Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Visualização de vetores de texto por meio de projeções multidimensionais

Luís Felipe B. G. Silva

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Orientador
Prof. Dr. Vinícius Ruela Pereira Borges

Brasília
2021

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Visualização de vetores de texto por meio de projeções multidimensionais

Luís Felipe B. G. Silva

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Prof. Dr. Vinícius Ruela Pereira Borges (Orientador)
CIC/UnB

Dra. Aurea Rossy Soriano Vargas
IC/Unicamp

Prof. Dr. Geraldo Pereira Rocha Filho
CIC/UNB

Prof. Dr. Marcelo Grandi Mandelli
Coordenador do Bacharelado em Ciência da Computação

Brasília, 24 de abril de 2021

Resumo

A maioria dos trabalhos na literatura relacionados com a visualização de textos baseada no posicionamento de pontos consideram a representação de textos no modelo de espaço vetorial, obtida por técnicas como *bag-of-words* e *Term Frequency-Inverse Document Frequency* (TF-IDF). Apesar de ser popular, essa representação apresenta limitações ao capturar o contexto dos textos, pois não leva em consideração a ordem das palavras presentes no texto. Recentemente, as representações baseadas em *text embeddings* se mostraram promissoras ao gerar vetores dos textos com preservação do contexto. Dessa maneira, este estudo tem como objetivo investigar se os gráficos de espalhamento gerados por visualizações baseadas em projeções multidimensionais são capazes de refletir informações associadas ao contexto dos textos, como também expressar apropriadamente suas relações de similaridade.

Para esse propósito, foram realizados experimentos a partir de diferentes corpos de texto, sobre os quais foram aplicadas as técnicas de *text embeddings* Vetor de Parágrafos e *Bidirectional Encoder Representations from Transformers* (BERT). A partir dos vetores obtidos, as técnicas de redução de dimensionalidade *t-Stochastic Distributed Neighbor Embedding* (t-SNE) e *Uniform Manifold Approximation Projection* (UMAP) foram utilizadas para gerar os gráficos de espalhamento, que tiveram suas qualidades avaliadas com base em métricas que medem a preservação das relações previamente presentes no espaço de alta dimensionalidade.

A análise visual dos gráficos de espalhamento obtidos pela técnica t-SNE permite observar que textos similares em relação ao contexto foram posicionados próximos uns aos outros, formando grupos de pontos com baixa separabilidade entre si. Por sua vez, utilizando a técnica UMAP, foi possível verificar uma boa separação de grupos de pontos, associados a textos diferentes. Entretanto, dependendo do conjunto de vetores sobre o qual o UMAP é utilizado, são gerados gráficos de espalhamentos com grupos de pontos esparsos, o que dificulta a identificação de padrões e grupos de textos similares na análise visual.

Palavras-chave: doc2vec, tsne, UMAP, BERT, visualização, texto, projeção multidimensional, Vetor de Parágrafos

Abstract

In literature, several researches related to point placement visualization consider representations for texts based on the vector space model, such as the bag-of-words and Ferm Frequency-Inverse Document Frequency. Although being useful in text analysis tasks, this approach presents limitations regarding the context preservation on texts, since the words order is lost in these representations. Recent, approaches based on text embeddings have emerged as promising representations by generating embedding vectors that captures the context. This study proposes to investigate if projection-based visualizations are able to reflect context-based information from texts, as well as to express properly its similarity relations.

For this purpose, experiments were carried out using different text corpus, in which the text embeddings techniques Paragraph Vector and Bidirectional Encoder Representations from Transformers (BERT) were applied. After the vectors were obtained, the dimensionality reduction techniques t-Stochastic Distributed Neighbor Embedding (t-SNE) and Uniform Manifold Approximation Projection (UMAP) were employed as visualizations to generate the scatter plots, and its quality was assessed based on metrics that measure the preservation of the relationships previously present in the high dimensional space.

The visual analysis of the scatter plots obtained by the t-SNE technique, shows that similar context texts were positioned close to each other, forming groups of points with low separability from each other. On the other hand, using the UMAP technique it was possible to verify a good separation of groups of points, associated with different texts. However, depending on the set of vectors on which UMAP is used, scatter plots are generated with widespread groups of points, which makes it difficult to identify patterns and groups of similar texts in visual analysis.

Keywords: doc2vec, tsne, UMAP, BERT, visualization, text, multidimensional projection, Paragraph Vector

Sumário

1 Introdução **1**
1.1 Objetivos . . . 3
1.2 Estrutura da monografia . . . 4

2 Fundamentos **5**
2.1 Textos . . . 5
2.2 Mineração de textos . . . 5
2.2.1 Pré-processamento de textos . . . 6
2.2.2 Caracterização de textos . . . 7
2.2.3 Text embeddings . . . 8
2.3 Visualização da Informação . . . 11
2.3.1 Projeções multidimensionais . . . 11
2.3.2 Medidas de avaliação da qualidade de projeções . . . 14
2.4 Considerações finais . . . 15

3 Revisão de literatura **16**

4 Metodologia proposta **18**
4.1 Corpos de textos . . . 18
4.2 Pré-processamento de textos . . . 19
4.3 Text Embeddings . . . 20
4.4 Projeção multidimensional . . . 20
4.5 Cálculo das métricas de avaliação . . . 21
4.6 Gráficos de espalhamento . . . 21
4.7 Análise dos Resultados . . . 22

5 Resultados experimentais **23**
5.1 Hiper-parâmetros . . . 23
5.1.1 Vetor de Parágrafos . . . 23
5.1.2 RoBERTa . . . 23

5.1.3 t-SNE 24
5.1.4 UMAP 24
5.2 Métricas MRPD e NBP 24
5.2.1 Pequenas vizinhanças 25
5.2.2 Grandes vizinhanças 27
5.3 Gráficos de espalhamento 28
5.3.1 News 28
5.3.2 Tweets variados 40
5.3.3 Tweets rotulados por polaridade 47
5.4 Análise dos resultados 54

6 Conclusão **57**
6.1 Trabalhos futuros 58

Referências **59**

Lista de Figuras

2.1 Representação simplificada dos modelos de *Word2Vec* apresentada em Mikolov et al. [1]. 9

4.1 Etapas do estudo. 18

5.1 Comparação dos gráficos de MRPD, obtidos para valores de k no intervalo $[1, 100]$ 25
5.2 Comparação dos gráficos de NBP, obtidos para valores de k no intervalo $[1, 100]$ 26
5.3 Comparação dos gráficos de MRPD, obtidos para valores de k no intervalo $[101, 600]$ 27
5.4 Comparação dos gráficos de NBP, obtidos para valores de k no intervalo $[101, 600]$ 28
5.5 Legenda adotada para associar as cores às categorias do corpus *News*. . . . 29
5.6 Layout obtido pela projeção FIt-SNE, para o corpus *News* a partir de Vetor de Parágrafos 300D. 29
5.7 Layout obtido pela projeção BH t-SNE, para o corpus *News* a partir de Vetor de Parágrafos 300D. 30
5.8 Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de Vetor de Parágrafos 300D. 31
5.9 Layout obtido pela projeção UMAP com *min_dist=0.1*, para o corpus *News* a partir de Vetor de Parágrafos 300D. 32
5.10 Layout obtido pela projeção BH t-SNE, para o corpus *News* a partir de vetores RoBERTa com modelo *roberta base*. 33
5.11 Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de vetores RoBERTa com modelo *roberta base*. 34
5.12 Região central do layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de vetores RoBERTa com modelo *roberta base*. 35
5.13 Região superior direta do layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de vetores RoBERTa com modelo *roberta base*. 35

5.14 Layout com marcações, obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de vetores RoBERTa com modelo *roberta base*. . . . 37
5.15 Layout com marcações, obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de Vetor de Parágrafos de 300 dimensões. 38
5.16 Layout obtido pela projeção BH t-SNE, para o corpus *News* a partir de vetores RoBERTa com o modelo *roberta large*. 39
5.17 Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de vetores RoBERTa com o modelo *roberta large*. 40
5.18 Layout obtido pela projeção FIt-SNE, para o corpus *Tweets* a partir de Vetor de Parágrafos 300D. 42
5.19 Layout obtido pela projeção FIt-SNE, para o corpus *Tweets* a partir de vetores RoBERTa. 43
5.20 Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *Tweets* a partir de Vetor de Parágrafos 300D. 44
5.21 Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *Tweets* a partir de vetores RoBERTa com o modelo *roberta base*. 45
5.22 Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *Tweets* a partir de vetores RoBERTa com o modelo *roberta large*. 45
5.23 Layouts obtidos pela projeção UMAP com *min_dist=0*, para o corpus *Tweets* a partir de vetores RoBERTa com o modelo *roberta base*. 46
5.24 Layouts obtidos pela projeção UMAP com *min_dist=0*, para o corpus *Tweets* a partir de vetores RoBERTa com o modelo *roberta large*. 47
5.25 Layout obtido pela projeção FIt-SNE, para o corpus *Labeled Tweets* a partir de Vetor de Parágrafos 300D. 48
5.26 Layout obtido pela projeção FIt-SNE, para o corpus *Labeled Tweets* a partir de vetores RoBERTa. 49
5.27 Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *Labeled Tweets* a partir de Vetor de Parágrafos 300D. 50
5.28 Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *Labeled Tweets* a partir de vetor RoBERTa com o modelo *roberta base*. 51
5.29 Layouts obtidos pela projeção UMAP com *min_dist=0*, para o corpus *Labeled Tweets* a partir de vetor RoBERTa com modelo o *roberta base*. 52
5.30 Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *Labeled Tweets* a partir de vetor RoBERTa com o modelo *roberta large*. 53
5.31 Layouts obtidos pela projeção UMAP com *min_dist=0*, para o corpus *Labeled Tweets* a partir de vetor RoBERTa com o modelo *roberta large*. . . . 54

Lista de Tabelas

5.1 $x.g$ indica que a categoria g é representada pelo número x. 28
5.2 Exemplos de textos do corpus *News*. 36
5.3 Textos representados pelos pontos marcados nos layouts do corpus *Tweets*. . 41
5.4 Características das distâncias entre vetores de alta dimensionalidade. 55

Capítulo 1

Introdução

Atualmente, um grande volume de texto é gerado diariamente e disponibilizado publicamente na internet. Entretanto, a análise de todos esses textos é extremamente difícil de ser realizada por especialistas humanos. Para esse fim, foram desenvolvidas diversas técnicas de mineração de texto, que visam extrair dados de maneira automática ou semi-automática e, a partir disso, obter informação e realizar a análise de textos, definindo uma estrutura da qual é possível extrair relações entre diferentes corpos de texto.

Grande parte das técnicas modernas de mineração de textos utilizam aprendizado de máquina para extrair estruturas e características de textos. Por meio destas técnicas, é possível realizar recomendação de texto similares [2], por exemplo, utilizando processamento de linguagem natural, pode-se criar modelos capazes de serem treinados para realizar tarefas específicas, como análise de sentimentos, além da geração de respostas contextualizadas ou identificar sarcasmo ou ironia [3].

Textos são dados naturalmente não estruturados. Assim, uma etapa comum na mineração de textos é representá-los de uma forma mais simples de ser processada pelo computador. Uma representação estruturada eficiente são vetores numéricos, uma estrutura simples de ser processada e que representa relações entre dados a partir de relações no espaço. Uma das representações de texto mais simples é o *bag-of-words* [4], que cria um vetor que armazena quantas vezes cada palavra ocorre no texto. O *Term Frequency-Inverse Document Frequency* (TF-IDF) [5] é uma técnica que normaliza os dados para valorizar palavras com base na quantidade de vezes que cada palavra ocorre em um texto com relação ao número de ocorrências no conjunto de textos analisados, e pode ser combinado com o *bag-of-words* para obter vetores que melhor representam o texto.

Apesar de viabilizar o uso das técnicas de análise de textos, o *bag-of-words*, ou suas variações mais simples, como o TF-IDF, não preservam toda a informação do texto representado, pois a ordem das palavras no texto, e consequentemente o contexto, não é preservada nesta abordagem. Técnicas de representação de textos baseadas em *text em-*

beddings, como o Vetor de Parágrafos [6], buscam representar cada palavra com base no contexto ao analisar a vizinhança de palavras. Como resultado, os modelos geram vetores que representam os textos de forma mais acurada que o *bag-of-words*. Recentemente, foi proposto o modelo *Bidirectional Encoder Representations from Transformers* (BERT) [7], que incorpora uma arquitetura de rede neural conhecida como *Transformer* [8], visando gerar redes neurais pré-treinadas que podem ser utilizadas para obter vetores que representam textos em um espaço multidimensional.

Apesar de mais simples de serem processados por computadores, é difícil para qualquer humano, mesmo especialista na área, extrair informação analisando vetores de alta dimensionalidade. Simultaneamente, ler, comparar e identificar uma estrutura em milhares ou milhões de textos diferentes, a partir de sua forma original, também exigiria um esforço sobre-humano. Para tal fim, as técnicas de visualização são alternativas viáveis para tarefas de análise de textos [9]. O objetivo da visualização é a geração de representações gráficas e intuitivas a partir de dados abstratos, visando facilitar o entendimento dos padrões relevantes e implícitos [10] contidos no conjunto de dados observado. Nesse cenário, um especialista é capaz de comparar uma grande quantidade de dados com facilidade, podendo avaliar tanto os dados observados como a técnica utilizada para gerar as representações visuais que refletem a informação mostrada, como exposto em Ali et al. [11].

As técnicas clássicas de visualização, como o gráfico de barra, linhas ou pizza, embora largamente utilizadas para representar uma grande quantidade de informação de forma visual, apresentam limitações para visualizar dados multidimensionais. Especificamente, para representar textos, buscando preservar as diversas relações e grupos que podem existir entre diferentes textos, as visualizações baseadas no posicionamento de pontos no espaço visual [12] são as mais adequadas. Nesse sentido, as projeções multi-dimensionais [13] surgem como alternativas apropriadas para tarefas de visualização de textos [14].

As projeções multidimensionais baseadas em redução de dimensionalidade permitem obter representações bidimensionais de vetores de alta dimensionalidade, preservando boa parte da informação e relações dos vetores no espaço original [15]. Uma das técnicas de redução de dimensionalidade mais populares é a *t-Stochastic Distributed Neighbor Embedding* (t-SNE) [16], que apresenta um método estatístico que modela a chance de pares de pontos serem vizinhos em uma distribuição t-Student. Recentemente, foi proposta a técnica *Uniform Manifold Approximation Projection* (UMAP) [17], que se baseia na geometria Riemanniana para definir um modelo que visa ser eficiente para projeções de dados não lineares, que preserva a estrutura global dos dados e com baixa complexidade de tempo.

A visualização de textos baseadas em projeções multi-dimensionais demandam que

cada texto esteja representado sob a forma de um vetor de características. Tais vetores podem ser obtidos por meio da extração de métricas dos textos, como ocorre nas técnicas baseadas no modelo espaço-vetorial, como o Bag-of-Words e o TF-IDF. Entretanto, a maioria dos trabalhos na literatura empregam essas representações para textos em tarefas de visualização analítica ou visualização exploratória [18]. Assim, nesse projeto, a ideia consiste em explorar as representações baseadas em *text embeddings*, uma vez que é possível obter vetores que preservam relações semânticas entre textos como a distância entre vetores.

Tendo em vista a consolidação do t-SNE e a base teórica e resultados apresentados no UMAP, essas foram as técnicas escolhidas para serem testadas no contexto de redução de dimensionalidade de vetores de texto. A partir da aplicação de técnicas eficientes dos *text embeddings* Vetor de Parágrafos e BERT, além do emprego das projeções multidimensionais baseadas em redução de dimensionalidade sobre os vetores de textos, foram realizadas análises visual e de qualidade dos *layouts* (representações gráficas) gerados por meio de métricas objetivas, testando-se também variações das técnicas e diferentes hiperparâmetros. Os experimentos foram realizados sobre dois conjuntos (*corpus*) contendo pequenos textos. Um deles se trata de um conjunto de *tweets* rotulados conforme a polaridade (positivo, negativo e neutro) e outro conjunto de textos de mensagens de um grupo de notícias, conhecido como *20 newsgroups* [19].

1.1 Objetivos

Com este estudo, espera-se observar se as técnicas de projeção multidimensional podem ser empregadas para visualização de textos que estejam representados por *text embeddings*. Assim, pretende-se investigar se os *layouts* gerados pelas visualizações permitem realizar análise visual de conjuntos de textos, identificando documentos similares e padrões implícitos relevantes. Especificamente, essa pesquisa tem como objetivos:

1. Verificar se os aspectos inerentes às representações vetoriais consideradas, como, por exemplo, o contexto dos textos, são refletidas apropriadamente nos *layouts* gerados pelas projeções multidimensionais.
2. Avaliar os *layouts* obtidos por meio de métricas de preservação de vizinhança e análise qualitativa dos gráficos de espalhamento.

Atingindo-se os objetivos propostos, serão destacados pontos de fortes de fracos dos algoritmos t-SNE e UMAP na redução de dimensionalidade de diferentes vetores de textos. A partir disso, almeja-se entender melhor em quais cenários cada técnica de projeção

multidimensional é aplicável, de acordo com a representação de textos adotada e objetivo da análise visual que se pretende realizar.

1.2 Estrutura da monografia

No Capítulo 2 serão apresentados os fundamentos básicos para o entendimento das técnicas analisadas nesta monografia, no Capítulo 3 uma revisão de trabalhos relacionados, no Capítulo 4 a apresentação do método adotado no estudo, no Capítulo 5 apresentada uma análise dos resultados obtidos e por fim, no Capítulo 6, as considerações finais sobre o estudo realizado.

Capítulo 2

Fundamentos

Neste capítulo serão apresentados conceitos básicos para a realização do estudo: fundamentos de mineração de textos, representação de textos, visualização de dados e redução de dimensionalidade.

2.1 Textos

Um texto é um documento constituído de palavras, sequências de caracteres, onde cada palavra possui uma função semântica que se relaciona às demais do contexto e contribui para o significado do todo. Características fora do corpo do texto, como o contexto em que o texto é apresentado, autor e meio de publicação, influenciam a forma com que o interpretamos. No entanto, a maior parte do contexto pode ser obtido a partir da extração das palavras, mantendo a ordem e desconsiderando a origem ou meio de publicação do texto, e é sobre esta forma mais básica de textos que este estudo trabalhará.

2.2 Mineração de textos

A mineração de textos é uma sub-área da mineração de dados focada em textos. Enquanto a mineração de dados pode ser utilizada para analisar grandes conjuntos de dados estruturado, a mineração de textos é utilizada para extrair informação de dados não estruturados, cujo significado não é naturalmente reconhecido pelo computador [20]. Para isso são utilizadas diversas técnicas que simplificam o texto, isolam os dados e constroem uma estrutura organizada e facilmente processável por um computador. Algumas destas técnicas são apresentadas nesta seção.

2.2.1 Pré-processamento de textos

Remoção de stop-words

Algumas palavras no texto podem ser consideradas irrelevantes para extrair informação, e estas são consideradas *stop-words*. As *stop-words* podem variar dependendo do contexto, linguagem e do objetivo da mineração de texto realizada.

Palavras consideradas *stop-words* podem ser identificadas e removidas do texto sem perda de informação, reduzindo o tamanho dos corpos a serem analisados. Exemplos comuns de *stop-words* são artigos, por exemplo "o" e "uma", conectivos, como "e" e "então" ou pronomes como "quem" e "qual".

