores. Suspeita-se que o motivo seja a origem dos documentos do *corpus* complementar, compostos por reportagem e textos longos, não por mercadorias ou por dados de comércio.

O cálculo da coerência geral de um modelo é a soma dos NPMI dos pares de principais palavras de cada tópico, para depois calcular uma média. Essa média de coerência para cada modelo variando os k's de cada produto é apresentada na Figura (3.3). Também foram calculados os NPMIs para várias palavras principais, utilizando uma lista de *top words*, com 5, 10, 15 e 20 palavras mais relevantes para cada tópico.

Como dito anteriormente, o valor da coerência geral do modelo varia bastante dado o número de tópicos, não seguindo uma tendência clara, como pode ser visto na Figura (3.3). Por isso, buscou-se um ponto de inflexão das coerências médias, onde há um pico alto de coerência seguido de uma queda, método que também foi utilizado no artigo de Tran e Truong (2019). Dessa forma, os valores selecionados para os dois produtos foram: $k = 4$ para **leite vaca integral em pó** e e $k = 9$ para **carne bovina resfriada bifes patinho**.

Após a etapa de *tuning* do parâmetro k, ajustou-se novamente o BTM, porém agora utilizando-se da totalidade dos dados que a capacidade computacional utilizada permitiu, o que equivale a cerca de 4 milhões de documentos de texto.

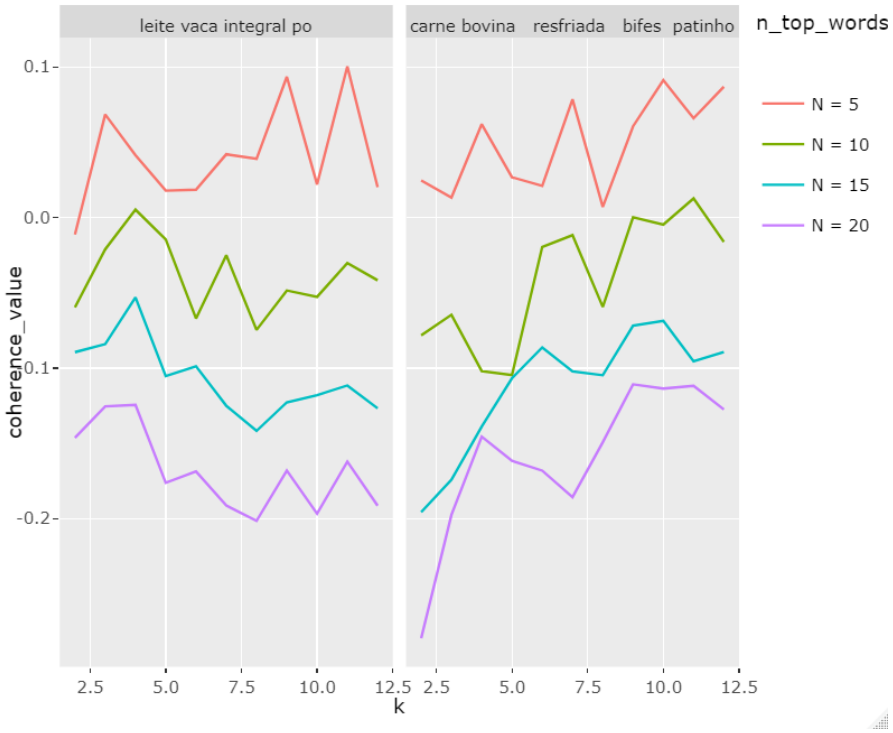


Figura 3.3: Valores de coerência NPMI para cada produto variando o k.

4 Resultados

Os resultados do ajuste do BTM a cada um dos produtos serão apresentados a seguir. Foram calculadas as coerências e gerados gráficos que representam os diferentes agrupamentos dos tópicos. As funções do pacote *LDAvis* (Sievert e Shirley, 2015) e do *App Shiny* (Chang et al., 2021) foram utilizadas para a confecção dos gráficos.

4.1 Coerência

Nas Tabelas 4.1 e 4.2 é possível observar a coerência dos tópicos com relação às suas N palavras mais frequentes. Esse *score* de coerência sumariza o quão semanticamente relacionadas estão as principais palavras daquele tópico. Portanto, quanto maior a coerência, melhor é o tópico formado. As tabelas estão ordenadas de forma decrescente.

