Universidade de São Paulo
Instituto de Matemática e Estatística
Bacharelado em Ciência da Computação

Modelagem de Tópicos em Textos Históricos utilizando LLMs

João Pedro Lukasavicus Silva

Monografia Final

mac 499 — Trabalho de Formatura Supervisionado

Supervisor: Mateus Espadoto

São Paulo

Esta seção é opcional e fica numa página separada;
ela pode ser usada para uma dedicatória ou epígrafe.

Agradecimentos

Do. Or do not. There is no try.
— Mestre Yoda

Texto texto. Texto opcional.

Resumo

João Pedro Lukasavicus Silva. **Modelagem de Tópicos em Textos Históricos utilizando LLMs**. Monografia (Bacharelado). Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo.

Elemento obrigatório, constituído de uma sequência de frases concisas e objetivas, em forma de texto. Deve apresentar os objetivos, métodos empregados, resultados e conclusões. O resumo deve ser redigido em parágrafo único, conter no máximo 500 palavras e ser seguido dos termos representativos do conteúdo do trabalho (palavras-chave). Deve ser precedido da referência do documento. Texto texto. Texto texto.

Palavras-chave: Palavra-chave1. Palavra-chave2. Palavra-chave3.

Abstract

João Pedro Lukasavicus Silva. **Title of the document: *a subtitle***. Capstone Project Report (Bachelor). Institute of Mathematics and Statistics, University of São Paulo, São Paulo.

Elemento obrigatório, elaborado com as mesmas características do resumo em língua portuguesa. De acordo com o Regimento da Pós-Graduação da USP (Artigo 99), deve ser redigido em inglês para fins de divulgação. É uma boa ideia usar o sítio www.grammarly.com na preparação de textos em inglês. Text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text. Text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text text.

Keywords: Keyword1. Keyword2. Keyword3.

Lista de abreviaturas

CFT	Transformada contínua de Fourier (*Continuous Fourier Transform*)
DFT	Transformada discreta de Fourier (*Discrete Fourier Transform*)
EIIP	Potencial de interação elétron-íon (*Electron-Ion Interaction Potentials*)
STFT	Transformada de Fourier de tempo reduzido (*Short-Time Fourier Transform*)
ABNT	Associação Brasileira de Normas Técnicas
URL	Localizador Uniforme de Recursos (*Uniform Resource Locator*)
IME	Instituto de Matemática e Estatística
USP	Universidade de São Paulo

Lista de símbolos

ω	Frequência angular
ψ	Função de análise *wavelet*
Ψ	Transformada de Fourier de ψ

Lista de figuras

Lista de tabelas

Lista de programas

Sumário

Introdução

Capítulo 1

Background

Capítulo 2

Experimentos

2.1 Conjunto de dados

2.1.1 Textos históricos

O conjunto de dados utilizados neste trabalho consiste na primeira versão completa traduzida do Latim para a língua Inglesa de *Etimologias* de *Isidoro de Sevilha*.[1] O texto está disponível em versão digital, em PDF, como um volume único. O texto foi extraído do arquivo PDF do livro e armazenado em formato texto puro, removendo-se cabeçalhos, rodapés, números de página, e outros artefatos dessa conversão.

2.1.2 Preparação dos dados

(TODO: Adicionar figura do esquema geral do processo)

Pré-processamento

Primeiramente, pelo modo como cada livro está formatado, fizemos um processamento inicial para separar cada livro em capítulos, e juntar palavras com hífen. TODO: exemplos. Então, cada capítulo foi subdividido em sentenças usando o módulo SentenceRecognizer da biblioteca Spacy.[2] TODO: exemplos.

Uma particularidade da edição escolhida da obra *Etimologias* é que cada capítulo é subdividido em seções relativamente curtas, a maioria tendo somente uma oração. Com isso, podemos fazer a modelagem dos tópicos usando três níveis diferentes de granularidade, a depender de como definimos um "documento", unidade básica de nossa análise:

- **Seções**, já presentes no texto;
- **Sentenças**, delimitadas automaticamente;

[1] Disponível em: https://www.cambridge.org/core/books/etymologies-of-isidore-of-seville/F2336BA779D4ED95E6D25AAE2CCBAD25

[2] https://spacy.io/api/sentencerecognizer

- **Misto**, com ambas seções e sentenças;

Em nossa análise, utilizamos todas essas abordagens e comparamos os resultados.

TODO: Exemplos das seções.

Stop words, stemming e lematização

Tarefas rotineiras de pré-processamento de linguagem natural incluem: remoção de *stop words*, que são palavras muito comuns e irrelevantes, *stemming*, e lematização, que são formas de padronizar palavras, reduzindo-as a sua forma mais básica. Apesar de outras técnicas de modelagem de tópicos adotarem essas tarefas como parte do seu processo de preparação dos dados, não utilizamos essas técnicas para este trabalho, pois os modelos de *embedding* que usamos utilizam informações contextuais de cada palavra em uma sentença, e remover palavras ou modificá-las poderia prejudicar a performance de tais modelos (Chaerul Haviana *et al.*, 2023).

Geração de *embeddings*

Depois da etapa de pré-processamento dos textos, geramos *embeddings* para cada documento, utilizando modelos do tipo *sentence embedders* pré-treinados. Os modelos usados para a geração desses *embeddings* foram: *sentence-transformers/LaBSE*, *jinaai/jina-embeddings-v3*, *intfloat/multilingual-e5-large-instruct*, *nomic-ai/nomic-embed-text-v2-moe*. Estes modelos foram desenhados e treinados para que sentenças semanticamente similares em línguas diferentes, ou traduções, estejam próximas umas das outras em um espaço latente. Podemos usar estes modelos para comparar textos em diferentes idiomas e analisar suas conexões.