Original	quem robou o bolo?	Eu amo os eles
Sem stop-words	robou bolo?	amo

Stemmização e Lematização

A stemmização e lematização têm como objetivo reduzir o vocabulário do texto, simplificando as palavras para uma forma mais básica que representa o sentido da palavra. Em ambas as técnicas um conjunto de palavras é mapeado para uma única palavra.

Na stemmização, o processo é realizado a partir da remoção de caracteres da palavra original, geralmente do sufixo, gerando uma palavra final reduzida que não necessariamente pertence à linguagem. Na stemmização o contexto não é considerado. Alguns resultados que podem ser obtidos são:

Original	bobagem	boba	boi	boiando	melhor
Stemmização	bobag	bob	boi	boi	melh

Na lematização o contexto é considerado e a palavra final obtida pertence à linguagem do texto. Este processo é mais lento que o de stemmização e é necessária uma base de dados com as palavras que são geradas. Alguns dos resultados que podem ser obtidos com lematização são:

Original	bobagem	boba	boi	boiando	melhor
Lematização	bobo	bobo	boi	boia	bom

Remoção de caracteres

Quando realizamos o processamento de um texto podemos nos deparar não somente com palavras mas com outros elementos, como caracteres de pontuação ou caracteres especiais diversos. Dependendo do que desejamos analisar, podemos remover alguns destes caracteres que consideramos não relevantes para nosso contexto. Alguns dos caracteres comumente removidos são:

? ! "/ # .) (| , ’

2.2.2 Caracterização de textos

Nas próximas seções são apresentados métodos de representação de texto utilizando vetores numéricos.

Bag-of-words

Bag-of-words [4] é uma técnica simples de representação de texto, em sua forma mais simples, cada palavra do vocabulário A é mapeada para um número inteiro e um vetor V de tamanho $|A|$ é criado para representar o texto. Cada posição V_i desse vetor guarda um número inteiro que representa a quantidade de vezes que a palavra mapeada para o inteiro i ocorre no texto representado.

Como já mencionado anteriormente, o *bag-of-words* é uma técnica simples, que caracteriza o texto com base na frequência de suas palavras, uma representação facilmente processável por um computador. A partir do *bag-of-words*, diferentes técnicas podem ser combinadas para superar os problemas nele presentes, como a perda do contexto e o tamanho dos vetores que crescem proporcionalmente ao vocabulário.

TF-IDF

O *Term Frequency-Inverse Document Frequency* (TF-IDF) [5] é uma técnica que tem como objetivo determinar quais palavras do texto são mais relevantes, com base no conjunto de textos analisados. O TF-IDF atribui um peso para cada palavra do vocabulário, baseado na frequência que a palavra ocorre em um documento específico em relação ao número de ocorrências no conjunto de documentos analisados. O cálculo do TF-IDF é obtido a partir da combinação da frequência de termos (TF) e com o inverso da frequência em documentos (IDF) o cálculo de ambas as partes é apresentado a seguir.

Seja p a palavra analisada no documento d e $f_{p,d}$ a quantidade de ocorrências desta palavra em d, o valor TF de p em d é definido como a divisão número de ocorrências de p em d pelo total de palavras em d, como mostrado na equação 2.1.

$$TF(p,d) = \frac{f_{p,d}}{\sum_{p' \in d} f_{p',d}} \tag{2.1}$$

O valor IDF para cada palavra p do conjunto de documentos D de tamanho N é calculado como $IDF(p,D) = log(\frac{N}{oc_{p,D}})$ onde $oc_{p,D}$ indica o número de documentos em D que contém a palavra p. O valor TF-IDF para cada palavra p em um documento $d \in D$ é dado então como TF-IDF$(p,d) = TF(p,d) \cdot IDF(p,D)$.

N-gram

Uma forma de obter mais informação sobre o contexto quando se utiliza *bag-of-words* é utilizar o *N-gram* [21]. O *N-gram* utiliza uma sequência de N termos consecutivos no texto para mapear cada índice do vetor que representa o texto, em oposição a uma palavra, como na versão básica do *bag-of-words*.

Em um *2-gram* cada par de palavras que aparecem em sequência no texto serão consideradas como uma unidade única que é mapeada para uma posição no vetor frequências. Está abordagem fornece mais informação sobre o contexto e é possível ampliar o intervalo N para obtermos mais informação, o problema que esta abordagem gera é a quantidade possível de pares, triplas ou n-uplas de palavras que podem existir em um texto.

Em um *2-gram* é possível existir $|A|^2$ pares de palavras diferentes, onde $|A|$ é o tamanho do vocabulário, e conforme aumentamos o valor de N o tamanho do nosso vetor pode aumentar ainda mais. Para um $N-gram$ a quantidade de n-uplas mapeadas para posições no vetor de frequências é da ordem de $O(|A|^N)$.

2.2.3 Text embeddings

Com base nas técnicas já apresentadas de obter informação sobre o contexto, e utilizando modelos de aprendizado de máquina, outras formas de gerar vetores, mais complexas e eficientes, que representam palavras e textos foram desenvolvidas. As técnicas a seguir apresentadas são conhecidas como técnicas de *word embeddings* ou *text embeddings* [22].

Word2vec

Word2vec [1] é uma técnica que tem como objetivo obter, através de um modelo de rede neural, vetores numéricos que representam palavras baseando-se em um contexto. Existem duas principais abordagem para construir estes vetores, são elas *Continuos Bag of words* (CBOW) e *Skip-gram*.

Ambas abordagens utilizam informação de palavras contíguas no treinamento do modelo. A principal diferença entre as duas abordagens é:

1. CBOW: utiliza informações sobre as palavras vizinhas para prever a palavra atual.
2. Skip-gram: utiliza a palavra atual para prever quais são as palavras vizinhas.

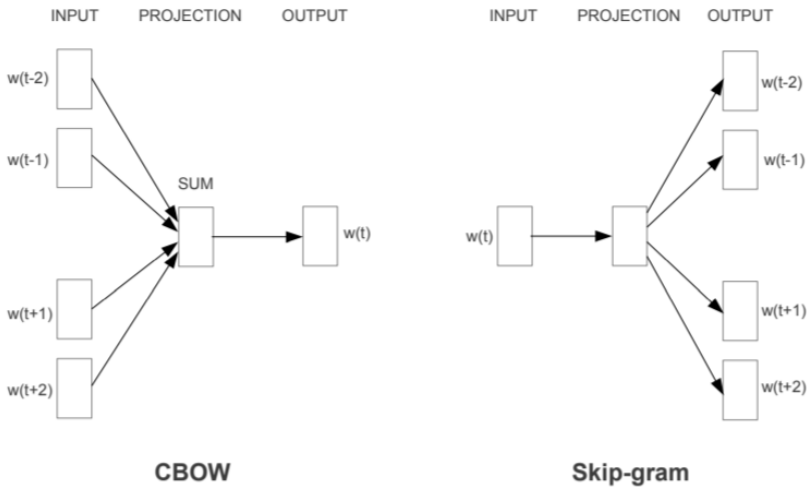


Figura 2.1: Representação simplificada dos modelos de *Word2Vec* apresentada em Mikolov et al. [1].

Na Figura 2.1, retirada de Mikolov et al. [1], é apresentado um modelo simplificado das técnicas de *Word2vec*, onde a camada de entrada recebe uma ou mais palavras, que são utilizadas pelas camadas ocultadas da rede, nomeadas como SUM, que possui matrizes de peso que são ajustados e acumulam informação semântica sobre as palavras durante o treinamento, e produz um resultado, em $OUTPUT$, que apresenta a probabilidade da palavra ou conjunto de palavras previstas.

Uma das características mais importantes desta técnica para este estudo é relação entre a distância dos vetores gerados e a semântica da palavra representada, vetores mais próximos representam palavras semanticamente mais próximas. As demais técnicas de *text embeddings* apresentadas também possuem esta característica.

Word2Vec é uma método eficiente para representar palavras, e a partir dele é possível criar diferentes representações de texto. Somando os vetores das palavras contidas em um texto e aplicando-se o TF-IDF, por exemplo, obtém-se uma representação vetorial do texto.

Vetor de Parágrafos

Baseando-se nas técnicas do *Word2Vec*, o Vetor de Parágrafos [6] constrói vetores para representar parágrafos através de duas técnicas distintas: Vetor de Parágrafos com memória distribuída (PV-DM) e Vetor de Parágrafos com *bag-of-words* distribuída (PV-DBOW), ambas utilizam redes neurais com gradiente descendente e *backpropagation* para construir os vetores dos parágrafos, porém a partir de abordagens diferentes.

- PV-DM: assemelha-se ao modelo CBOW, onde um valor de entrada, PV, representando o parágrafo, é adicionado à camada de entrada da rede utilizada para o treinamento dos vetores de palavras. A entrada extra PV é utilizada juntamente com as palavras do contexto para prever a próxima palavra e acumula informação semântica sobre o parágrafo durante o treinamento.
- PV-DBOW: assemelha-se ao modelo Skip-gram, assim como no modelo Skip-gram uma entrada é utilizada para prever as palavras contidas no contexto, a diferença é que no PV-DBOW a entrada representa o parágrafo e não uma palavra, e esta entrada é utilizada para prever diferentes janelas de palavras ao longo do parágrafo analisado, enquanto acumula informação semântica.

Ambas as técnicas para construir vetores de parágrafos são eficientes, e podem ser utilizadas isoladamente ou combinadas para uma representação mais acurada do parágrafo, como sugerido em Le et al. [6].

BERT

O *Bidirectional Encoder Representations from Transformers* (BERT) [7] é uma técnica que utiliza a arquitetura de redes neurais *Transformer* [8] para construir um modelo pré-treinado de representação de linguagem. O principal diferencial do BERT em relação a outros modelos é a captura de contexto de forma bidirecional, através do uso da arquitetura *Transformer*.

O *WordPiece* [23] é utilizado pelo BERT para representar as palavras do texto em símbolos menores que contém seu significado, chamados tokens. A palavra "*walking*" é dividida em dois tokens, "*walk*" e "*##ing*", por exemplo. O *WordPiece* já realiza o pré-processamento necessário, portanto não é necessário aplicar técnicas como remoção de *stop-words* e stemmização.

O treinamento do modelo é realizado sobre a tarefa de *Mask Language Modeling (Masked LM)*. A tarefa *Masked LM* [7] visa evitar que as palavras utilizadas enxerguem a si mesmas durante o treinamento. Para isto é realizada a substituição de 15% dos tokens do texto por um token especial, denominado $[MASK]$, a tarefa então se torna prever os tokens originais que ocupavam as posições de $[MASK]$. Em Devlin et al. [7] é apresentada a tarefa *Next Sequence Prediction (NSP)*, que é utilizada juntamente com a *Masked LM* para treinar o modelo, porém em Liu et al. [24] foi demonstrado que a tarefa de NSP não é necessária para se obter um boa representação do texto.

O BERT possui um passo pós treinamento de refinamento, realizada para adaptar os vetores pré-treinados para tarefas específicas. Enquanto os vetores pré treinados são

obtidos através de um treinamento que pode demorar dias, o processo de refinamento é realizado rapidamente para permitir que os vetores se adaptarem a qualquer tarefa.

2.3 Visualização da Informação

A forma básica de se analisar dados, é por meio de técnicas de visualização e interpretação de características observadas. Existem diferentes formas de apresentar a informação a ser visualizada, com gráficos de barras, espalhamento ou grafos, por exemplo, cada forma de apresentação apresenta diferentes características do conjunto de dados e é adequada a diferentes contextos e tipos de dados.

Alguns tipos de representação capturam a estrutura e guardam informação de forma bem estruturada sobre os dados mostrados, como é o caso de grafos, onde cada vértice pode representar uma instância de dados e as arestas as relações entre estes vértices. Porém, em conjuntos nos quais se deseja observar a relação entre vários dados, a quantidade de arestas pode dificultar a análise visual. Além disso, quando limitamos as relações, utilizando árvores por exemplo, pode-se perder informação de relações entre os dados.

Uma abordagem popular são os gráficos de dispersão, em que cada instância dado é representada como um ponto no gráfico, que pode ter duas ou três dimensões. A análise desses gráficos consiste em observar a distância entre pontos, regiões densas e pontos isolados para inferir os padrões implícitos do conjunto de dados. Nesta abordagem, o conjunto representado pode conter menos informação do que em um grafo denso, porém é de mais fácil visualização para grandes conjuntos e não são impostas grandes restrições à estrutura, como em árvores.

Os gráficos de dispersão não podem ser obtidos diretamente de um conjunto de textos representado em sua forma original. Ao gerar uma representação vetorial dos textos, se os vetores obtidos tiverem dimensão menor ou igual a 3, estes podem ser utilizados para representar os textos em um gráfico de dispersão. Porém, as técnicas de *text embeddings* produzem vetores com dimensionalidade superior a três, portanto estes vetores não podem ser visualizados em uma representação espacial. Para obter um gráfico de espalhamento visível a partir desses vetores, são utilizadas as técnicas de projeção multidimensional.

2.3.1 Projeções multidimensionais

Como descrito em Tejada et al. [25], uma técnica de projeção multidimensional mapeia um conjunto de dados representados em um espaço m-dimensional para um espaço p-dimensional, com p menor que m, preservando o máximo possível das relações contidas no espaço original. Para utilizar as projeções multidimensionais como técnicas de visuali-

zação, utiliza-se p menor ou igual a 3, em que o resultado obtido é um conjunto de pontos no espaço.

As técnicas de representação de textos mais eficiente atualmente utilizam vetores multidimensionais, e, utilizando técnicas de projeção multidimensional sobre estes vetores, podemos representar visualmente conjuntos de textos. Serão apresentadas a seguir, duas das técnicas atualmente mais relevantes de projeção multidimensional.

t-SNE

O t-SNE [16] é uma técnica de projeção popular baseada na técnica SNE, porém aplica algumas modificações para resolver os problemas de sobreposição de pontos no espaço visual e otimização do SNE. O t-SNE mapeia a distância euclidiana no espaço de alta dimensionalidade $\mathbb{R}^m$ para probabilidades condicionais que representam a similaridade entre os pontos, onde $p_{i,j}$ é a probabilidade de que um ponto $x_i \in \mathbb{R}^m$ escolha $x_j \in \mathbb{R}^m$ como seu vizinho, sendo que tal probabilidade é proporcional à densidade de probabilidade de uma distribuição Gaussiana com centro em x_i. O cálculo de $p_{i,j}$ é apresentado em 2.2.

$$p_{i,j} = \frac{exp(||x_i - x_j||^2 \div 2\sigma^2)}{\sum_{k \neq i} exp(||x_i - x_k||^2 \div 2\sigma^2)} \tag{2.2}$$

Onde σ é a variância. Mais detalhes sobre como o valor da variância é escolhida em Maaten e Hinton [16].

Os pontos do espaço de dimensão reduzida $\mathbb{R}^n$, também tem suas distâncias mapeadas para uma função de probabilidade, $q_{i,j}$, porém diferentemente dos pontos de alta dimensionalidade e da abordagem usada no SNE, o t-SNE utiliza uma distribuição de probabilidade t-Student com 1 grau de liberdade para este mapeamento. Ao utilizar uma distribuição de cauda longa, o t-SNE ameniza o problema de sobreposição de pontos no espaço visual que pode ocorrer ao utilizar a distribuição SNE. Além de ser uma distribuição próxima à Gaussiana, as probabilidades da t-Student são computacionalmente simples de serem calculadas, já que não envolvem uma exponencial. A fórmula para obter a probabilidade $q_{i,j}$ entre os pontos projetados no espaço de dimensão reduzida $y_i \in \mathbb{R}^n$ e $y_j \in \mathbb{R}^n$ é apresentada na Equação 2.3.

$$q_{i,j} = \frac{(1 + ||y_i - y_j||^2)^{-1}}{\sum_k \sum_{l \neq k} (1 + ||y_k - y_l||^2)^{-1}} \tag{2.3}$$

O objetivo do t-SNE então é minimizar a divergência de Kullback-Leibler entre as probabilidades P, entre os pontos de alta dimensionalidade, e Q, dos pontos de baixa dimensionalidade. O processo de minimização da função de custo $C = KL(P|Q) = p_{i,j} \cdot log(\frac{p_{i,j}}{q_{i,j}})$ é realizado por meio de um esquema gradiente descendente, onde as diferentes probabili-

dades atuam como forças que aproximam ou repelem os pontos. A função gradiente e os passos para obtê-la são demonstrados em Maaten e Hinton [16].

Quando comparada outras técnicas de projeção multidimensional, o t-SNE apresentar boa preservação de características do conjunto de dados projetado [16]. Porém a implementação do t-SNE é computacionalmente ineficiente em tempo para grandes conjuntos de dados, já que é necessário calcular as distâncias entre cada par de pontos a cada iteração. Assim, pode-se utilizar diferentes técnicas para calcular estas distâncias com menor custo computacional. Neste estudo utilizaremos a implementação referência de Linderman et al. [26], que apresenta implementações eficientes de duas variantes do t-SNE:

- Barnes-Hut t-SNE [27], que utiliza árvores de ponto de vantagem para calcular a distância somente entre pontos que mais influenciam no gradiente descendente a cada iteração, com pouca perda de qualidade na projeção, e o algoritmo de Barnes-Hut para simular a força aplicada entre os pares de pontos a cada iteração (Simulações de N-corpos).
- Flt-SNE [26], utiliza a biblioteca Annoy [28] para obter um aproximação dos pares de pontos mais próximos e realiza convoluções utilizando Transformada Rápida de Fourier (FFT) para realizar a Simulações de N-corpos.

UMAP

O Uniform Manifold Approximation and Projection (UMAP) [17] é um algoritmo com bases teóricas na geometria Riemanniana. No artigo em que o UMAP é proposto, as decisões realizadas no algoritmo são apresentadas com base na topologia algébrica [29] e na teoria de categorias [30]. Aqui, o UMAP será apresentado com base na abordagem focada em conceitos computacionais do algoritmo, apresentada em Leland et al. [17] e Oslkolkov [31].

Inicialmente, uma métrica de dissimilaridade d e um tamanho de vizinhança k são escolhidos, e assim, para cada ponto no espaço de alta dimensionalidade $x_i \in \mathbb{R}^m$, um conjunto $v_{k,i}$ dos k vizinhos mais próximos de x_i é construído, sendo $v_{k,i,j}$ o j-ésimo vizinho mais próximo de x_i. Na implementação referência do UMAP, que será utilizada neste estudo, é utilizado o algoritmo *nearest neighbor descent* [32] para obter as listas de vizinhos.

Para obter os pesos das arestas são calculados os valores ρ e σ para cada ponto x_i do espaço de alta dimensionalidade, em que a função d define a distância entre pontos. ρ_i é dado por:

$$\rho_i = min\{d(x_i, v_{k,i,j}) | 1 \leq j \leq k, d(x_i, v_{k,i,j}) > 0\} \tag{2.4}$$

σ_i é o valor que satisfaz:

$$\sum_{j=1}^{k} exp\left(\frac{-max(0, d(x_i, v_{k,i,j}) - \rho_i)}{\sigma_i}\right) = log_2(k) \tag{2.5}$$

Uma matriz de adjacência A é então construída, onde $A_{i,j}$, o peso da aresta que conecta o vértice i ao j, é dado por:

$$A_{i,j} = exp\left(\frac{-max(0, d(x_i, v_{k,i,j}) - \rho_i)}{\sigma_i}\right) \tag{2.6}$$

O valor $A_{i,j}$ pode ser interpretado como a probabilidade da aresta do vértice i para o vértice j existir. Para representar os pontos, é utilizada uma representação simétrica dada pela matriz de adjacência B, onde $p_{i,j} = B_{i,j}$ representa a chance de uma das arestas entre i e j existirem. B é definido como:

$$B = A + A^T - A \circ A^T \tag{2.7}$$

Em que $\circ$ representa a operação de multiplicação ponto a ponto.