Tabela 4.1: Tabela com coerências para o modelo do produto **leite**.

Tópicos	N			
	5	10	15	20
Tópico Leite	0,255	0,023	0,015	-0,047
Tópico Pó	0,021	0,0448	0,008	-0,012
Tópico Derivados de Leite	-0,008	-0,005	-0,035	-0,115
Tópico Integral	0,029	-0,042	-0,102	-0,246

Os nomes dados aos tópicos serão explicados na seção de interpretação dos tópicos

Tabela 4.2: Tabela com coerências para o modelo do produto **carne**.

Tópicos	Coerência das N principais palavras			
	N = 5	N = 10	N = 15	N = 20
Tópico Carne Bovina	0,130	0,111	0,047	-0,062
Tópico Derivados de Carne	0,081	0,015	0,028	0,012
Tópico Derivados de Carne	0,017	0,038	0,031	-0,030
Tópico Ração	0,018	0,025	0,006	-0,050
Tópico Carne Suína	0,049	-0,014	-0,058	-0,079
Tópico Ração	0,301	-0,071	-0,184	-0,155
Tópico Carne Bovina	-0,050	-0,080	-0,116	-0,142
Tópico Derivados de Carne	0,085	-0,165	-0,148	-0,193
Tópico Carne de Frango	0,214	-0,160	-0,233	-0,285

Os nomes dados aos tópicos serão explicados na seção de interpretação dos tópicos

De modo geral, a coerência desses dados tende a diminuir conforme aumente-se o N, isso faz sentido, pois quanto maior o N mais palavras não tão relevantes para o tópico estão sendo selecionadas para o cálculo da coerência.

4.2 LDAvis

LDAvis (Sievert e Shirley, 2015) é um pacote do **R** de visualização de modelos de tópicos. Além do **R** a biblioteca possui implementação na linguagem Python, o *pyLDAvis*. O pacote disponibiliza dois gráficos que interagem entre si, o primeiro, à esquerda, permite visualizar a distância semântica entre os tópicos. Os tamanhos dos círculos indicam a distribuição marginal dos tópicos, onde os maiores sinalizam maior probabilidade.

Já no segundo gráfico, à direita, é apresentado um gráfico de barras com as principais palavras em ordem decrescente de relevância para um determinado tópico selecionado. O gráfico contém duas cores, a vermelha representa a frequência relativa ao tópico daquela palavra, e a barra azul representa a frequência total da palavra no banco de dados. O pacote ainda fornece uma personalização da métrica de relevância, o Λ indicado no canto superior direito. É possível selecionar uma métrica mais penalizada pela frequência absoluta de palavras (valores próximos de 1.0), ou mais penalizada pela frequência relativa da palavra naquele tópico frente a frequência total daquela palavra (valores próximos de 0).

Vale destacar a funcionalidade que há quando se passa o mouse por cima de uma palavra no gráfico da direita, que faz com que o gráfico da esquerda se ajuste a distribuição de tópicos dado aquela palavra selecionada. As Figuras 4.1 e 4.2 exibem os resultados para os dois produtos abordados no estudo.

4.3 App Shiny

O pacote *shiny* do **R** (Chang et al., 2021) auxilia o desenvolvimento de aplicativos interativos, o que facilita o compartilhamento de resultados, proporcionando uma interface interativa para o usuário sem que o mesmo precise instalar algum software