Redução de dimensionalidade

A seguir, fizemos uma redução de dimensionalidade de cada um dos conjuntos de embeddings gerados. Para isso, utilizamos o UMAP, com os seguintes parâmetros:

Tabela 2.1: *Parâmetros usados para o UMAP*

Nome do parâmetro	Descrição	Valor usado	Nota
`n_neigbors`	Controla como o UMAP equilibra estrutura local versus global nos dados. Valores maiores levam a uma visão mais global da estrutura dos dados, e valores menores a uma visão mais local.	`10`	O valor padrão para o BERTopic é 15.
`n_components`	Dimensionalidade do *dataset* resultante.	`5`	Valor padrão para o BERTopic.

Continua na próxima página

Tabela 2.1 – continuação

Nome do parâmetro	Descrição	Valor usado	Nota
`min_dist`	Controla o quão dispersos os pontos estarão na projeção de baixa dimensionalidade. Valores baixos podem ser interessantes para tarefas de clustering.	`0`	Valor padrão para o BERTopic.
`low_memory`	Restringe o uso de memória em detrimento da velocidade da computação. Útil quando há pouca memória disponível.	`false`	
`metric`	Métrica usada para calcular distâncias entre pontos.	`"cosine"`	Valor padrão para o BERTopic.
`random_state`	Usado para garantir determinismo.	`42`	

TODO: normalmente o BERTopic usa `n_neighbors = 15`, então por que eu mudei esse valor? A implementação usada foi a da biblioteca umap-learn.[3]

A etapa de redução de dimensionalidade é importante para o desempenho da etapa posterior do método, de clusterização, por conta de um fenômeno conhecido como a Maldição da Dimensionalidade: em espaços de alta dimensionalidade, os conceitos de proximidade, distância, ou vizinhos mais próximos perdem sua significância (Radovanović *et al.*, 2010). Este fenômeno então acaba por prejudicar a performance de algoritmos e técnicas que utilizem esses conceitos, como algoritmos de clusterização populares (Radovanović *et al.*, 2010; Aggarwal *et al.*, 2001).
Além disso, nessa etapa também computamos reduções desses embeddings a 2 dimensões, para serem utilizadas posteriormente em visualizações, também utilizando o UMAP. Fora o número de dimensões do espaço resultante, todos os outros parâmetros permanecem os mesmos.

Conjuntos de dados

Após essas etapas iniciais, definimos nossos conjuntos de dados. Cada conjunto é definido por um modelo de embedding e um nível de granularidade, já explicado. TODO: exemplos, visualizações dos espaços de embeddings.

2.1.3 Persistência

Os embeddings gerados, suas reduções, e diversos metadados foram armazenados em um banco de dados do *ChromaDB*, juntamente com cada documento.

TODO: versão final do modelo de dados no ChromaDB.

[3] https://umap-learn.readthedocs.io/en/latest/

Dessa forma, podemos fazer diversos tipos de busca, como busca por similaridade, busca textual, filtrar resultados baseados em metadados de cada documento, como autor, idioma, etc.

2.2 Experimentos

Cada experimento consistiu em aplicar o BERTopic, utilizando os diferentes conjuntos de dados mencionados anteriormente, variando o número de tópicos a serem descobertos: 10, 20, 50 e 100 tópicos.

Com os embeddings gerados e suas reduções já computadas, inicializamos o BERTopic de modo a pular essas etapas iniciais e começar pela etapa de clustering. O algoritmo utilizado foi o HDBSCAN, disponível na biblioteca Scikit-learn,[4] e utilizamos os seguintes parâmetros:

Tabela 2.2: *Parâmetros usados para o HDBSCAN*

Nome do parâmetro	Descrição	Valor usado	Nota
`min_cluster_size`	Controla o tamanho mínimo e a quantidade de *clusters* gerados.	`10`	Valor padrão para o BERTopic.
`min_samples`	Controla a quantidade de pontos classificados como ruído.	`1`	Escolhemos um valor baixo para tentar diminuir a quantidade pontos classificados como outliers.

Para melhorar a representação dos tópicos gerados, fizemos alguns ajustes nas etapas finais do método:

- Remoção de stop-words, no CountVectorizer
- Remoção de palavras comuns e uso de BM-25 weighting measure.

[4] https://scikit-learn.org/stable/modules/generated/sklearn.cluster.HDBSCAN.html

Capítulo 3

Resultados

Capítulo 4

Conclusão

Referências

[Aggarwal *et al.* 2001] Charu C. Aggarwal, Alexander Hinneburg e Daniel A. Keim. "On the surprising behavior of distance metrics in high dimensional spaces". In: *Proceedings of the 8th International Conference on Database Theory*. ICDT '01. Berlin, Heidelberg: Springer-Verlag, 2001, pp. 420–434. isbn: 3540414568 (citado na pg. 7).

[Chaerul Haviana *et al.* 2023] Sam Farisa Chaerul Haviana, Sri Mulyono e Badie'Ah. "The effects of stopwords, stemming, and lemmatization on pre-trained language models for text classification: a technical study". In: *2023 10th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI)*. 2023, pp. 521–527. doi: 10.1109/EECSI59885.2023.10295797 (citado na pg. 6).

[Radovanović *et al.* 2010] Miloš Radovanović, Alexandros Nanopoulos e Mirjana Ivanović. "Hubs in space: popular nearest neighbors in high-dimensional data". *J. Mach. Learn. Res.* 11 (dez. de 2010), pp. 2487–2531. issn: 1532-4435 (citado na pg. 7).

Índice remissivo