Dados os hiper-parâmetros a e b, as distâncias entre os pontos y, pertencentes ao espaço de baixa dimensionalidade $\mathbb{R}^n$, são dadas pela função:

$$q_{i,j} = (1 + a(y_i - y_j)^{2b})^{-1} \tag{2.8}$$

A partir disso, é utilizado gradiente descendente para obter a representação dos pontos de baixa dimensionalidade a partir da otimização da função de custo:

$$C = \sum_i \sum_j p_{i,j} log\left(\frac{p_{i,j}}{q_{i,j}}\right) + (1 - p_{i,j}) log\left(\frac{(1 - p_{i,j})}{(1 - q_{i,j})}\right) \tag{2.9}$$

2.3.2 Medidas de avaliação da qualidade de projeções

A seguir são apresentadas as métricas utilizadas para avaliar a qualidade das projeções analisadas.

Preservação de vizinhança (NBP)

Essa métrica tem como objetivo verificar quantos dos k vizinhos mais próximos de cada ponto no espaço de alta dimensionalidade são preservados no espaço de baixa dimensão. Dado que hk_i é o conjunto dos k pontos mais próximos do ponto x_i no espaço de alta dimensionalidade e lk_i o conjunto dos k pontos mais próximos de y_i que mapeia x_i no espaço projetado, a medida de NBP [14] da projeção de n pontos é dada por:

$$NBP_k = \frac{1}{n} \cdot \sum_{i=1}^{n} \frac{hk_i \cap lk_i}{k}$$

O resultado dado pela Preservação de vizinhança é um valor racional entre 0 e 1, sendo que valores maiores representam uma melhor preservação da vizinhança. O valor de k pode variar de acordo com o objetivo da análise, ondem valores pequenos de k indicam a preservação da estrutura local e valores maiores de k a estrutura global.

Diferença de posição relativa (MRPD)

A medida Diferença de posição relativa (MRPD) também avalia a qualidade da preservação da vizinhança, porém, ao contrário da NBP, não é utilizada uma medida binária que considera somente os vizinhos presentes em ambas as representações, de alta e baixa dimensionalidade, como vizinhos preservados. Ao invés disso, para cada um dos k vizinhos mais próximos de um ponto y_i projetado, a preservação é medida verificando-se a diferença entre a posição original, na lista de pontos ordenados pela distância em relação ao ponto x_i do espaço original, e a nova posição na lista de pontos ordenados pela distância em relação ao ponto y_i.

Dada uma projeção de pontos x no espaço de alta dimensionalidade $\mathbb{R}^m$, para pontos y no espaço dimensionalidade reduzida $\mathbb{R}^n$. Temos $ln_{i,j}$ como o j-ésimo ponto mais próximo do ponto projetado y_i no espaço de dimensão reduzida, e $f(i, ln_{i,j})$ indica a posição de $ln_{i,j}$ na lista de vizinhos ordenados do ponto x_i, no espaço original. O valor de $MRPD_k$, considerando uma vizinhança de tamanho k em um conjunto de q pontos é dada pela Equação 2.10.

$$MRPD_k = 1 - \frac{1}{q} \cdot \sum_{i=1}^{n} \sum_{j=1}^{k} \frac{|f(i, ln_{i,j}) - j|}{k} \tag{2.10}$$

2.4 Considerações finais

Nesta seção, foram apresentados fundamentos essenciais para o entendimento do método proposto. Tendo conhecimento de como os textos são representados em um espaço multidimensional, do conceito de projeção multidimensional e algoritmos utilizados para realizar as projeções, podemos entender como visualizar textos utilizando gráficos de espalhamento e como avaliar a qualidade da projeção através das métricas apresentadas.

Capítulo 3

Revisão de literatura

Nesta seção, serão apresentados trabalhos relacionados à análise de técnicas de projeção multidimensional, representação e visualização de textos. Os artigos aqui citados apresentam conceitos e técnicas utilizadas neste trabalho, além de abordagens diferentes para a visualização de textos e projeção multidimensional, e complementam este estudo proporcionando um entendimento melhor das técnicas e aprofundamento maior em diferentes aspectos dos tópicos abordados.

Em Paulovich [15] são apresentados conceitos básicos de projeção multidimensional e diversas técnicas, como PCA [33] e FastMap [34], são comparadas. Técnicas clássicas são comparadas projetando diferentes tipos de vetores multidimensionais, através de um modelo de análise que serve de base para este e outros trabalhos. As técnicas de projeção são analisadas utilizando métricas como Preservação de vizinhança e observação do gráfico de espalhamento.

Uma análise de técnicas de projeção multidimensional mais modernas é encontrada em Wang et al. [35]. O trabalho apresenta diferentes métricas para avaliar os resultados obtidos por cada técnica e compara gráficos de espalhamento produzidos a partir de conjuntos de dados diferentes, com foco em entender como cada técnica de projeção funciona. No artigo são analisados o t-SNE, UMAP e o TriMap [36]. A partir dos resultados obtidos uma nova técnica de redução de dimensionalidade é proposta, o *Pairwise Controlled Manifold Approximation Projection (PaCMAP)*.

No artigo McInnes et al. [17], onde é apresentada a técnica de projeção UMAP, o UMAP é comparado com diversas outras técnicas, focando em comparações com o t-SNE e também apresentando resultados do LargeVis [37], Eigenmaps [38] e Isomaps [39]. O artigo mostra gráficos de espalhamento e uma análise quantitativa da preservação de vizinhança utilizando *k-nearest neighbor classifier*. São apresentados resultados obtidos a partir de vetores de texto, porém este não é o foco do artigo, portanto, somente um conjunto de vetores que utiliza *skip-gram* é analisado.

Com foco em visualização de textos, Shusen et al. [40] apresenta uma ferramenta para representar visualmente relações entre palavras em um espaço bidimensional. Na ferramenta apresentada, grupos de palavras são separados e comparados como analogias, e projetados em um espaço bidimensional que visa destacar as relações de analogia entre estes grupos, utilizando PCA e t-SNE.

Apesar de mostrarem layouts de projeções multidimensionais obtidas a partir de vetores de textos, nenhum trabalho apresentado nesta seção tem como foco a visualização deste tipo de dado, ou analisa diferentes formas de representação de textos. Neste trabalho é pretendido contribuir para a área apresentando resultados de projeções obtidas a partir de diferentes técnicas modernas de *text embeddings* e projeção multidimensional.

Capítulo 4

Metodologia proposta

Neste capítulo serão apresentados os passos utilizados para cumprir os objetivos propostos nessa monografia. Serão apresentados o conjuntos de textos utilizados, algoritmos e métricas aplicadas.

Na Figura 4.1 é apresentado um fluxograma simplificado das etapas que constituem a metodologia proposta. Cada uma das etapas será detalhada nas subseções seguintes e os resultados obtidos através deste processo são apresentados no Capítulo 5.

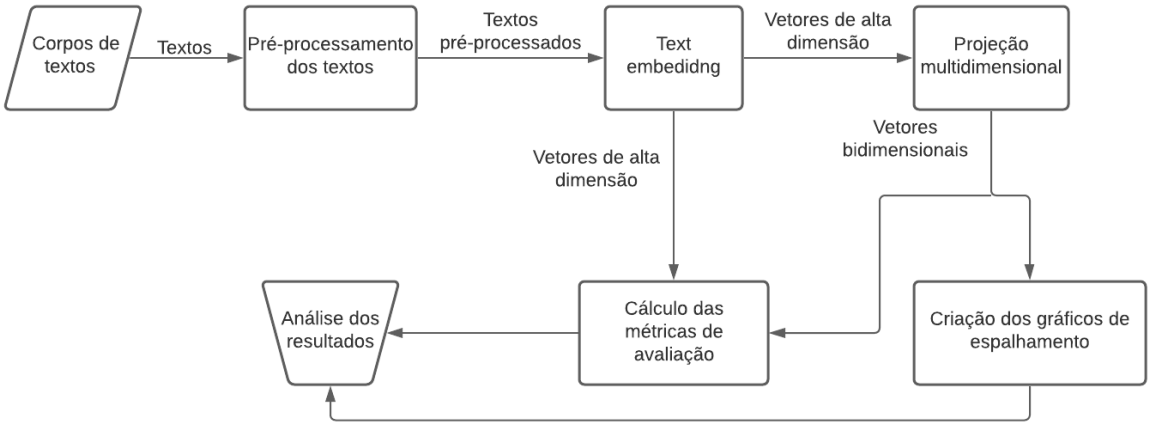


Figura 4.1: Etapas do estudo.

4.1 Corpos de textos

Foram utilizados três conjuntos de corpos de textos diferentes neste estudo. Cada corpus contém um conjunto de textos variados e foi utilizado para verificar diferentes comportamentos das técnicas de projeção multidimensional e *text embeddings*. Os três conjuntos são apresentados a seguir.

O primeiro conjunto é uma coleção de 1 milhão de *tweets* variados [41], postagens de até 128 caracteres da rede social Twitter, sem classificação. O conjunto completo foi utilizado no treinamento das técnicas que exigem este passo, porém a redução de dimensionalidade e análise foi realizada somente sobre os 30 mil primeiros *tweets*. Neste estudo, esse corpus será referenciado como *Tweets*.

O segundo conjunto contém cerca de 10 mil *tweets* variados [42], onde cada *tweet* é rotulado por uma polaridade: neutro, positivo ou negativo, para verificar o agrupamento de sentimentos similares no espaço bidimensional projetado. Esse corpus é referenciado como *Labeled Tweets*.

O terceiro conjunto de dados é o *20 newsgroups* [19], referenciado como *News* nesta monografia, um conjunto de cerca de 11 mil mensagens de fóruns sobre temas específicos, extraídas de *newsgroups* e divididas em 20 categorias diferentes. Este conjunto contém textos maiores que *tweets* e foi utilizado para verificar a preservação dos grupos de textos conforme o tema o abordado, após redução de dimensionalidade.

4.2 Pré-processamento de textos

Na metodologia proposta, foram aplicados diferentes pré-processamentos dependendo da técnica de *text embedding* utilizada. Para a técnica Vetor de Parágrafos foram utilizados os seguintes processamentos:

- Remoção de *stop-words*: foram consideradas as *stop-words* do conjunto de palavras da língua inglesa da biblioteca nltk [43].
- Stemmização: foi utilizado o *Snowball Stemmer* [44]. A lematização não foi empregada, pois não é necessário obter palavras que pertencem à linguagem do texto. Lembrando que a stemmização reduz o vocabulário suficientemente e é computacionalmente mais eficiente do que a lematização.
- Remoção de caracteres: foram removidos os caracteres de pontuação apresentados no Capítulo 2 e também palavras com mais de 100 caracteres.

Para utilização do BERT [7], foi aplicado somente o *WordPiece* [23] para obter a lista de tokens pré-processados. Em textos com mais de 512 tokens foram utilizados somente os últimos 512 tokens do texto. Para o Vetor de Parágrafos, os textos foram convertidos em listas de tokens, em que cada item da lista é um token gerado pelos pré-processamentos. As listas de tokens são consumidas no próximo passo, para a criação dos vetores numéricos.

4.3 Text Embeddings

Foram utilizados duas técnicas *Text Embeddings* neste estudo, Vetor de Parágrafos e RoBERTa. Ambas as técnicas geram representações vetoriais, onde a posição no espaço guarda informação semântica sobre os textos representados e a similaridade entre textos pode ser avaliada pela distância cosseno.

Foi utilizada a otimização RoBERTa [24] do modelo BERT nos experimentos realizados. Este modelo foi escolhido pois apresenta bons resultados para tarefas gerais em processamentos de textos [24], podendo ser utilizado para classificação e tradução, por exemplo. Foram realizados testes utilizando a rede pré-treinada sobre os conjuntos *roberta base* e *roberta large*, apresentados em Yinhan et al. [24]. Não foi aplicado o passo pós treinamento de refinamento dos vetores, pois o foco não consiste em utilizar os vetores para uma tarefa específica, mas sim obter vetores que contém mais informação sobre o texto representado, para então comparar os layouts para visualização de textos gerados pelas estratégias Vetor de Parágrafos e BERT.

A variação do Vetor de Parágrafos utilizada foi o PV-DBOW, pois apesar da PV-DM apresentar melhores resultados na representação geral de textos [6], para pequenos textos, como os analisados neste estudo, é possível obter bons resultados utilizando Vetor de Parágrafos com *bag-of-words* distribuída. Além disso, como o PV-DBOW é computacionalmente mais eficiente que o PV-DM, foi possível executar mais iterações de *backpropagation* para obter vetores mais refinados.

4.4 Projeção multidimensional

As técnicas de projeção multidimensional analisadas, t-SNE e UMAP, foram testadas utilizando a distância euclidiana como métrica de dissimilaridade. Para isto, os vetores originais foram normalizados para norma 1, assim mantendo a distância cosseno, que é utilizada para medir similaridade entre vetores de texto, proporcional à distância euclidiana.

A Equação 4.1 apresenta o cálculo da função de distância cosseno entre os vetores x e y, de mesma dimensionalidade. Na Equação 4.3 é demostrado que a distância cosseno é proporcional à distância euclidiana de vetores com norma 1.

$$cosD(x, y) = \frac{x \cdot y}{||x|| * ||y||} \tag{4.1}$$

$$\sqrt{x \cdot x} = ||x|| = 1 \tag{4.2}$$

$$
\begin{aligned}
edist(x,y)^2 &= ||x-y||^2 \\
&= (x-y)\cdot(x-y) \\
&= x\cdot x - 2*(x\cdot y) + y\cdot y \\
&= 2 - 2*(x\cdot y) \\
&= 2 - 2*\frac{x\cdot y}{||x||*||y||} \\
&= 2 - 2*cosD(x,y)
\end{aligned}
\tag{4.3}
$$

Nas visualizações baseadas em t-SNE, foram analisadas as variações Barnes-Hut t-SNE (BH t-SNE) e FIt-SNE, uma vez que possuem complexidade de tempo e memória baixas quando comparadas ao t-SNE original. Para o UMAP a versão referência presente em McInnes et al. [17] foi utilizada, já que esta apresenta uma implementação eficiente do algoritmo em complexidade de tempo e memória.

4.5 Cálculo das métricas de avaliação

A partir dos resultados gerados nas etapas anteriores, sobre os vetores gerados a partir de cada técnica de projeção e cada conjunto de dados, foi calculado a Preservação de vizinhança e Diferença de posição relativa.

Os resultados obtidos por estas métricas são divididos de acordo com parâmetros utilizados, sendo elas utilizadas tanto para avaliar a estrutura local como a estrutura global preservadas nas projeções multidimensionais.

4.6 Gráficos de espalhamento

A partir dos vetores bidimensionais foram gerados gráficos de espalhamento para cada resultado obtido. Os gráficos representam cada texto utilizando um ponto no espaço 2D. Os pontos no espaço visual foram coloridos de forma a apresentar a classe das instâncias, no caso de dados rotulados. Para dados não-rotulados, alguns pontos foram coloridos, rotulados e tiveram os textos que representam apresentados, para permitir verificar a qualidade do posicionamento de textos similares nas projeções.

4.7 Análise dos Resultados

Foi realizada uma análise a partir da observação e comparação dos resultados obtidos nas métricas e gráficos de espalhamento gerados. As comparações são realizadas entre combinações de diferentes técnicas de *text embeddings* e projeção multidimensional. Estes serão apresentados no próximo capítulo deste trabalho em detalhes.

Capítulo 5

Resultados experimentais

Neste capítulo serão apresentados os resultados para validar a metodologia proposta. Primeiramente serão descritas as implementações e os hiper-parâmetros utilizados em cada algoritmo. Em seguida, são apresentadas comparações baseadas nos valores obtidos pelas métricas de avaliação e os gráficos gerados por cada projeção são comparados entre si por meio de análise visual e métricas de qualidade. Por fim, serão comparados os resultados obtidos pelas métricas e gráficos. Todos os resultados e conjuntos de dados podem ser encontrados em `https://github.com/luisfbgs/Text-visualization`.

5.1 Hiper-parâmetros

5.1.1 Vetor de Parágrafos

Foi utilizada a implementação presente na biblioteca Gensim [45] para construir os Vetores de Parágrafos. Somente o modelo DBOW é analisado, com os parâmetros: *min_count=2*, *epochs=300* e *window=2*. Foram gerados vetores de 300 e 1000 dimensões para cada conjunto de textos considerado.

5.1.2 RoBERTa

Foi utilizada a implementação do modelo RoBERTa presente em *Fairseq* [46], com os modelos pré-treinados *roberta base* e *roberta large*. Não foi realizado nenhum refinamento sobre os vetores, para representar cada texto foram utilizados os vetores obtidos a partir da extração da última camada do modelo pré-treinado aplicado a cada texto.

5.1.3 t-SNE

Foi escolhida a implementação referência de Linderman et al. [26]. Foram testadas duas variantes do t-SNE, uma utilizando o *nbody_algo='Barnes-Hut'* e *knn_algo='vp-tree'* e outra utilizando *nbody_algo='FFT'* e *knn_algo='annoy'*.

Em ambas variações, os hiper-parâmetros foram ajustados como: *perplexity=20* e *theta=0.2*. Foram testados valores de *perplexity* entre 15 e 25, em que não foram percebidas grandes diferenças nas métricas de avaliação. Os layouts obtidos a partir do corpus *News* foram comparados com os resultados obtidos quando utilizado *perplexity=50*, mas não foram percebidas grandes diferenças no posicionamento dos pontos no espaço visual. O valor de *theta=0.5* foi testado, em que, apesar de uma execução mais rápida quando comparada ao valor *theta=0.2*, os resultados obtidos nas métricas de avaliação de qualidade foram inferiores. Por fim, valores de *theta* menores do que 0.2 resultam em uma execução mais lenta do t-SNE, ao passo em que resultados similares ao valor de *theta=0.2* puderam ser obtidos nas métricas testadas.

5.1.4 UMAP

A implementação referência apresentada em McInnes et al. [17] foi utilizada, com os hiper-parâmetros:

- *min_dist=0*, que trouxe melhores resultados nas métricas utilizadas neste estudo, e *min_dist=0.1* para verificar se as projeções são beneficiadas.
- O valor de *n_neighbors* foi escolhido para cada conjunto de vetores individualmente. Foram testados valores de *n_neighbors* entre 2 e 20 e o valor que gerou melhores resultados nas métricas de NBP e MRPD foi escolhido para cada conjunto de dados. Na prática, foram utilizados valores de *n_neighbors* entre 4 e 11.

5.2 Métricas MRPD e NBP

Nesta seção são apresentados os resultados obtidos a partir das métricas de avaliação de preservação de vizinhança MRPD e NBP. Os gráficos apresentados mostram o resultado médio obtido pelas projeções. Os resultados de RoBERTa apresentam as médias dos resultados medidos em layouts gerados a partir de vetores obtidos pelos modelos pré-treinados *roberta base* e *roberta large*. Nos gráficos que apresentam resultados de Vetor de Parágrafos, o valor de média apresentado agrupa os resultados de vetores de 300 e 1000 dimensões. Os resultados denominados como *Tweets* agrupam os resultados dos corpus *Tweets* e *Labeled Tweets*.

5.2.1 Pequenas vizinhanças

Para analisar a preservação de pequenas vizinhanças, foram observados os resultados das métricas Preservação de Vizinhança (NBP) e Diferença de Posição Relativa (MRPD). O valor do parâmetro k, que determina, em ambas as métricas, o número que cada ponto possui em sua vizinhança, foi utilizado com valores intervalo $[1, 100]$.