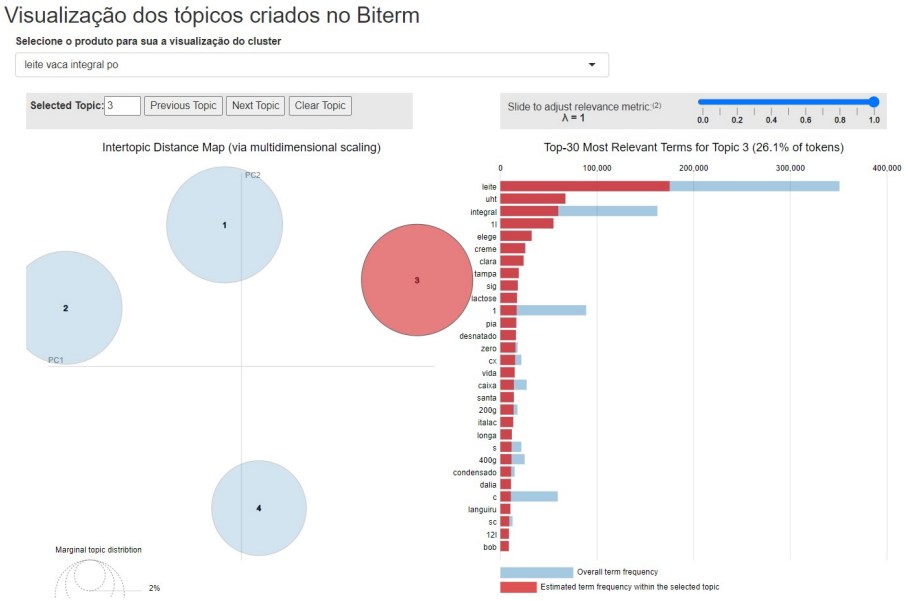


Figura 4.1: Print com o gráfico do LDAvis para o modelo do produto **leite de vaca integral em pó**.

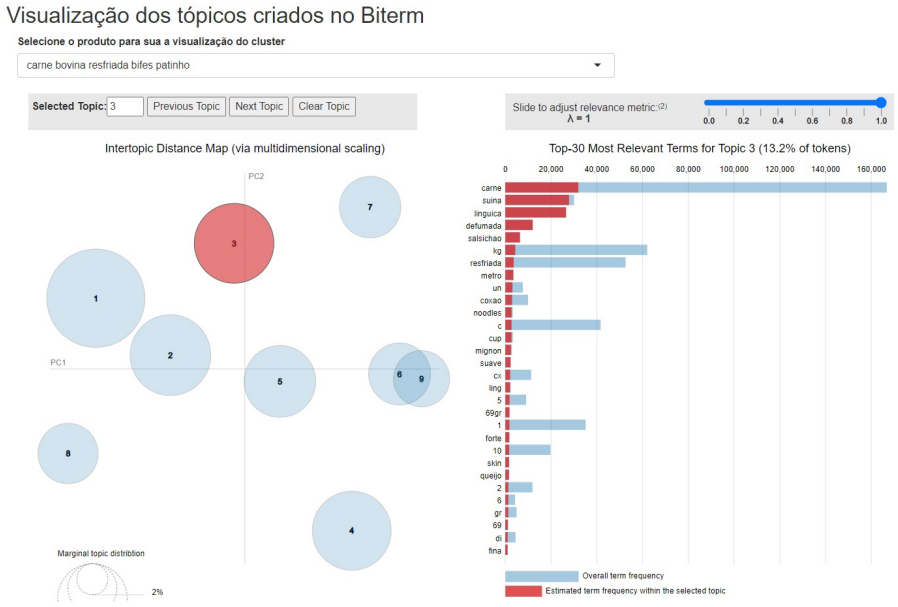


Figura 4.2: Print com o gráfico do LDAvis para o modelo do produto **Carne bovina resfriada bifes patinho**.

ou ter acesso ao banco de dados em seu próprio dispositivo. O pacote de visualização de tópicos abordado anteriormente possui integração direta com o *shiny*, promovendo uma fácil incorporação dos gráficos interativos do *LDAvis* em um aplicativo *web*.

O aplicativo com os resultados desse trabalho podem ser acessados pelo link a seguir https://gustavoutpott.shinyapps.io/tcc_biterm/. A partir da visualização dos tópicos e suas principais palavras é possível perceber como foram distribuídos os grupos de notas fiscais relacionados a cada produto.

4.4 Interpretação dos tópicos

Nesta seção serão apresentadas as palavras mais relevantes de cada tópico. Esses resultados possibilitam a interpretação dos subgrupos formados pelos diferentes produtos descritos nas notas fiscais.

4.4.1 Leite de Vaca Integral em Pó

O BTM ajustado para documentos referentes ao produto **leite de vaca integral em pó** ficou bem agrupado. Com 4 tópicos pôde-se separar de forma coerente as diferentes notas fiscais. Abaixo com o auxílio da Figura (4.3), resumiu-se as características de cada tópico formado:

(a) Tópico Leite

(b) Tópico Derivados de Leite

(c) Tópico Integral

(d) Tópico Pó

Figura 4.3: Gráficos com palavras mais relevantes dos tópicos do modelo sobre o produto **leite de vaca integral em pó**.