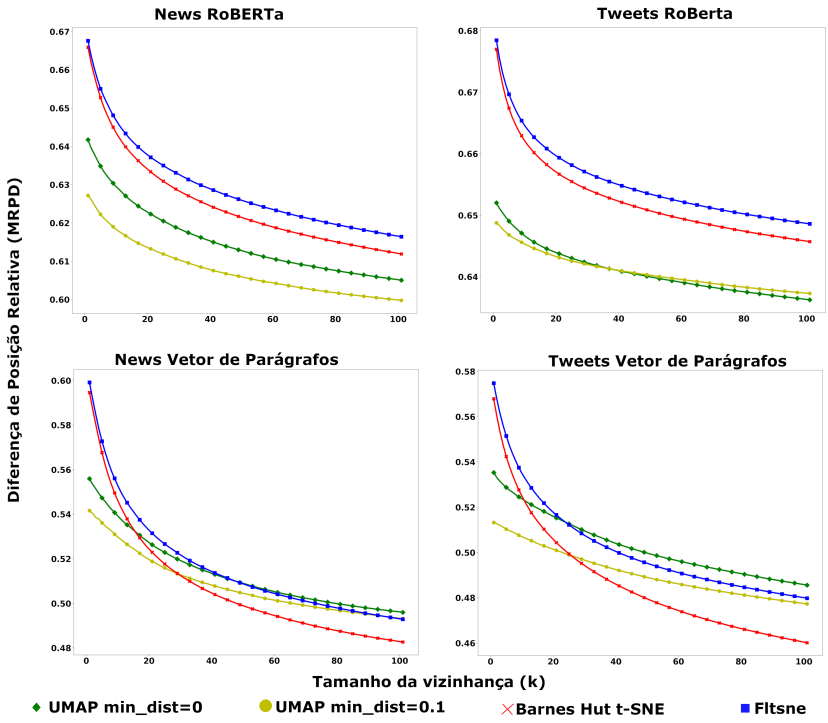


Figura 5.1: Comparação dos gráficos de MRPD, obtidos para valores de k no intervalo $[1, 100]$

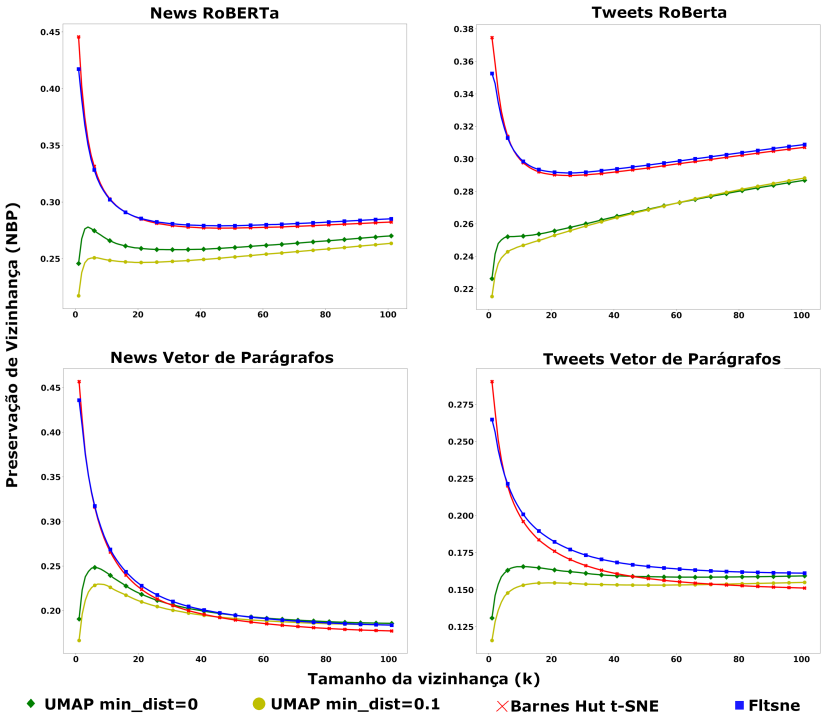


Figura 5.2: Comparação dos gráficos de NBP, obtidos para valores de k no intervalo $[1, 100]$

Quando observamos os resultados com base na métrica MRPD, na Figura 5.1, percebemos que o desempenho, em ambas as técnicas de projeção, diminui conforme aumentamos o tamanho da vizinhança considerada. Na NBP, Figura 5.2, os resultados obtidos são em grande parte similares aos vistos na MRPD. Nos layouts obtidos a partir do RoBERTa, principalmente dos grupos de *tweets*, o resultado da NBP melhora para vizinhanças maiores, a partir de 50 vizinhos. É esperado que os resultados da NBP sejam melhores para maiores valores de k, já que a avaliação se baseia na interseção entre conjuntos, e conforme estes conjuntos crescem, a chance de haver interseções é maior.

Para pequenas vizinhanças, considerando até 50 vizinhos por ponto, as técnicas de t-SNE apresentam melhores resultados que o UMAP. Entretanto, é possível observar que a diferença é menor nos layouts gerados a partir de Vetor de Parágrafos quando comparadas a layouts obtidos pela projeção do RoBERTa. Quando o tamanho da vizinhança supera 50 pontos, os resultados do UMAP são melhores em relação ao t-SNE para Vetor de Parágrafos, porém continuam piores quando o RoBERTa é utilizado.

Quando comparamos as variações das técnicas, é possível observar que o UMAP com *m_dist=0* sempre obteve melhores resultados do que quando utilizamos *m_dist=0.1*, com exceção dos *Tweets* com RoBERTa e *News* com Vetor de Parágrafos, onde ambos apresentam resultados similares. O FIt-SNE obteve resultados bem próximos, mas melhores que o Barnes-Hut t-SNE, em que a diferença é maior quando observamos vizinhanças

maiores. O Barnes-Hut t-SNE obteve vantagem em relação ao FIt-SNE somente quando observamos grupos de *Tweets* representados com Vetor de Parágrafos.

5.2.2 Grandes vizinhanças

Para avaliar a qualidade das projeções multidimensionais em grandes vizinhanças, foram utilizadas as mesmas métricas como nas pequenas vizinhanças, em que somente o valor de k foi alterado, utilizando valores no intervalo $[101, 600]$. O limite 600 foi escolhido, uma vez que este é o tamanho do maior grupo do conjunto de textos *News*.

Nos gráficos que apresentam os resultados em grandes vizinhanças das métricas MRPD e NBP, nas Figuras 5.3 e 5.4, respectivamente, é possível observar que a situação é similar às pequenas vizinhanças quando consideramos os vetores gerados pelo RoBERTa, com as técnicas baseadas em t-SNE obtendo resultados melhores que o UMAP. Os resultados obtidos a partir de Vetor de Parágrafos utilizando a técnica UMAP se tornam melhores comparados ao t-SNE, quando aumentamos a vizinhança.

A visualização por meio da técnica UMAP com *min_dist=0* apresenta resultados melhores em relação a *min_dist=0.1*, com exceção do grupos de *Tweets* com RoBERTa, em que utilizar *min_dist=0.1* gerou melhores resultados. Quando comparamos as variações do t-SNE em consideração, o FIt-SNE possui vantagem em todos os cenários testados.

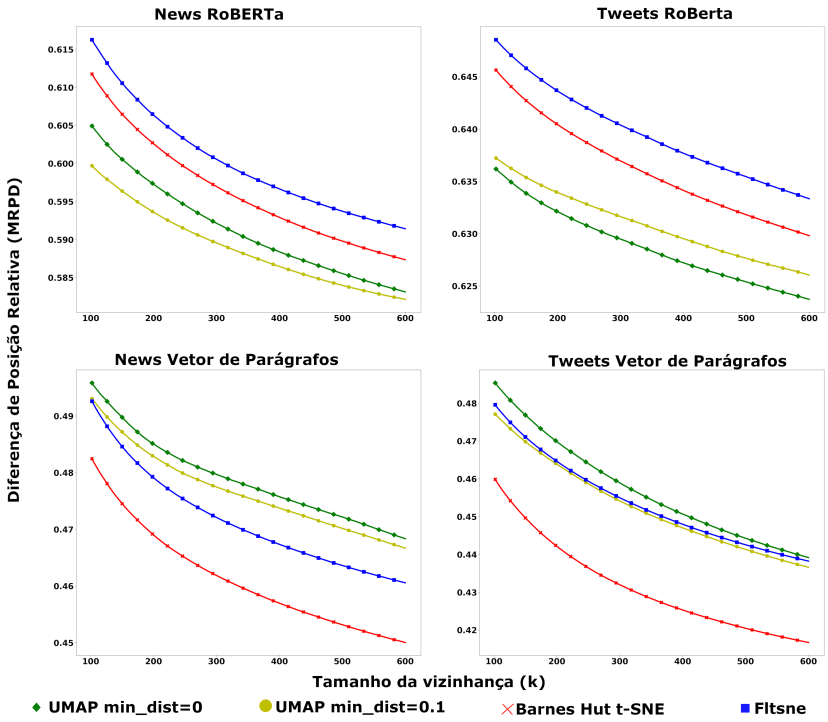


Figura 5.3: Comparação dos gráficos de MRPD, obtidos para valores de k no intervalo $[101, 600]$

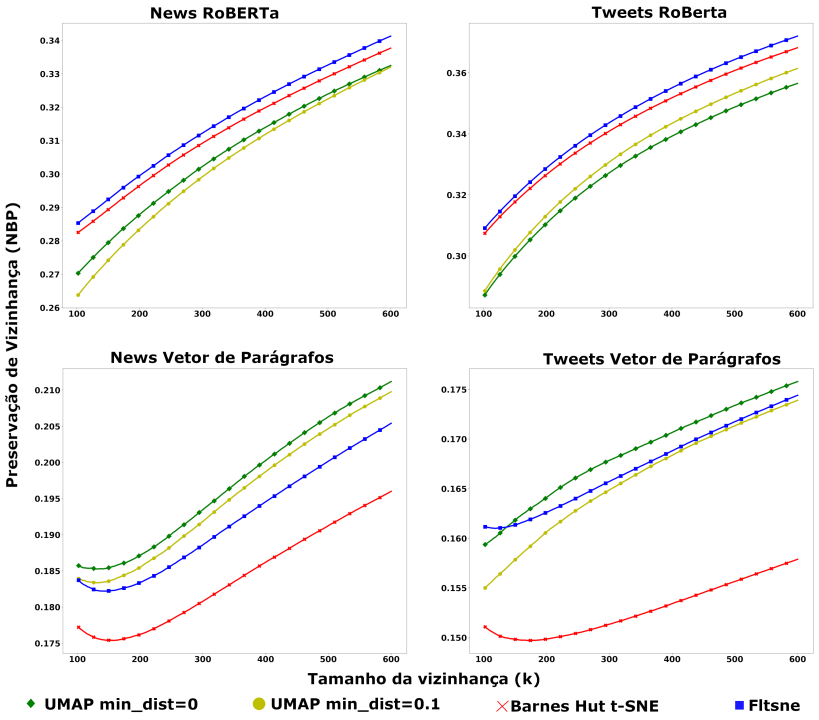


Figura 5.4: Comparação dos gráficos de NBP, obtidos para valores de k no intervalo $[101, 600]$

5.3 Gráficos de espalhamento

5.3.1 News

Nesta seção, serão analisados alguns gráficos de espalhamento obtidos a partir das projeção do conjunto de textos *News*. Foram utilizados todos os 20 grupos presentes no corpus, em que um número foi associado a cada categoria, como mostrado na Tabela 5.1. A cada número foi associada uma cor que representa a categoria, as cores são indicadas na Figura 5.5. Nesta seção, os grupos serão referenciados pelos números da Tabela 5.1.

1.alt.atheism	2.comp.graphics	3.comp.os.ms-windows.misc
4.comp.sys.ibm.pc.hardware	5.comp.sys.mac.hardware	6.comp.windows.x
7.misc.forsale	8.rec.autos	9.rec.motorcycles
10.rec.sport.baseball	11.rec.sport.hockey	12.sci.crypt
13.sci.electronics	14.sci.med	15.sci.space
16.soc.religion.christian	17.talk.politics.guns	18.talk.politics.mideast
19.talk.politics.misc	20.talk.religion.misc	

Tabela 5.1: $x.g$ indica que a categoria g é representada pelo número x.

Figura 5.5: Legenda adotada para associar as cores às categorias do corpus *News*.

Vetor de Parágrafos

Nesta seção, serão apresentados os gráficos obtidos a partir do *text embedding* Vetor de Parágrafos. Foram realizados testes utilizando vetores de 300 e 1000 dimensões, mas serão apresentados apenas os resultados das projeções dos vetores de 300 dimensões, pois, para ambas as dimensionalidades, os resultados foram similares.

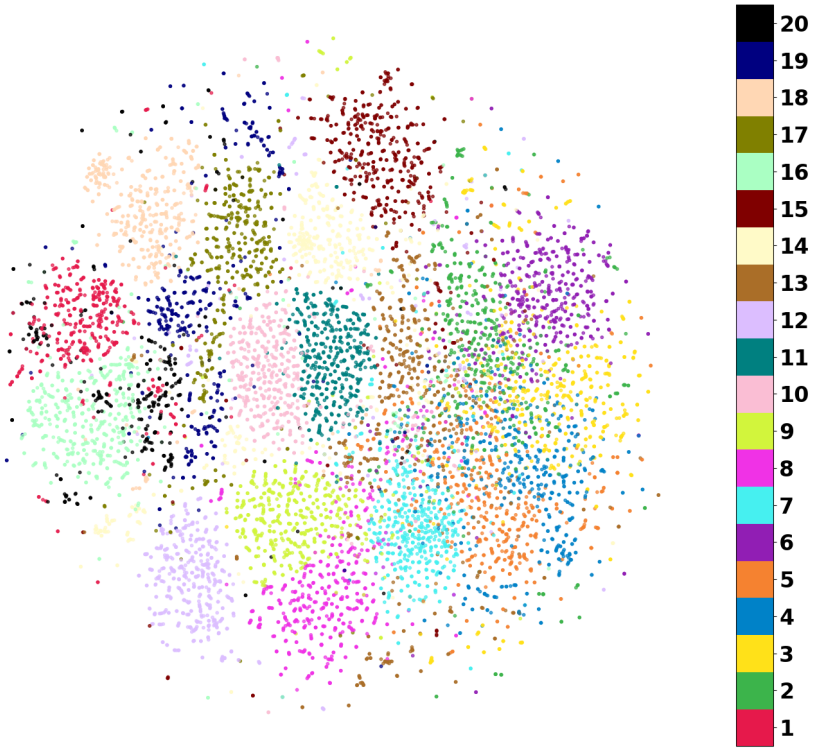


Figura 5.6: Layout obtido pela projeção FIt-SNE, para o corpus *News* a partir de Vetor de Parágrafos 300D.

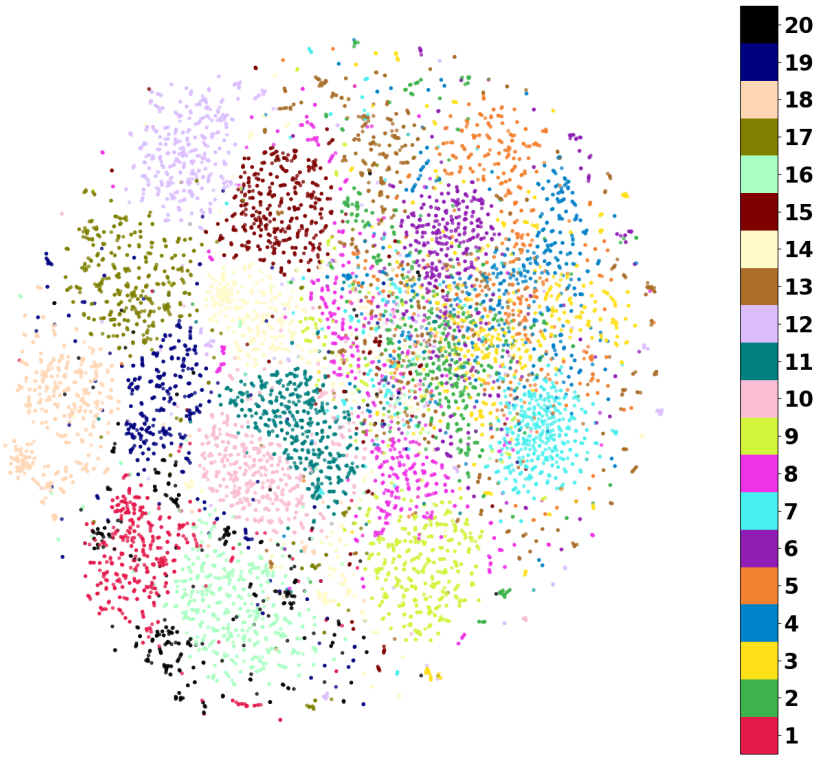


Figura 5.7: Layout obtido pela projeção BH t-SNE, para o corpus *News* a partir de Vetor de Parágrafos 300D.

As Figuras 5.6 e 5.7 apresentam os layouts obtidos pelas técnicas FIt-SNE e BH t-SNE, respectivamente. Quando observamos a separação de grupos ambas técnicas apresentam resultados similares, com um bom agrupamento de textos de mesma categoria. As categorias 2, 3, 4 e 5 estão aglutinados em um único espaço em ambas as projeções, porém com uma separação melhor definida na FIt-SNE. A categoria 8 é melhor isolada na FIt-SNE. No layout obtido pelo BH t-SNE, as categorias 17 e 18 estão bem separadas, enquanto que na FIt-SNE alguns pontos dessas categorias se misturam. A categoria 20 tem seus pontos espalhados entre as categorias 1 e 16 em ambos os layouts.

Quando observamos a relação entre categorias diferentes, o resultado também é similar em ambas as técnicas. Diferentes grupos próximos apresentam temas similares, as categorias 2, 3, 4 e 5, que não possuem uma boa separação entre si, abordam temas de computação, por exemplo. A maior diferença entre o posicionamento dos grupos está na categoria 12, que está mais próximo das categorias 15 e 17 na BH t-SNE, enquanto na FIt-SNE as categorias mais próximos são os 8 e 9.

A análise visual dos layouts obtidos pelas abordagens baseadas em t-SNE permite observar que, em um cenário onde as categorias do conjunto de dados são desconhecidas, não seria possível obter grupos de pontos com boa separabilidade no layout. Como os

layouts apresentam os pontos bem distribuídos e áreas com densidade de pontos similares, é difícil determinar onde existe a separação entre um grupo e outro.