- Tópico Derivados de Leite

  O tópico 1 contém muitas palavras derivadas de leite, como é o caso dos produtos chocolate ao leite, doce de leite, leite de coco, leite de magnésia e etc. É o tópico com a maior probabilidade contendo 28,3% dos *tokens* (palavras).

- Tópico Pó

  O 2 é composto principalmente por produtos vendidos em pó, como produtos de limpeza, achocolatados, refrescos e etc. 26,8% dos *tokens* estão contidos

nesse tópico fazendo dele o segundo mais frequente. Observe que a quantidade de *tokens* **leite** é ínfima, indicando que esse grupo não possui produtos derivados do leite.

- Tópico Leite

  O tópico 3 contém 26,1% dos tokens e foi o mais semelhante ao produto de interesse, destacando-se os tokens **leite**, **uht**, **integral**, **desnatado**, **lactose** além de diversas marcas que são conhecidas no mercado como **elege**, **santa clara**, **pia**, **italac** entre outras. Porém, não é uma seleção perfeitamente ajustada, há documentos com as palavras creme de leite e leite condensado, indicando itens que não são similares ao produto de interesse. Palavras frequentes como **instantaneo** e marcas de leite em pó como **ninho** e **piracanjuba** estão todas alocadas à essa categoria, o que confirma a interpretação de que esse seria o tópico com documentos com descrições mais semelhante ao produto de interesse.

- Tópico Integral

  O tópico 4 foi o de menor frequência, possuindo apenas 18,9% dos tokens. O nome Integral foi dado em alusão às muitas palavras referentes à produtos como **pao**, **trigo**, **forma**, **arroz**, **suco** e etc, além de marcas relacionadas a essas mercadorias como **seven boys** e **pullman**. Esse grupo é composto por notas fiscais com as palavras **integral**, **leite** o **pó**, entretanto com uma quantidade grande de ruído, como pode ser visto na Figura (4.1).

De modo geral, os tópicos 3 e 4 são aqueles com maior similaridade com o produto de interesse.

4.4.2 Carne Bovina Resfriada Bifes Patinho

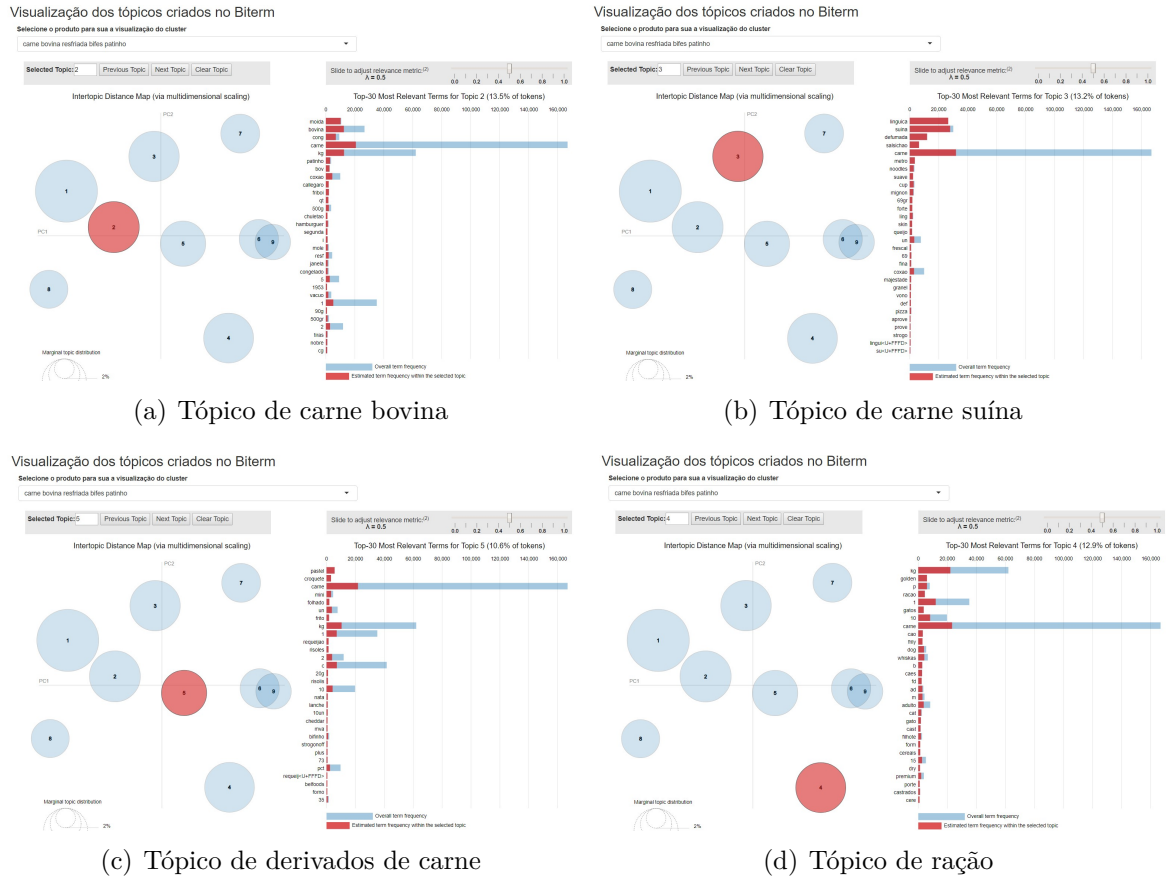
O BTM ajustado para os dados referentes ao produto de **carne bovina resfriada bifes patinho** é composto por 9 tópicos (Figura (4.4)). Pôde-se observar uma maior variabilidade do que no caso anterior, do produto de leite.

- Tópicos de carne bovina

  Os tópicos mais relacionados a carne bovina foram o 1 e o 2, pois quase todas as notas fiscais que continham os termos **bovina** e **bovino** foram alocados nesse grupo. Além disso, palavras relevantes como **carne**, **moida**, **patinho**, **costela** e **resfriada** foram muito frequentes. Uma ressalva, a palavra **suino** também possui uma frequência alta, o que indica um pouco de ruído.

- Tópicos de carne suína e de frango

  Os tópicos 3 e 8 também se qualificam como carnes de animais, porém mais focado em carnes suína e de frango respectivamente. É interessante observar que esses foram os mais próximos dos tópicos 1 e 2 (Figura (4.4)), o que faz sentido semanticamente. Dentre os *tokens* que mais se destacam há: **linguica**, **suina**, **salsichao** e **carne** para o tópíco suíno e **coxa**, **resfriada**, **asa**, **frango** para o tópico de carne de frango.



(a) Tópico de carne bovina

(b) Tópico de carne suína

(c) Tópico de derivados de carne

(d) Tópico de ração

Figura 4.4: Gráficos com palavras mais relevantes de dos tópicos do modelo do produto carne bovina resfriada bifes patinho.

- Tópicos com derivados de carne

  Os tópicos 5,7 e 9 são formados por produtos derivados da carne, ou com sabor carne. Entre as palavras que condizem com essa interpretação pode-se citar: **pastel**, **croquete**, **carne**, **lamen**, **macarrão**, **caldo** e **sopão**.

- Tópicos de rações

  Os tópicos 6 e 4 foram rotulados como de ração para animais. Palavras que se destacam são: **golden**, **racao**, **gatos**, **carne**, **cao**, **sache**, **adulto**, além de diversas marcas de ração como **pedigree**, **whiskas**, **primocao** e **primogato**.

4.5 Seleção de tópicos

Para se atingir o objetivo principal de seleção de documentos semelhantes à produtos de interesse é proposto selecionar os tópicos (gerados pelo BTM) mais relacionados aos produtos. Essa escolha pode ser realizada a partir das interpretações semânticas descritas nas subseções anteriores, analisando-se o conceito geral dos produtos agrupados em cada tópico, o que caracteriza um processo mais minucioso e manual.

Basear-se na métrica de coerência também é uma alternativa, possibilitando maior praticidade e automatização ao processo de seleção, entretanto não é o ideal,

pois um tópico mais coerente não necessariamente significa que será mais semelhante a um produto de interesse.