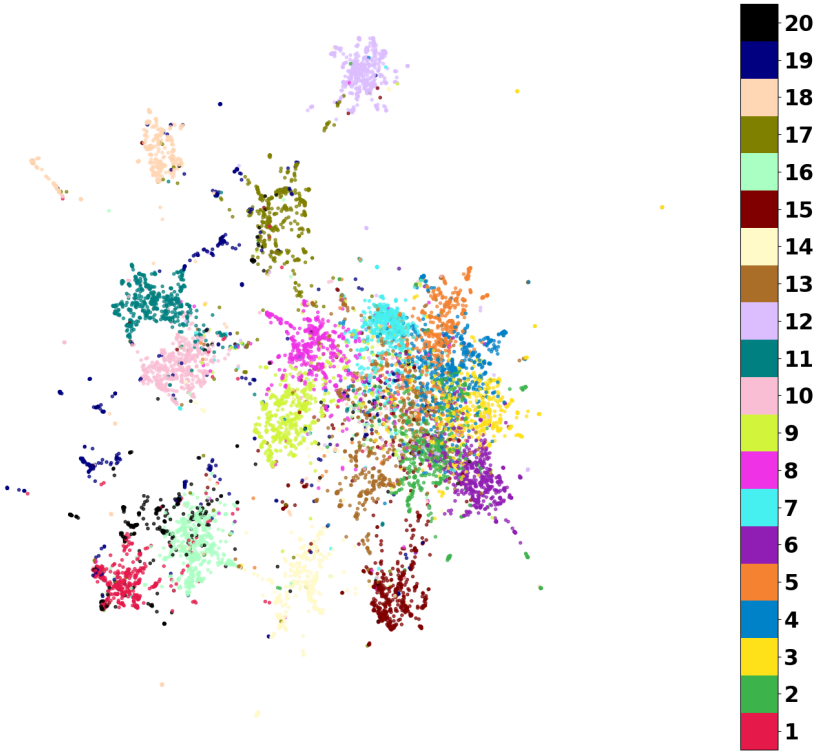


Figura 5.8: Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de Vetor de Parágrafos 300D.

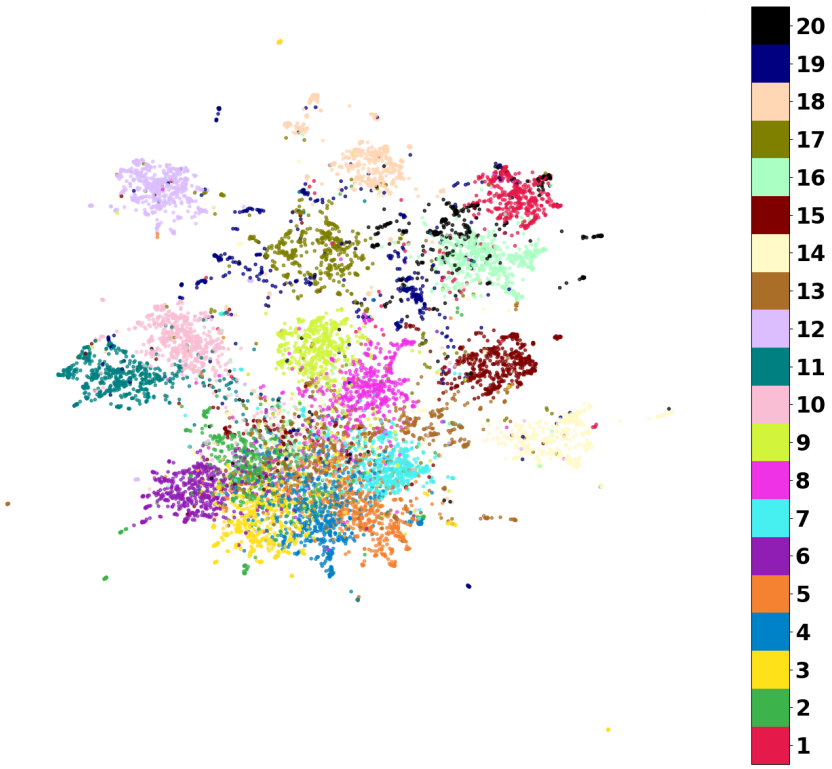


Figura 5.9: Layout obtido pela projeção UMAP com *min_dist=0.1*, para o corpus *News* a partir de Vetor de Parágrafos 300D.

As Figuras 5.8 e 5.9 apresentam os layouts obtidos pelo UMAP utilizando dois valores distintos para o parâmetro *min_dist*, 0 e 0.1, respectivamente. No layout obtido, a separação entre os grupos é boa em ambas as variações do UMAP, sendo que o UMAP com *min_dist=0.1*, o layout apresenta grupos de pontos mais próximos e pontos mais esparsos dentro de um mesmo grupo. Assim como no t-SNE, as categorias 2, 3, 4 e 5 se misturam nas projeções UMAP. O grupo da categoria 19 não está bem definido em nenhum layout, porém está bem atrelado à categoria 17, principalmente no layout obtido com *min_dist=0.1*. A categoria 20 tem seus pontos distribuídos entre os grupos das categorias 1 e 16 em ambos os layouts.

Quando observamos a relação entre categorias, percebemos as maiores diferenças entre as projeções UMAP e t-SNE. Nas projeções UMAP, as categorias com temas mais similares também são posicionados mais próximos, assim como no t-SNE, porém os grupos são melhores isolados no UMAP, com regiões com densidade variada, o que permite identificar os diferentes grupos mesmo sem a coloração. Em ambas representações gráficas obtidas pela técnica UMAP, as categorias 2, 3, 4, 5, 6, 7, 8, 9, e 13 têm seus pontos posicionados próximos uns aos outros, enquanto que os outros grupos apresentam melhor separação, com conjuntos menores de grupos formando regiões isoladas, como os grupos das categorias 1, 16 e 20.

RoBERTa Base

Nesta seção, serão apresentados os resultados obtidos a partir das projeções dos vetores gerados pelo RoBERTa, quando utilizado o modelo pré treinado *roberta base*. Somente os resultados obtidos pelas projeções BH t-SNE e UMAP com *min_dist=0* são analisados, já que as diferenças entre as variações de t-SNE e UMAP são similares as apresentadas para Vetor de Parágrafos. O FIt-SNE apresenta resultados similares ao BH t-SNE, com alguns grupos menos definidos. No UMAP com *min_dist=0.1*, os resultados são similares ao UMAP com *min_dist=0*, porém com grupos mais esparsos.

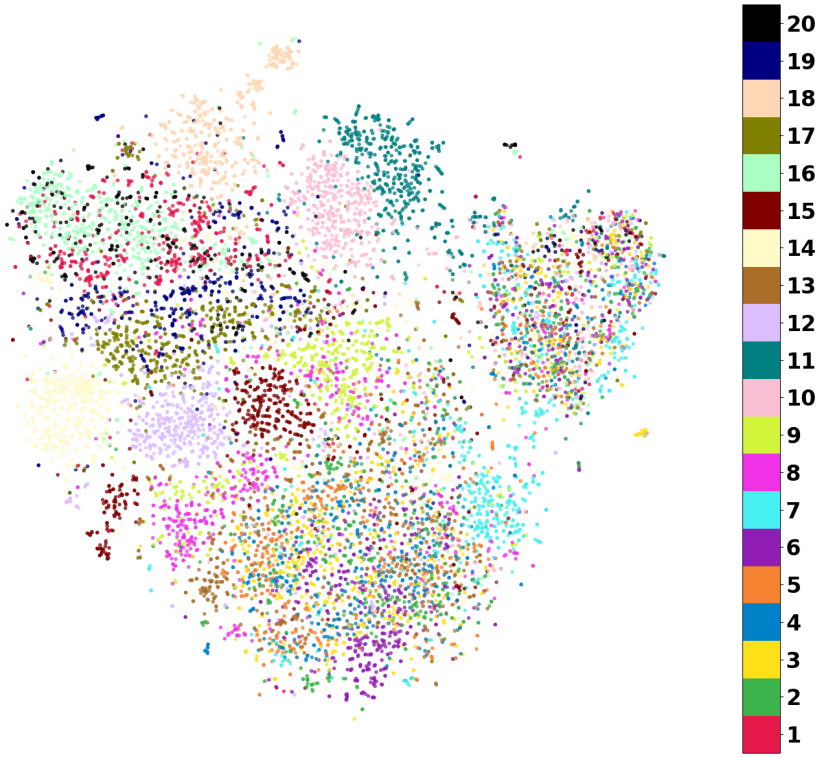


Figura 5.10: Layout obtido pela projeção BH t-SNE, para o corpus *News* a partir de vetores RoBERTa com modelo *roberta base*.

A Figura 5.6 ilustra o layout obtido pela projeção BH t-SNE para o corpus *News* representado pelo RoBERTa, com o modelo pré-treinado *roberta base*. Alguns grupos são apresentados bem isolados neste layout, mas na maior parte os grupos são pior definidos do que no layout obtido a partir de Vetor de Parágrafos, apresentados na Figura 5.7, com pontos próximo às bordas se misturando a grupos vizinhos. Os mesmos grupos que apresentavam pontos de diferentes categorias quando utilizado o Vetor de Parágrafos continuam com suas categorias indistinguíveis na representação gráfica. O grupo contendo textos das categorias 2, 3, 4 e 5, continua sendo o que apresenta as categorias com pior

separaação entre grupos de pontos. Na parte superior direita, um conjunto de pontos de temas variados formam um grupo de pontos isolado.

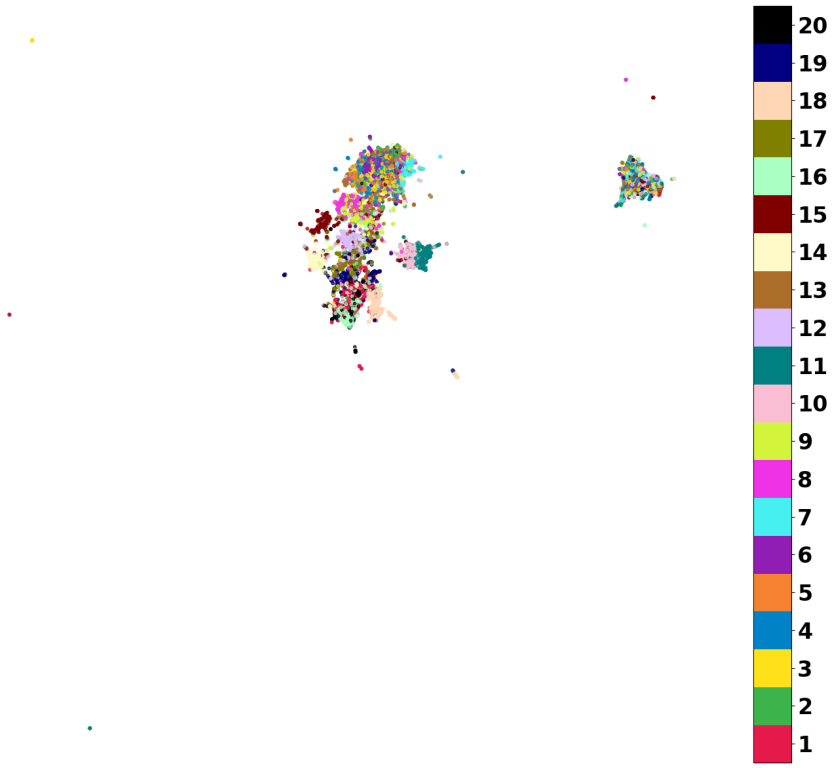


Figura 5.11: Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de vetores RoBERTa com modelo *roberta base*.

Na Figura 5.11 são apresentados os resultados do UMAP. Neste layout, alguns pontos aparecem bem espalhados pelo gráfico, com duas regiões com maior densidade de pontos, estas regiões são melhor visualizadas nas imagens 5.12 e 5.13, que apresentam uma visão mais próxima das regiões central e da parte direita, respectivamente.

A região central, mostrada na Figura 5.12, concentra a maior parte do pontos. Alguns grupos bem definidos, e mais isolados do que no t-SNE, com as mesmas categorias 2, 3, 4 e 5 aglutinadas em uma única região. Os grupos não aparecem bem definidos, quando comparados com os resultados obtidos a partir de Vetor de Parágrafos.

Na parte superior direita do layout obtido a partir dos vetores de RoBERTa utilizando UMAP, Figura 5.13, aparecem vários pontos de categorias diferentes. Nenhuma região na representação gráfica apresenta grande concentração de pontos, não sendo possível identificar grupos isolados, nem separação das categorias. Esse aspecto se assemelha à região superior direita do layout obtido pelo BH t-SNE a partir do *roberta base*, apresentada na Figura 5.10.

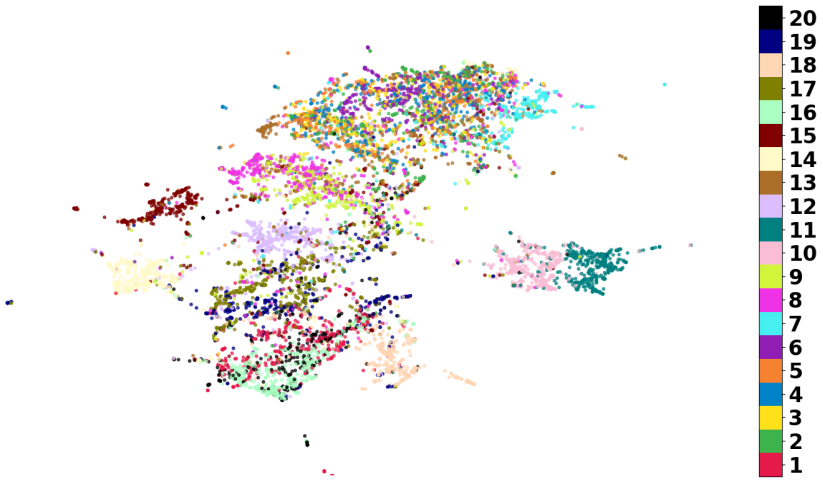


Figura 5.12: Região central do layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de vetores RoBERTa com modelo *roberta base*.

Figura 5.13: Região superior direta do layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de vetores RoBERTa com modelo *roberta base*.

Para investigar o motivo da presença de pontos isolados e grupo de pontos de categorias diferentes observados no layout, foram selecionados alguns pontos próximos para serem destacados nos gráficos. Os textos representados por cada ponto selecionado e seus respectivos identificadores são apresentados na Tabela 5.2. Os textos identificados como D e E são muito grandes para serem mostrados por completo na tabela e, por isso, foram extraídos alguns trechos que capturam a semântica dos textos e os trechos removidos dos textos foram substituídos pelo símbolo "[...]".

Identificador	**Tweet**
A / Categoria 12	From: hooper@ccs.QueensU.CA (Andy Hooper) Subject: Re: text of White House announcement and Q&As on clipper chip encryption Organization: Queen's University, Kingston Distribution: na Lines: 3 Isn't Clipper a trademark of Fairchild Semiconductor? Andy Hooper
B/ Categoria 4	From: passman@world.std.com (Shirley L Passman) Subject: help with no docs for motherboard Organization: The World Public Access UNIX, Brookline, MA Lines: 1 ,
C / Categoria 8	From: rgc3679@bcstec.ca.boeing.com (Robert G. Carpenter) Subject: Thinking About Buying Intrepid - Good or Bad Idea? Organization: Boeing Computer Services Lines: 7 I'm thinking of buying a new Dodge Intrepid - Has anyone had any experiences that they'd like to share? Thanks. BobC
D / Categoria 8	[...] Lines: 33 In rec.autos you write: >if ayrton senna can drive a racecar with fully automatic transmission, >it can't be half bad.. :-) This McLaren auto-transmission [...] servo motors, which do the shifting. That means, there is no power loss in the drivetrain (if You take out minimal mechanical friction), [...]
E / Categoria 9	[...] Subject: Re: Misc./buying info. needed [...] Lines: 28 [...] >Is there a pricing guide for new/used motorcycles [...] >Are there any books/articles on riding cross country, motorcycle camping, etc? [...] >Is there an idiotsǵuide to motorcycles? [...]

Tabela 5.2: Exemplos de textos do corpus *News*.

Na Figura 5.14, é apresentado o layout obtido pela UMAP do corpus *News*, utilizando *roberta base*. Neste layout, todos os pontos foram coloridos pela mesma cor, com exceção dos pontos que representam os textos destacados na Tabela 5.2, que mantém a cor de suas respectivas categorias. Os pontos A, B e C são sobrepostos, e por consequência, só é possível observar a cor do ponto C.

Os pontos D e E, apesar de não pertencerem a mesma categoria, tratam de assuntos similares, indicando que o posicionamento de ambos parece adequado. Os A, B e C possuem algumas palavras chaves que podem ter contribuído para serem posicionados próximos, como "*Semicondutor*", "*motherboard*" e "*Computer*", nos textos A, B e C, respectivamente. Também vale ressaltar que os três textos são consideravelmente menores que os textos D e E. Como no BERT somente os últimos 512 *tokens* são considerados, o cabeçalho, que é similar em todos os textos do corpus, é levado em consideração pelo *text embedding* somente em textos pequenos.

Para verificar se o cabeçalho é a característica que determina quais pontos são isolados no grupo mostrado na Figura 5.13, foi calculado o número médio de palavras dos textos, uma média menor indica que mais textos tiveram seus cabeçalhos considerados pelo RoBERTa na construção dos vetores que os representam. No grupo de pontos da região central, onde estão a maioria dos pontos, o número médio de palavras dos textos é 336, no grupo da direita a média é de 215 palavras por texto e nos demais pontos, espalhados pela área esquerda do layout, o número médio de palavras é de 553 por texto.

O texto de alguns dos pontos isolados na região esquerda do layout da Figura 5.14 foram verificados. Os textos observados apresentam uma sequência de caracteres aleatórios, possivelmente obtidos a partir do conteúdo binário de arquivos, ou *ASCII Art* no final do texto.

Figura 5.14: Layout com marcações, obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de vetores RoBERTa com modelo *roberta base*.

Na Figura 5.15 são apresentados os pontos A, B, C, D e E marcados no layout obtido a

partir da projeção UMAP aplicada a Vetor de Parágrafos de 300 dimensões. Neste layout, os pares de pontos B, C e D, E são representados próximos, porém não tão próximos quanto na Figura 5.14. O ponto A, que antes aparecia próximo aos ponto B e C, aparece isolado dos outros pontos na Figura 5.15, em um grupo de pontos da categoria 12, como visto na Figura 5.8.

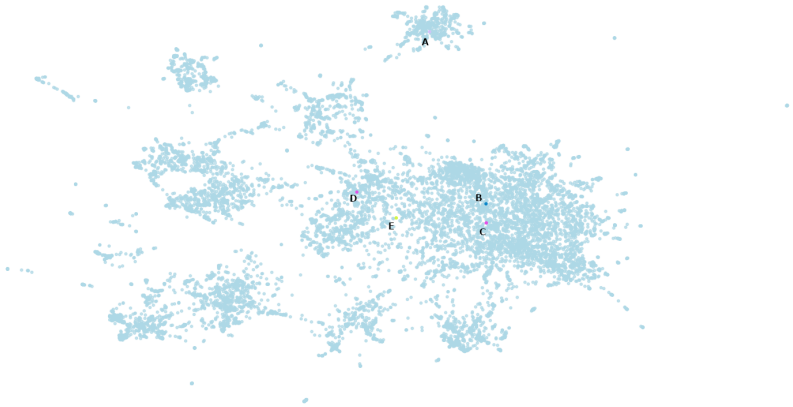


Figura 5.15: Layout com marcações, obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de Vetor de Parágrafos de 300 dimensões.

RoBERTa Large

Nesta seção, serão apresentados os resultados obtidos a partir das projeções dos vetores gerados pelo RoBERTa quando utilizado o modelo pré treinado *roberta large*. Serão apresentados somente os resultados obtidos pelas projeções BH t-SNE e UMAP com *min_dist=0*.

A Figura 5.16 apresenta o resultado obtido pelo BH t-SNE, em que a maior parte dos pontos estão distribuídos uniformemente por todo o layout, com poucas regiões de densidade variada e pequenas aglomerações de pontos de mesma categoria espalhadas pelos gráficos. Este layout não apresenta isolamento de grupos de texto de mesmo tópico, como também a projeção não captura satisfatoriamente as relações entre os textos. O resultado obtido pelo Fltnse é similar ao obtido pelo BH t-SNE.

A Figura 5.17(a) apresenta o layout obtido pela projeção UMAP, em que alguns pontos estão isolados e uma região central contém a maior parte dos pontos. Quando analisamos a região central do layout, na Figura 5.17(b), encontramos um cenário similar ao layout obtido pela técnica baseada em t-SNE. Nesse caso, a diferença refere-se ao fato de existirem algumas regiões com pouca densidade de pontos. Entretanto, pontos de grupos próximos não estão bem agrupados, não sendo possível identificar grupos bem isolados.

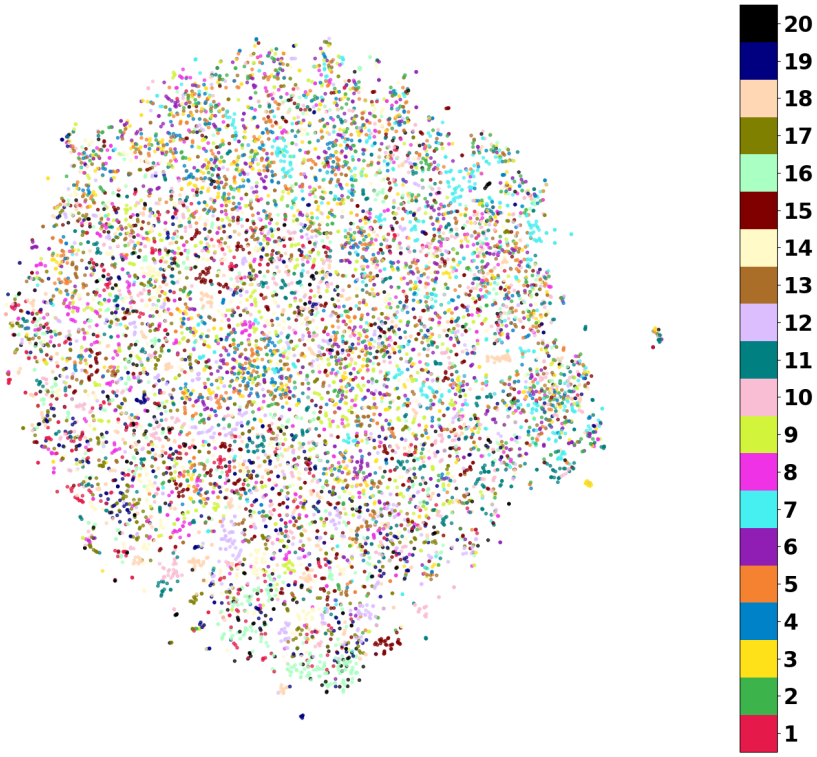
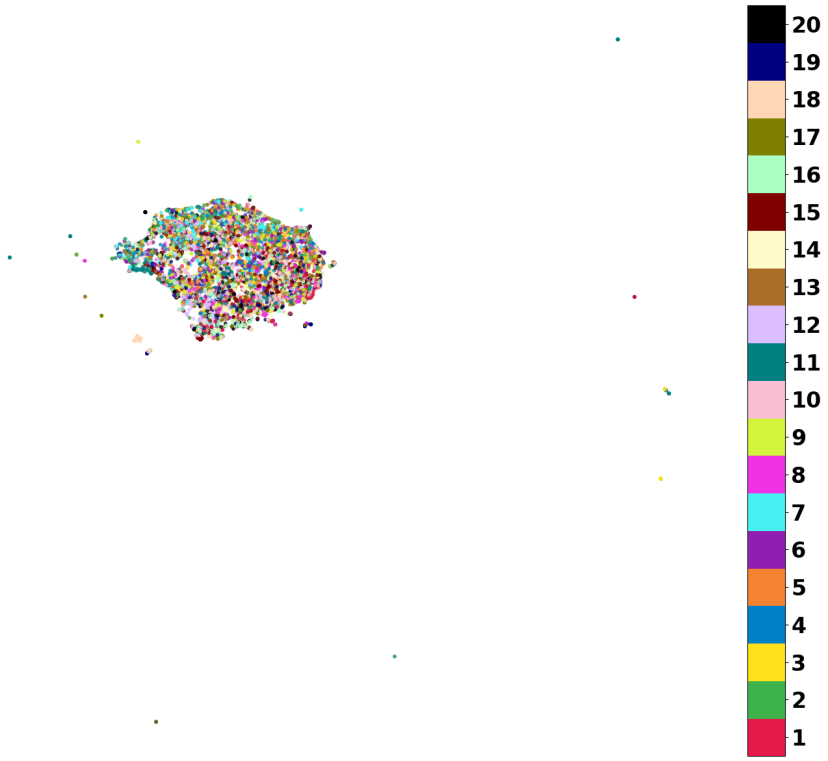


Figura 5.16: Layout obtido pela projeção BH t-SNE, para o corpus *News* a partir de vetores RoBERTa com o modelo *roberta large*.



(a) Gráfico completo.

(b) Região superior esquerda.

Figura 5.17: Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *News* a partir de vetores RoBERTa com o modelo *roberta large*.

5.3.2 Tweets variados

Nesta seção, serão apresentados os gráficos de espalhamento obtidos a partir dos 30 mil primeiros *tweets* do conjunto de dados *Tweets*, que contém pequenos textos, não rotulados, sobre assuntos variados. Foram omitidos os resultados obtidos por Vetor de Parágrafos

de 1000 dimensões e pelo BH t-SNE, pois estes se assemelham aos obtidos com vetores de 300 dimensões e pelo FIt-SNE, respectivamente. Somente resultados do UMAP com *min_dist=0* serão mostrados, pois as diferenças para o uso de *min_dist=0.1* são as mesmas encontradas no conjunto *News*.

Como os textos deste conjunto não possuem rótulos, foram selecionados algumas triplas de pontos próximos e estas foram marcadas nos layouts. As triplas selecionadas são identificadas por uma letra, A, B ou C, e um número, 1, 2 ou 3. Pares de mesma letra e números 1 e 2, como o par $(A1, A2)$, representam textos próximos no espaço de alta dimensionalidade gerado por Vetor de Parágrafos de 300 dimensões. Pontos com mesma letra e números 1 e 3, como o par $(A1, A3)$, representam textos próximos no espaço de alta dimensionalidade gerado pelo RoBERTa utilizando os vetores pré-treinados de *roberta large*. Na Tabela 5.3, são apresentados os textos que cada um dos pontos selecionados representam.

Identificador	**Tweet**
A1	Can't sleep. It's 2:05am–Ugh!!! I'm not even sleepy
A2	I should be so sleepy...but I can't sleep
A3	4.34am... Feels more like pm... Sleep is not my friend!!!
B1	i am sorry to hear about your grandma anything i can do for you?
B2	@tbauer254 sorry to hear
B3	@JBFutureboy when your album is gonna be released? sorry for the bad english
C1	Updated my twitter background and picture. Updating myspace next. Too bad Facebook does not let you customize and express your creativity
C2	why won't twitter let me change my picture?
C3	Added the DNS system, compatibility is okay and am looking forward to the possibilities now open. But I need to rework the intro sequence

Tabela 5.3: Textos representados pelos pontos marcados nos layouts do corpus *Tweets*.

A Figura 5.18 apresenta resultados obtidos pela projeção FIt-SNE, a partir de Vetor de Parágrafos de 300 dimensões. No gráfico, é possível observar que a maior parte dos pontos é bem distribuída, porém diversos pequenos grupos podem ser identificados em regiões de maior densidade, e alguns grupos isolados podem ser encontrados próximo às bordas do gráfico.

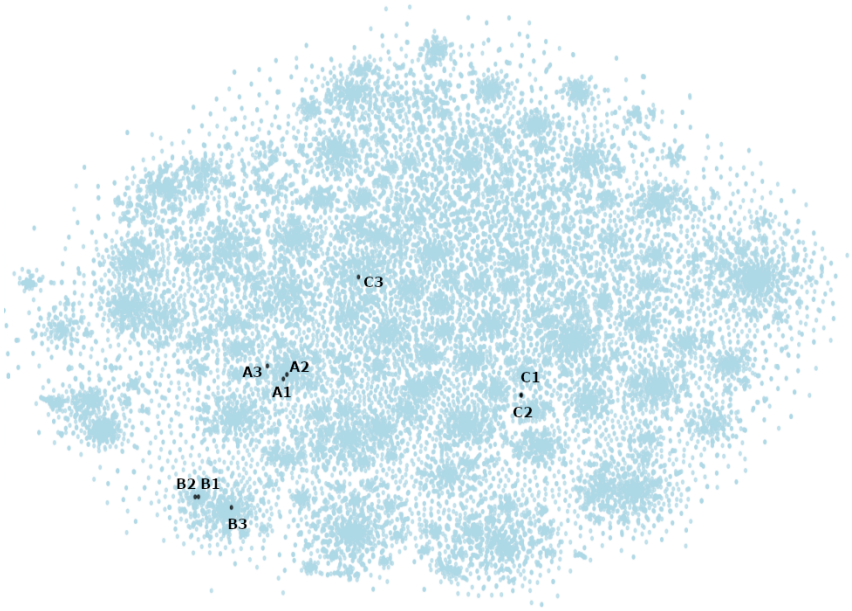
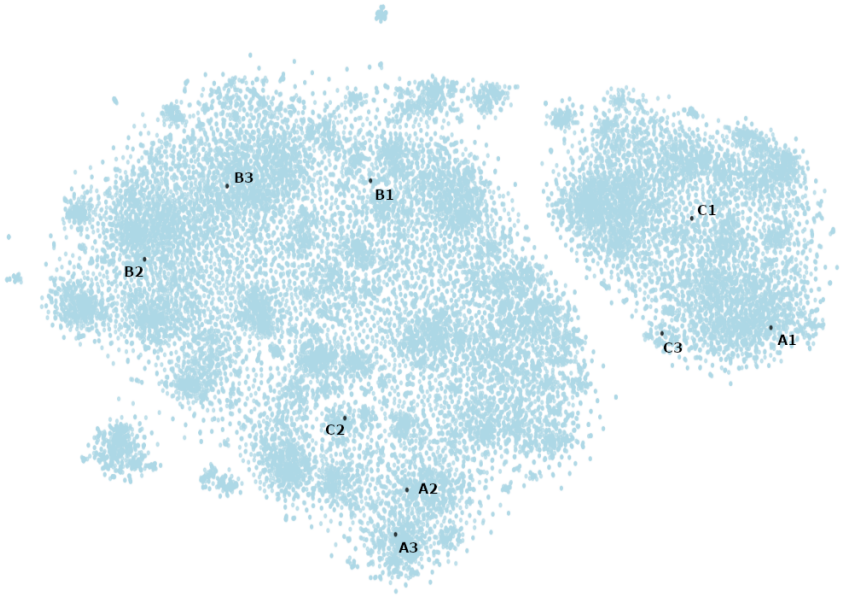
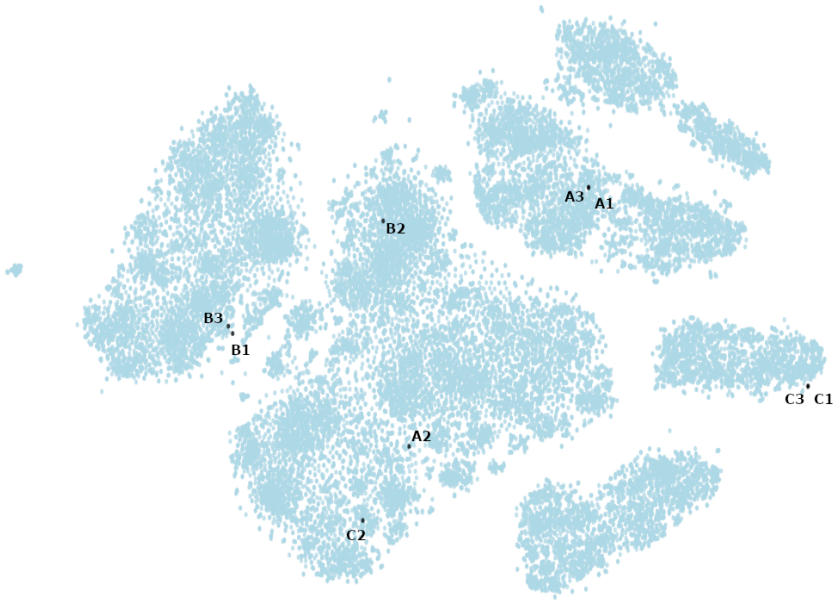


Figura 5.18: Layout obtido pela projeção FIt-SNE, para o corpus *Tweets* a partir de Vetor de Parágrafos 300D.

Na Figura 5.19(a) é apresentado o resultado da projeção de vetores obtidos a partir do RoBERTa, utilizando o pré-treinamento do *roberta base*. Nesse gráfico, assim como no caso de Vetor de Parágrafos, é possível identificar diversas pequenas regiões densas espalhadas pelo gráfico. Porém, aqui existem dois grandes grupos de pontos separados e bem definidos, e os pequenos grupos menores próximos às bordas do layout estão mais isolados.



(a) *roberta base*.



(b) *roberta large*.

Figura 5.19: Layout obtido pela projeção FIt-SNE, para o corpus *Tweets* a partir de vetores RoBERTa.

Ao analisar o resultado obtido pelo FIt-SNE sobre o RoBERTa utilizando o pré-treinamento *roberta large*, Figura 5.19(b), percebemos ainda mais grande grupos isolados do que no caso do *roberta base*. No layout apresentado, existem pequenas regiões mais densas, porém existem 6 grandes grupos bem definidos que não podiam ser observados nos layouts anteriores.

Figura 5.20: Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *Tweets* a partir de Vetor de Parágrafos 300D.

No resultado obtido pelo UMAP a partir de Vetor de Parágrafos, ilustrado na Figura 5.20, foram formados diversos pequenos grupos densos. A região central do layout é mais densa e possui menos grupos bem definidos. Nas regiões mais próximas as bordas do gráfico, é possível observar diversos pequenos grupos isolados, separadas por regiões pouco densas.

Nos resultados obtidos pela projeção a partir do RoBERTa, nas Figuras 5.21 e 5.22, existem grandes grupos de pontos concentrados e alguns poucos pontos isolados espalhados pelo layout. Quando utilizado o pré treinamento *roberta base*, foram formados menos grupos de pontos do que ao utilizar *roberta large*. Os grupos de *roberta base* também são menos densos que os gerados a partir do *roberta large*.

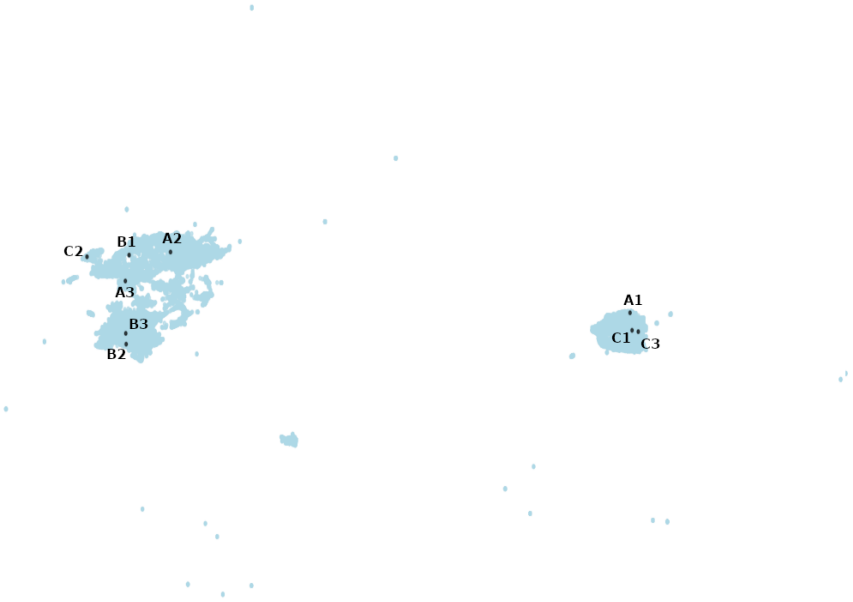


Figura 5.21: Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *Tweets* a partir de vetores RoBERTa com o modelo *roberta base*.

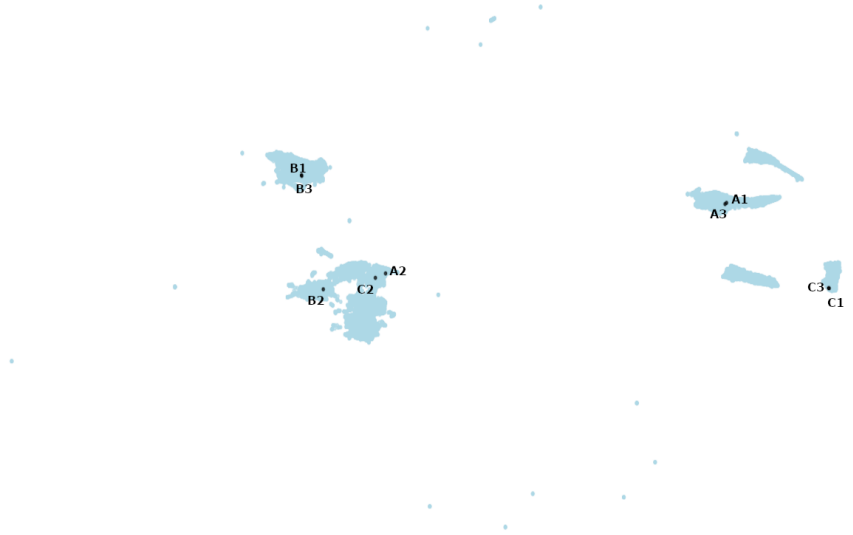
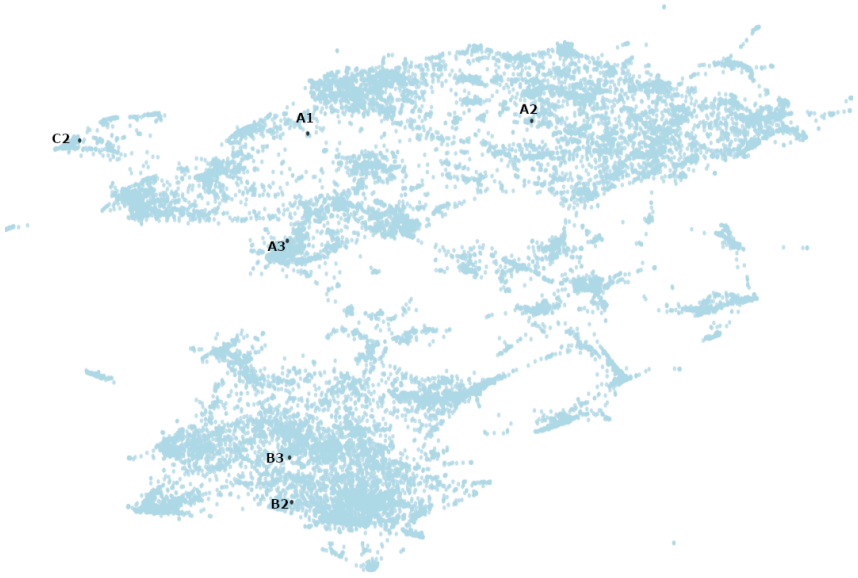


Figura 5.22: Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *Tweets* a partir de vetores RoBERTa com o modelo *roberta large*.

Na Figura 5.23, a análise local das regiões no layout obtido pela projeção UMAP e empregando a representação *roberta base*, é possível perceber que na região da direita, Figura 5.23(b), os pontos são mais esparsados, com algumas regiões mais densas. Na região da esquerda, apresentada na Figura 5.23(a), existe um grande grupo denso na parte inferior do layout e outro na região superior, com alguns grupos menos densos no meio do gráfico.



(a) Região esquerda.