Buscando automatizar o processo, em um cenário com muitos produtos e necessidade de maior celeridade na obtenção dos resultados, propõem-se o uso de uma métrica baseada nos *biterms* da descrição do produto de interesse, estas combinações serão denotadas como *biterms base*. No caso do produto **Leite de Vaca Integral em Pó** formam-se os *biterms base* indicados na Tabela 4.3:

Tabela 4.3: Tabela com *biterms base* do produto **Leite de Vaca Integral em Pó**.

biterms base		
1	Leite	Vaca
2	Leite	Integral
3	Leite	Pó
4	Vaca	Integral
5	Vaca	Pó
6	Integral	Pó

Os termos **de** e **em** não foram considerados, pois são *stopwords* e, portanto, foram eliminados da base de dados.

Um próximo passo é calcular:

$$P_{bitermsbase} = \frac{\text{número de } \textit{biterms base} \text{ de } \textbf{todos} \text{ os documentos alocados ao tópico } k}{\text{número de } \textit{biterms} \text{ de } \textbf{todos} \text{ os documentos alocados ao tópico } k}.$$

Dessa forma os tópicos que possuírem mais documentos com esses *biterms base*, proporcionalmente, terão um valor maior na métrica elaborada. Segue abaixo as Tabelas 4.4 e 4.5 referentes aos modelos do produto de leite e carne, respectivamente.

Tabela 4.4: Tabela da proporção de *biterms base* para o produto de **leite de vaca integral em pó**.

Tópico	$P_{bitermsbase}$
Tópico Leite (3)	0.0176413
Tópico Integral (4)	0.012057
Tópico Derivados de Leite (1)	0.0032485
Tópico Pó (2)	0.00030230

Tabela 4.5: Tabela da proporção de *biterms base* para o produto **carne bovina resfriada bifes patinho**.

Tópico	$P_{bitermsbase}$
Tópico Carne Bovina (1)	0,0345045
Tópico Carne Bovina (2)	0,0177799
Tópico Carne Suína (3)	0,0063020
Tópico Carne de Frango (8)	0,0030856
Tópico Derivados de Carne (5)	0,000401376
Tópico Derivados de Carne (7)	0,000280264
Tópico Derivados de Carne (9)	0,00027023
Tópico Ração (4)	0,000184742
Tópico Ração (6)	0,000120281

A partir desta medida, ordena-se a relevância dos tópicos para com o produto de interesse, identifica-se um ponto de inflexão, ou seja, um valor onde haja uma queda brusca da proporção e a partir dessa quantidade cria-se um limite para seleção dos tópicos compostos por notas fiscais com descrições similares ao produto de interesse.

Dessa forma se houverem tópicos com proporções muito maiores, e portanto, muito mais relevantes, eles serão selecionados em detrimento dos outros. Caso não haja muita discrepância nas proporções, serão selecionados mais tópicos, pois significa que os possíveis produtos similares estão bem diluídos e espalhados em diferentes grupos.

Aplicando essa regra aos valores obtidos para cada produto, obtém-se a seleção dos tópicos 3 e 4 para o produto leite e os tópicos 1,2,3,8 para o produto carne.

5 Conclusão

Através da análise dos resultados descrita no capítulo anterior, conclui-se que a metodologia do BTM aplicada à base de notas fiscais é uma alternativa interessante quando o objetivo é agrupar descrições de produtos semelhantes.

Os documentos de texto filtrados a partir dos tópicos selecionados representaram uma diminuição relevante no número de mercadorias consideradas na base de dados. Eliminando-se diversos itens que não eram compatíveis com o produto a ser precificado, retornando uma base de dados mais consistente. Dependendo do tipo de produto ainda é possível tomar uma decisão mais ou menos conservadora com relação a seleção dos tópicos, caso após a filtragem de documentos ainda se tenha um banco com grande variabilidade, talvez seja de interesse aplicar uma regra de decisão mais rigorosa, selecionado-se menos tópicos, e portanto, reduzindo ainda mais o banco de dados para alcançar maior precisão, aumentando a similaridade entre os documentos selecionadas e a descrição do produto de interesse.

Apesar do BTM ter alcançado um bom resultado para análise das notas fiscais, o método ainda pode ser aprimorado, através da substituição de abreviações e da escolha automática do número de tópicos k. Como trabalhos futuros, pretende-se estudar distribuições de valores extremos, na busca de um teste para substituição ou não das abreviações. Outra possibilidade é a utilização de métodos de *word embeddings*, que talvez, possam auxiliar não só na decisão de troca de abreviações, mas também na descobertas de palavras candidatas.