(b) Região direita.

Figura 5.23: Layouts obtidos pela projeção UMAP com *min_dist=0*, para o corpus *Tweets* a partir de vetores RoBERTa com o modelo *roberta base*.

Na análise local do layout mostrado na Figura 5.24, obtido pela projeção UMAP e empregando a representação *roberta large*, percebe-se que existem grandes grupos de pontos, que são densos e separados entre si. Os grupos da região direita são bem definidos e isolados, enquanto que a região esquerda possui dois grandes grupos isolados, um na parte superior e outro na parte inferior, e os grupos de pontos na parte inferior se dividem em outros grupos menores.

(a) Região esquerda.

(b) Região direita.

Figura 5.24: Layouts obtidos pela projeção UMAP com *min_dist=0*, para o corpus *Tweets* a partir de vetores RoBERTa com o modelo *roberta large*.

5.3.3 Tweets rotulados por polaridade

Nesta seção, serão apresentados os resultados obtidos a partir do conjunto *Labeled Tweets*, onde cada texto é rotulado em uma dentre três categorias, de acordo com a polaridade do *tweet*, podendo ser negativo, positivo ou neutro. Foram utilizadas as cores 1, 2 e 3 mostradas ao lado de cada figura, para representar as polaridades negativa, neutra e positiva, respectivamente. O conjunto possui 1803 textos negativos, 5325 neutros e 4224 positivos.

Quando observamos os resultados de cada técnica de projeção e *text embeddings*, os resultados são similares aos observados no conjunto de *Tweets*, de textos sem rótulos. Ao analisar a distribuição de pontos pela polaridade, não é possível perceber nenhum padrão de agrupamento com base nesta classificação.

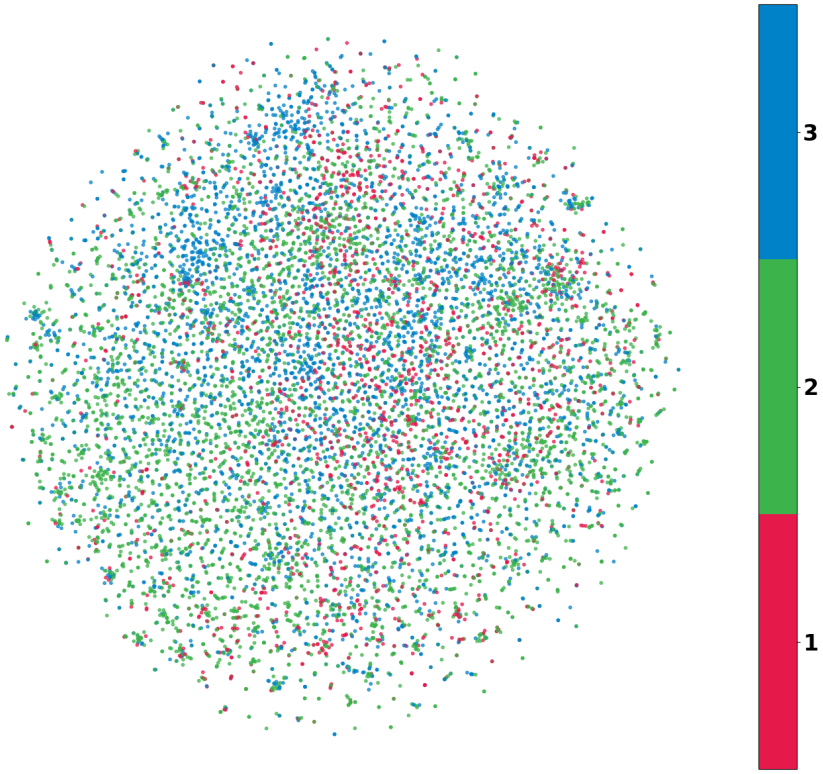
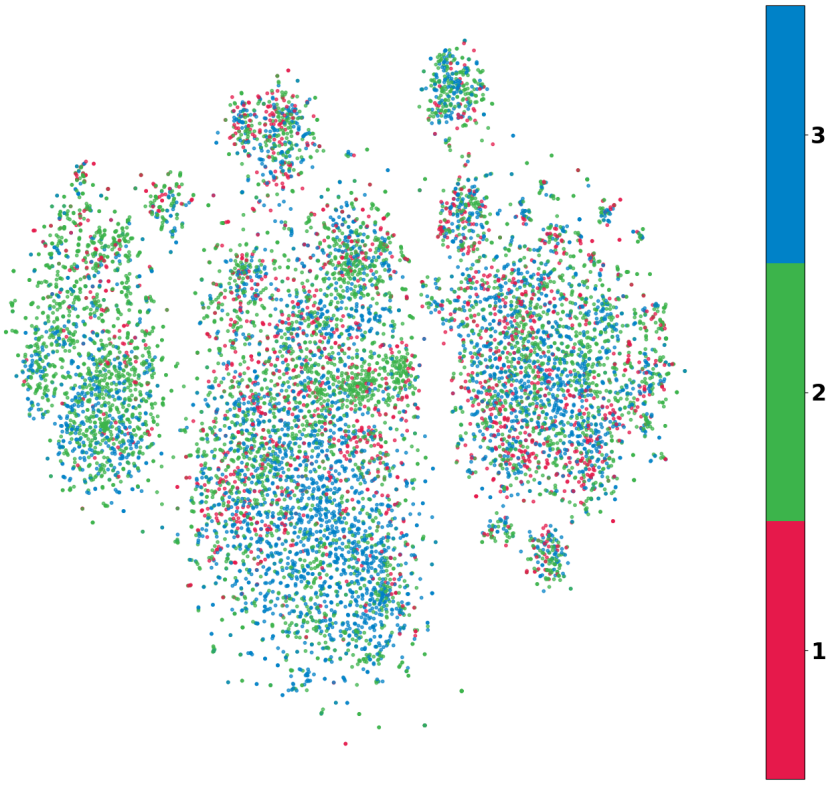
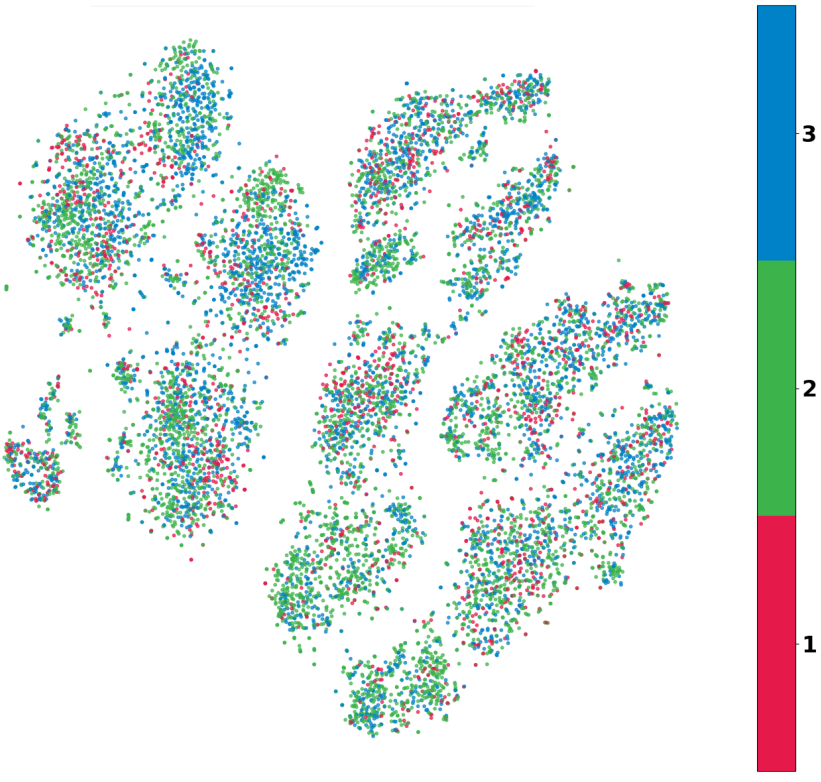


Figura 5.25: Layout obtido pela projeção FIt-SNE, para o corpus *Labeled Tweets* a partir de Vetor de Parágrafos 300D.



(a) *roberta base*.



(b) *roberta large*.

Figura 5.26: Layout obtido pela projeção FIt-SNE, para o corpus *Labeled Tweets* a partir de vetores RoBERTa.

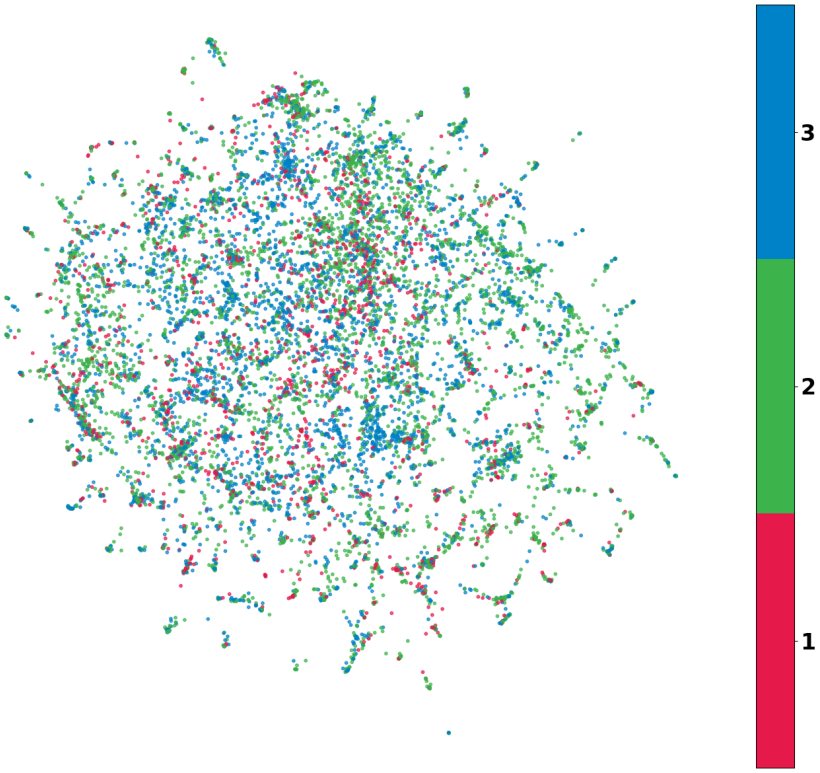


Figura 5.27: Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *Labeled Tweets* a partir de Vetor de Parágrafos 300D.

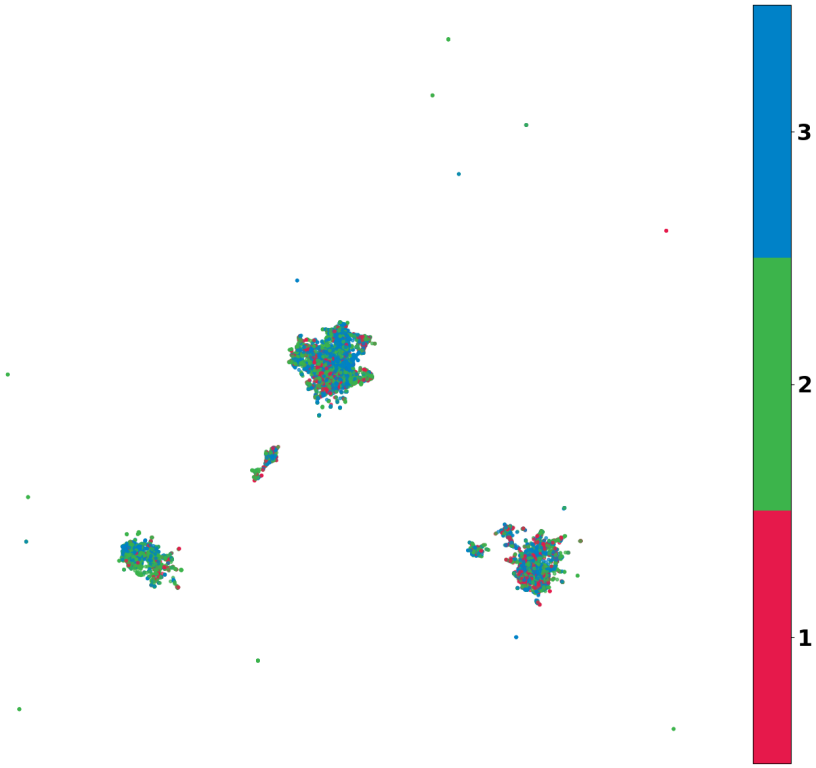
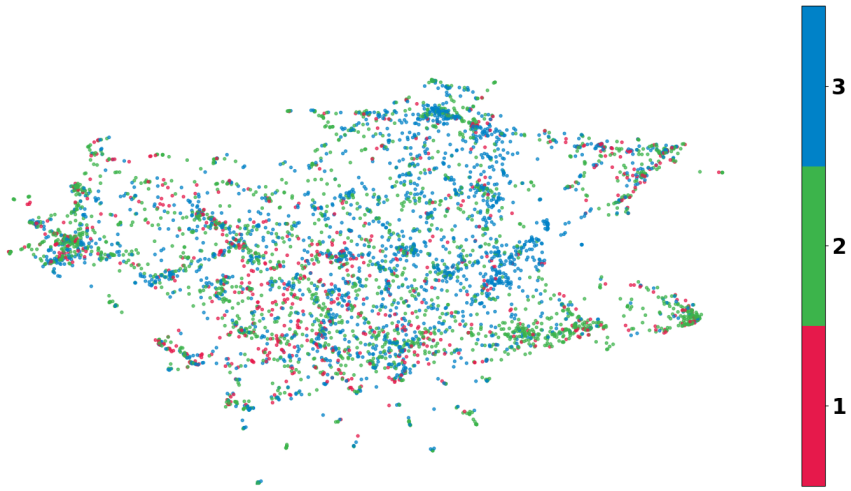
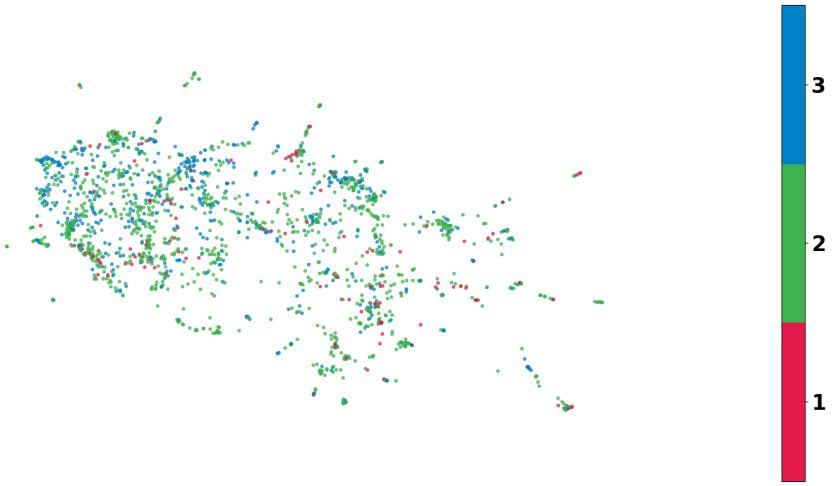


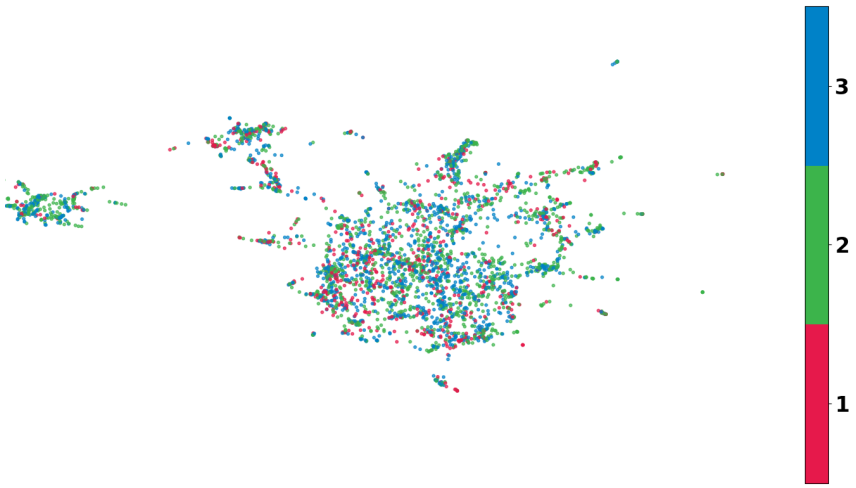
Figura 5.28: Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *Labeled Tweets* a partir de vetor RoBERTa com o modelo *roberta base*.



(a) Região central.



(b) Região esquerda.



(c) Região Direita.

Figura 5.29: Layouts obtidos pela projeção UMAP com *min_dist=0*, para o corpus *Labeled Tweets* a partir de vetor RoBERTa com modelo o *roberta base*.

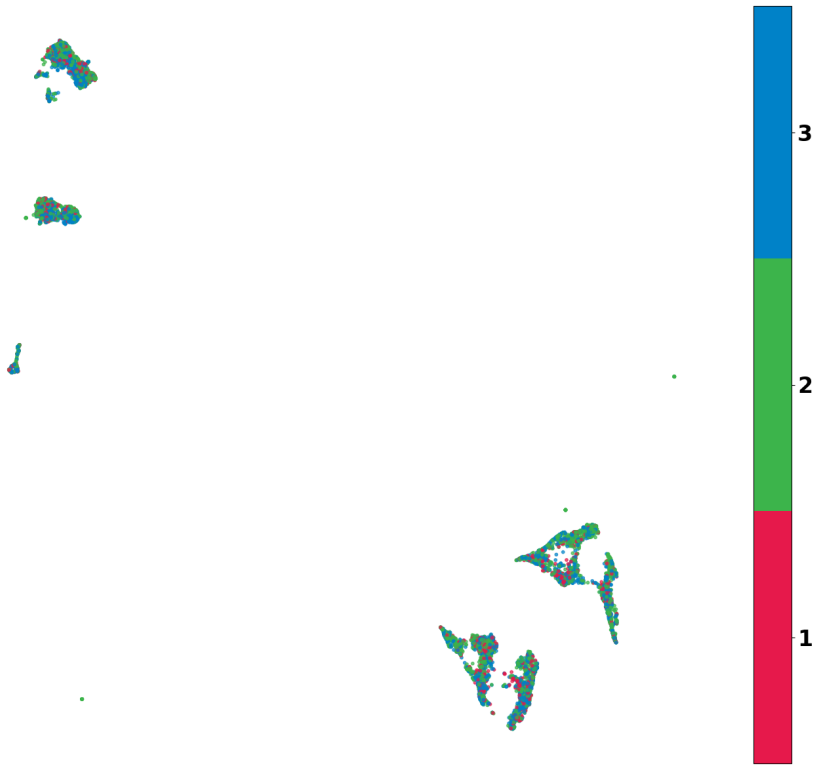
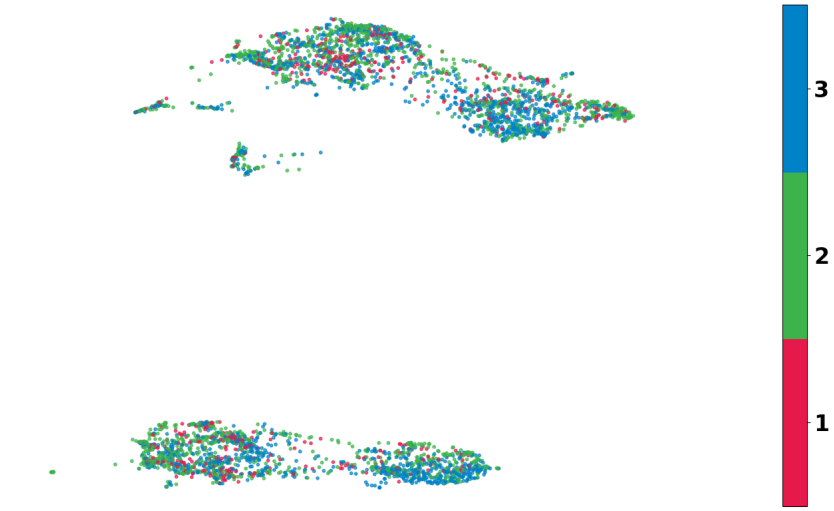
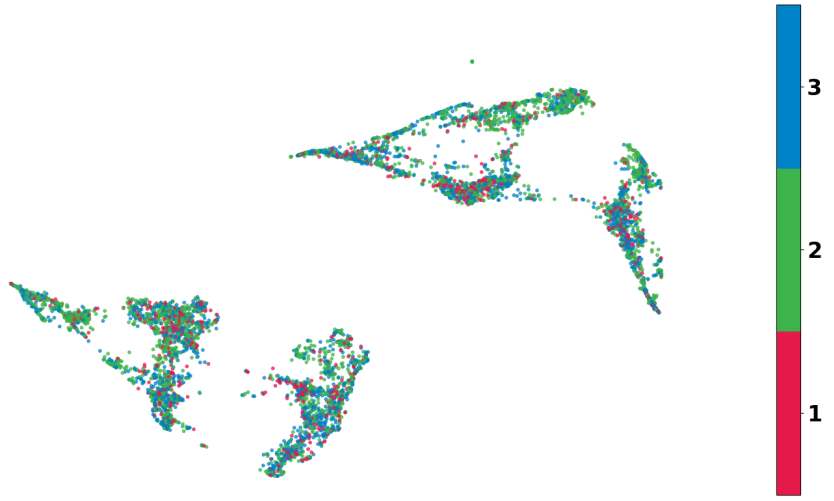


Figura 5.30: Layout obtido pela projeção UMAP com *min_dist=0*, para o corpus *Labeled Tweets* a partir de vetor RoBERTa com o modelo *roberta large*.