Os resultados deste trabalho podem ser de interesse de *marketplaces* ou órgãos públicos, que buscam um preço de referência de diferentes produtos ou serviços, conquistando mais transparência e evitando fraudes ao automatizar o processo de obtenção de uma lista de produtos similares.

Referências Bibliográficas

Blei, D. M., Ng, A. Y., e Jordan, M. I. (2003). Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022.

Chang, W., Cheng, J., Allaire, J., Sievert, C., Schloerke, B., Xie, Y., Allen, J., McPherson, J., Dipert, A., e Borges, B. (2021). *shiny: Web Application Framework for R*. R package version 1.6.0.

De Haan, L., Ferreira, A., e Ferreira, A. (2006). *Extreme value theory: an introduction*, volume 21. Springer.

Devlin, J., Chang, M.-W., Lee, K., e Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Feinerer, I., Hornik, K., e Meyer, D. (2008). Text mining infrastructure in r. *Journal of Statistical Software*, 25(5):1–54.

Feuerriegel, S., Ratku, A., e Neumann, D. (2016). Analysis of how underlying topics in financial news affect stock prices using latent dirichlet allocation. In *2016 49th Hawaii International Conference on System Sciences (HICSS)*, pages 1072–1081. IEEE.

Gilleland, E. e Katz, R. W. (2016). extremes 2.0: an extreme value analysis package in r. *Journal of Statistical Software*, 72:1–39.

Hofmann, T. (1999). Probabilistic latent semantic indexing. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 50–57.

Hong, L. e Davison, B. D. (2010). Empirical study of topic modeling in twitter. In *Proceedings of the first workshop on social media analytics*, pages 80–88.

James, G., Witten, D., Hastie, T., e Tibshirani, R. (2013). *An introduction to statistical learning*, volume 112. Springer.

Jelodar, H., Wang, Y., Yuan, C., Feng, X., Jiang, X., Li, Y., e Zhao, L. (2019). Latent dirichlet allocation (lda) and topic modeling: models, applications, a survey. *Multimedia Tools and Applications*, 78(11):15169–15211.

Johnson, R. A., Wichern, D. W., et al. (2014). *Applied multivariate statistical analysis*, volume 6. Pearson London, UK:.

Kannan, S., Gurusamy, V., Vijayarani, S., Ilamathi, J., Nithya, M., Kannan, S., e Gurusamy, V. (2014). Preprocessing techniques for text mining. *International Journal of Computer Science & Communication Networks*, 5(1):7–16.

Kotz, S., Balakrishnan, N., e Johnson, N. L. (2004). *Continuous multivariate distributions, Volume 1: Models and applications*, volume 1. John Wiley & Sons.

Mikolov, T., Chen, K., Corrado, G., e Dean, J. (2013). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Qiang, J., Qian, Z., Li, Y., Yuan, Y., e Wu, X. (2020). Short text topic modeling techniques, applications, and performance: a survey. *IEEE Transactions on Knowledge and Data Engineering*.

R Core Team (2022). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

Röder, M., Both, A., e Hinneburg, A. (2015). Exploring the space of topic coherence measures. In *Proceedings of the eighth ACM international conference on Web search and data mining*, pages 399–408.

Russell, S. e Norvig, P. (2010). *Artificial Intelligence: A Modern Approach*. Prentice Hall, 3 edition.

Sievert, C. e Shirley, K. (2015). *LDAvis: Interactive Visualization of Topic Models*. R package version 0.3.2.

Tran, M. e Truong, M. (2019). Clustering short text messages using unsupervised machine learning. *LU-CS-EX 2019-20*.

van der Loo, M. (2014). The stringdist package for approximate string matching. *The R Journal*, 6:111–122.

Wickham, H. (2019). *stringr: Simple, Consistent Wrappers for Common String Operations*. R package version 1.4.0.

Wijffels, J. (2021a). *BTM: Biterm Topic Models for Short Text*. R package version 0.3.6.

Wijffels, J. (2021b). *udpipe: Tokenization, Parts of Speech Tagging, Lemmatization and Dependency Parsing with the 'UDPipe' 'NLP' Toolkit*. R package version 0.8.6.

Yan, X., Guo, J., Lan, Y., e Cheng, X. (2013). A biterm topic model for short texts. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1445–1456.