(a) Região esquerda.



(b) Região direita.

Figura 5.31: Layouts obtidos pela projeção UMAP com *min_dist=0*, para o corpus *Labeled Tweets* a partir de vetor RoBERTa com o modelo *roberta large*.

5.4 Análise dos resultados

Quando analisamos os resultados das projeções, tanto o UMAP quando o t-SNE apresentam boa preservação de grupos locais, porém nas métricas observadas, o t-SNE apresenta melhores resultados para pequenas vizinhanças. A projeção UMAP, possivelmente, perde precisão quando são observados pequenos grupos por formar regiões mais densas, nas quais é difícil distribuir pontos mantendo as vizinhanças.

A UMAP apresenta maior separação de grandes grupos de pontos, em que as métricas refletem os melhores resultados do UMAP quando utilizado Vetor de Parágrafos. Entretanto, quando a representação RoBERTa é empregada, os grupos de pontos acabam sendo

posicionados distantes entre si, o que prejudica a visualização do layout completo, e exige observar mais de perto grupos de pontos separados para analisar o layout.

O t-SNE apresenta menor separação de grupos grandes, ao analisar os grupos de *tweets*, os gráficos de espalhamento apresentam melhor definição de grupos isolados do que em *News*, principalmente nos vetores gerados pelo BERT. Os grupos do corpus *Tweets*, isolados pelo t-SNE nos gráficos de espalhamento, são pequenos grupos de pontos, que podem representar pequenas variações de temas entre os *tweets*. Já no corpus *News*, cada categoria engloba uma grande quantidade de textos do corpus, e os grandes grupos não são bem isolados nas técnicas de projeção baseadas em t-SNE, ao contrário do que ocorre na projeção UMAP.

Para identificar algumas das características dos vetores de alta dimensionalidade que podem ter influenciado nos resultados projetados, foram extraídas algumas métricas sobre as distâncias euclidianas entre pares de vetores multidimensionais. As métricas são média, desvio padrão, média das 10% menores distâncias e média das 10% maiores distâncias. As métricas foram calculadas a partir dos vetores normalizados para norma 1 e são apresentados na Tabela 5.4.

Corpus de textos / Text Embedding	**Média**	**Desvio Padrão**	**10% menores**	**10% maiores**
News / RoBERTa roberta base	0.007282	0.003550	0.003661	0.015191
News / RoBERTa roberta large	0.024830	0.018048	0.008922	0.064596
News / Vetor de Parágrafos 300D	1.657899	0.118443	1.437567	1.854170
News / Vetor de Parágrafos 1000D	1.659214	0.103105	1.459402	1.821463
Tweets / RoBERTa roberta base	0.007123	0.003332	0.003040	0.013672
Tweets / RoBERTa roberta large	0.036699	0.039549	0.001688	0.121755
Tweets / Vetor de Parágrafos 300D	1.452187	0.151248	1.172357	1.704647
Tweets / Vetor de Parágrafos 1000D	1.452025	0.147837	1.175972	1.695912

Tabela 5.4: Características das distâncias entre vetores de alta dimensionalidade.

Observando os resultados, é possível perceber que a diferença entre as 10% menores e 10% maiores distâncias é maior nos vetores gerados pelo RoBERTa. As menores distâncias ocorrem entre pontos de grupos locais, enquanto as maiores distâncias separam os grandes grupos de pontos. A grande diferença entre as maiores e menores distâncias nos vetores do RoBERTa refletem nos gráficos gerados pelo UMAP, em que existem grupos de pontos com maior separabilidade.

Por utilizar um modelo pré-treinado, técnicas baseadas em BERT são projetadas para se adequar a qualquer texto e ao vocabulário do treinamento. Sem o contexto dos dados ao qual será aplicado, o RoBERTa produz os vetores para cada instância de texto de forma independente, o que pode ser a causa de existir uma grande variação nas distâncias entre vetores, no espaço de alta dimensionalidade. O Vetor de Parágrafos é treinado com o conjunto de textos para o qual os vetores são gerados, por ter conhecimento do conjunto de dados, é possível distribuir melhor os vetores em um espaço único.

Capítulo 6

Conclusão

Nesta monografia, o objetivo foi analisar como as técnicas de visualização baseadas em projeção multidimensional se comportam quando aplicadas vetores de textos. Foram analisadas duas técnicas populares de projeção multidimensional, a t-Stochastic Distributed Neighborhood Embedding (t-SNE) e a Uniform Mapping Approximation Mapping (UMAP), e duas representações de textos, conhecidas como *text embeddings*, o Vetor de Parágrafos e o BERT. Os resultados foram avaliados por métricas quantitativas e análise qualitativa de visualizações baseadas em projeções multidimensionais.

Ao analisar os layouts, foi observado que ambas as técnicas de projeção multidimensional tem resultados similares em agrupar dados similares, apesar da técnica UMAP separar melhor os grupos de pontos. Nos vetores obtidos a partir do *text embedding* BERT, as projeções obtidas geraram grupos mais isolados do que ao utilizar o Vetor de Parágrafos. Quando o *text embedding* BERT é combinada com a técnica de projeção UMAP, foram obtidos grupos de pontos muito distantes nos gráficos de espalhamento, o que pode dificultar a visualização dos dados. Nesse caso, a t-SNE se torna uma opção apropriada, pois gera *layouts* que não possuem uma grande separabilidade de grupos mais distantes.

Com os resultados obtidos neste estudo, foi observado que é possível gerar gráficos de espalhamento que destacam relações de um conjunto de textos, para análise visual de textos. Utilizando o UMAP, observamos gráficos que preservam bem a estrutura global dos vetores de texto, permitindo identificar grupos de dados similares a partir da análise visual. O t-SNE apresenta resultados que, apesar de não isolarem grupos de dados tão bem quanto o UMAP, agrupam textos similares e podem ser utilizados para busca ou recomendação de textos similares a partir do gráfico de espalhamento obtido.

6.1 Trabalhos futuros

Em trabalhos futuros, pode-se realizar uma análise mais detalhada da qualidade dos resultados, utilizando outras métricas de avaliação das projeções como o coeficiente de silhueta [47] [48]. Comparações entre os vetores de alta dimensionalidade também podem ser úteis, para identificar quais características dos textos são bem representadas pelos *text embeddings* e como elas são representadas no espaço de dimensionalidade reduzida. Os resultados obtidos poderiam ser utilizados para aprimorar ou adaptar as técnicas de projeção multidimensional para o contexto de vetores de texto.

As técnicas de *text embeddings* e projeção multidimensional estão em constante evolução, e já existem outras técnicas modernas que poderiam ser analisadas em trabalhos futuros. Técnicas de projeção recentes como o PaCMAP [35] e o TriMap podem ser analisadas para visualização de dados e textos. Assim como diferentes métodos de visualização podem ser utilizados, com análises sobre os métodos de visualização propostos em Shusen et al. [40] ou Kessler [49].

Referências

[1] Mikolov, Tomas, Kai Chen, Greg Corrado e Jeffrey Dean: *Efficient estimation of word representations in vector space*, 2013. viii, 8, 9

[2] Li, Yize, Jiazhong Nie, Yi Zhang, Bingqing Wang, Baoshi Yan e Fuliang Weng: *Contextual recommendation based on text mining*. COLING '10, página 692–700, USA, 2010. Association for Computational Linguistics. 1

[3] Prasad, Anukarsh G., S. Sanjana, Skanda M. Bhat e B. S. Harish: *Sentiment analysis for sarcasm detection on streaming short text data*. Em *2017 2nd International Conference on Knowledge Engineering and Applications (ICKEA)*, páginas 1–5, 2017. 1

[4] Park, Youngki, Sungchan Park, Sang goo Lee e Woosung Jung: *Greedy filtering: A scalable algorithm for k-nearest neighbor graph construction*. Em Bhowmick, Sourav S., Curtis E. Dyreson, Christian S. Jensen, Mong Li Lee, Agus Muliantara e Bernhard Thalheim (editores): *Database Systems for Advanced Applications*, páginas 327–341, Cham, 2014. Springer International Publishing, ISBN 978-3-319-05810-8. 1, 7

[5] Ramos, Juan: *Using tf-idf to determine word relevance in document queries*, 1999. 1, 7

[6] Le, Quoc V. e Tomas Mikolov: *Distributed representations of sentences and documents*, 2014. 2, 9, 10, 20

[7] Devlin, Jacob, Ming Wei Chang, Kenton Lee e Kristina Toutanova: *Bert: Pre-training of deep bidirectional transformers for language understanding*, 2019. 2, 10, 19

[8] Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser e Illia Polosukhin: *Attention is all you need*, 2017. 2, 10

[9] Liu, Shixia, Xiting Wang, Christopher Collins, Wenwen Dou, Fangxin Ouyang, Mennatallah El-Assady, Liu Jiang e Daniel A Keim: *Bridging text visualization and mining: A task-driven survey*. IEEE Transactions on Visualization and Computer Graphics, 25(7):2482–2504, 2018. 2

[10] Card, Stuart: *Information visualization*. CRC press, 2009. 2

[11] Ali, S. M., N. Gupta, G. K. Nayak e R. K. Lenka: *Big data visualization: Tools and challenges*. Em *2016 2nd International Conference on Contemporary Computing and Informatics (IC3I)*, páginas 656–660, 2016. 2

[12] Minghim, Rosane, Fernando Vieira Paulovich e Alneu de Andrade Lopes: *Content-based text mapping using multi-dimensional projections for exploration of document collections*. Em *Visualization and Data Analysis 2006*, volume 6060, página 60600S. International Society for Optics and Photonics, 2006. 2

[13] Tejada, Eduardo, Rosane Minghim e Luis Gustavo Nonato: *On improved projection techniques to support visual exploration of multi-dimensional data sets*. Information Visualization, 2(4):218–231, 2003. 2

[14] Paulovich, Fernando V, Luis G Nonato, Rosane Minghim e Haim Levkowitz: *Least square projection: A fast high-precision multidimensional projection technique and its application to document mapping*. IEEE Transactions on Visualization and Computer Graphics, 14(3):564–575, 2008. 2, 14

[15] Paulovich, Fernando Vieira: *Mapeamento de dados multi-dimensionais – integrando mineração e visualização*. Tese de Doutoramento, Instituto de Ciências Matemáticas e de Computação, University of São Paulo, São Paulo, setembro 2008. 2, 16

[16] Maaten, L.J.P. van der e G.E. Hinton: *Visualizing high-dimensional data using t-sne*. Journal of Machine Learning Research, (9):2579–2605, 2008. 2, 12, 13

[17] McInnes, Leland, John Healy e James Melville: *Umap: Uniform manifold approximation and projection for dimension reduction*, 2020. 2, 13, 16, 21, 24

[18] Lebanon, Guy, Yi Mao e Joshua Dillon: *The locally weighted bag of words framework for document representation*. Journal of Machine Learning Research, 8:2405–2441, outubro 2007. 3

[19] *The 20 newsgroups text dataset*. https://scikit-learn.org/0.19/datasets/twenty_newsgroups.html. 3, 19

[20] Tan, Ah hwee: *Text mining: The state of the art and the challenges*. Em *In Proceedings of the PAKDD 1999 Workshop on Knowledge Disocovery from Advanced Databases*, páginas 65–70, 1999. 5

[21] Sethy, Abhinav e Bhuvana Ramabhadran: *Bag-of-word normalized n-gram models*. páginas 1594–1597, janeiro 2008. 8

[22] Jurafsky, Daniel e James H. Martin: *Speech and language processing : an introduction to natural language processing, computational linguistics, and speech recognition*. Prentice Hall., Upper Saddle River, N.J., 2000, ISBN 978-0-13-095069-7. 8

[23] Wu, Yonghui, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes e Jeffrey Dean: *Google's neural machine translation system: Bridging the gap between human and machine translation*, 2016. 10, 19

[24] Liu, Yinhan, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer e Veselin Stoyanov: *Roberta: A robustly optimized bert pretraining approach*, 2019. 10, 20

[25] Tejada, Eduardo, Rosane Minghim e Luis Nonato: *On improved projection techniques to support visual exploration of multi-dimensional data sets*. Information Visualization, 2:218–231, dezembro 2003. 11

[26] Linderman, G.C., Rachh M. Hoskins J.G. et al.: *Fast interpolation-based t-sne for improved visualization of single-cell rna-seq data*. Nat Methods, (16):243–245, 2019. 13, 24

[27] Maaten, L.J.P. van der: *Accelerating t-sne using tree-based algorithms*. Journal of Machine Learning Research, (15):3221–3245, 2014. 13

[28] Bernhardsson, Erik: *Annoy approximate nearest neighbors oh yeah*. https://github.com/spotify/annoy, 2018. 13

[29] May, J Peter: *Simplicial objects in algebraic topology*, volume 11. University of Chicago Press, 1992. 13

[30] Lane, Saunders Mac: *Categories for the working mathematician*, volume 5. Springer Science Business Media, 2013. 13

[31] Oskolkov, Nikolay: *How exactly umap works*. https://towardsdatascience.com/how-exactly-umap-works-13e3040e1668, 2019. 13

[32] Dong, Wei, Charikar Moses e Kai Li: *Efficient k-nearest neighbor graph construction for generic similarity measures*. Em *Proceedings of the 20th International Conference on World Wide Web*, WWW '11, página 577–586, New York, NY, USA, 2011. Association for Computing Machinery, ISBN 9781450306324. `https://doi.org/10.1145/1963405.1963487`. 13

[33] Jolliffe, Ian: *Principal Component Analysis*, páginas 1094–1096. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, ISBN 978-3-642-04898-2. `https://doi.org/10.1007/978-3-642-04898-2_455`. 16

[34] Faloutsos, Christos e King Ip Lin: *Fastmap: A fast algorithm for indexing, data-mining and visualization of traditional and multimedia datasets*. Em *Proceedings of the 1995 ACM SIGMOD International Conference on Management of Data*, SIGMOD '95, página 163–174, New York, NY, USA, 1995. Association for Computing Machinery, ISBN 0897917316. `https://doi.org/10.1145/223784.223812`. 16

[35] Wang, Yingfan, Haiyang Huang, Cynthia Rudin e Yaron Shaposhnik: *Understanding how dimension reduction tools work: An empirical approach to deciphering t-sne, umap, trimap, and pacmap for data visualization*, 2020. 16, 58

[36] Amid, Ehsan e Manfred K. Warmuth: *Trimap: Large-scale dimensionality reduction using triplets*, 2019. 16

[37] Tang, Jian, Jingzhou Liu, Ming Zhang e Qiaozhu Mei: *Visualizing large-scale and high-dimensional data*. Proceedings of the 25th International Conference on World Wide Web, Apr 2016. http://dx.doi.org/10.1145/2872427.2883041. 16

[38] Belkin, Mikhail e Partha Niyogi: *Laplacian eigenmaps for dimensionality reduction and data representation*. Neural Computation, 15(6):1373–1396, 2003. 16

[39] Tenenbaum, Joshua B., Vin de Silva e John C. Langford: *A global geometric framework for nonlinear dimensionality reduction*. Science, 290(5500):2319–2323, 2000, ISSN 0036-8075. https://science.sciencemag.org/content/290/5500/2319. 16

[40] Shusen Liu, Peer-Timo Bremer, Jayaraman J. Thiagarajan Vivek Srikumar Bei Wang Yarden Livnat e Valerio Pascucci: *Visual exploration of semantic relationships in neural word embeddings*. IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS, 24(1):553–562, janeiro 2018. 17, 58

[41] Michailidis, Marios: *Sentiment140 dataset with 1.6 million tweets*. https://www.kaggle.com/kazanova/sentiment140, 2017. 19

[42] Borges, Vinícius Ruela Pereira: *natural_language_processing*. https://github.com/viniciusrpb/natural_language_processing, 2021. 19

[43] Loper, Edward e Steven Bird: *Nltk: The natural language toolkit*. Em *Proceedings of the ACL-02 Workshop on Effective Tools and Methodologies for Teaching Natural Language Processing and Computational Linguistics - Volume 1*, ETMTNLP '02, página 63–70, USA, 2002. Association for Computational Linguistics. https://doi.org/10.3115/1118108.1118117. 19

[44] Porter, Martin F.: *Snowball: A language for stemming algorithms*. Published online, October 2001. http://snowball.tartarus.org/texts/introduction.html, Accessed 11.03.2008, 15.00h. 19

[45] *Doc2vec paragraph embeddings*. https://radimrehurek.com/gensim/models/doc2vec.html. 23

[46] Ott, Myle, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier e Michael Auli: *fairseq: A fast, extensible toolkit for sequence modeling*. Em *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019. 23

[47] Tan, P.N., M. Steinbach e V. Kumar: *Introduction to Data Mining*. Always learning. Pearson Addison Wesley, 2006, ISBN 9780321321367. 58

[48] Eler, Danilo Medeiros, Jaqueline Batista Martins Teixeira, Priscila Alves Macanha e Rogério Eduardo Garcia: *Simplified stress and simplified silhouette coefficient to a faster quality evaluation of multidimensional projection techniques and feature spaces*. Em *19th International Conference on Information Visualisation*, páginas 133–139. IEEE, 2015. 58

[49] Kessler, Jason: *Scattertext: a browser-based tool for visualizing how corpora differ*. Em *Proceedings of ACL 2017, System Demonstrations*, páginas 85–90, Vancouver, Canada, julho 2017. Association for Computational Linguistics. `https://www.aclweb.org/anthology/P17-4015`. 58