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

O conteúdo deste trabalho é publicado sob a licença CC BY 4.0
(Creative Commons Attribution 4.0 International License)

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 nas obras *Etimologias - Isidoro de Sevilha*, ... TODO: lista de obras. Pergunta: Vamos usar apenas *Etimologias* ou usaremos outras obras? Vamos usar outras edicoes? Outros idiomas?

Todos os arquivos estão em formato texto puro, e foram obtidos em... TODO: Como se pode obter esses arquivos?

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. PERGUNTA: Eu deveria explicar como esse módulo funciona? TODO: exemplos.

Uma particularidade da edição escolhida da obra *Etimologias* é que cada capítulo possui uma subdivisão em *subcapítulos* PERGUNTA: Qual o melhor nome para essa subdivisão? TODO: exemplos. Com isso, podemos fazer a modelagem dos tópicos usando três níveis diferente de granularidade, a depender de como definimos um "documento", unidade básica de nossa análise:

- **Subcapítulos**, já presentes no texto;
- **Sentenças**, delimitadas pelo módulo SentenceRecognizer, do Spacy;
- **Misto**, com ambos subcapítulos e sentenças;

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

Apesar de algumas tarefas rotineiras em processamento de linguagem natural serem usadas em outras abordagens de modelagem de tópicos, como remoção de stop-words, stemming, lematização, etc, não fizemos esses procedimentos na fase de preparação dos dados para a abordagem utilizada neste 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. PERGUNTA: Devo explicar remoção de stop words, stemming e lematização?

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 parâmetros `n_neighbors = 10`, `n_components = 5`, `min_dist = 0.0`, `low_memory = false`, `metric = "cosine"`, parâmetros padrão usados no BERTopic, e `random_state = 42`, para prevenir comportamento estocástico. TODO: normalmente o BERTopic usa `n_neighbors = 15`, então por que eu mudei esse valor? A implementação usada foi a da biblioteca umap-learn.

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 dimensão, os conceitos de proximidade, distância, ou vizinhos mais próximos perdem sua significância. Este fenômeno então acaba por prejudicar a performance de algoritmos e técnicas que utilizem esses conceitos, como praticamente todos os algoritmos de clusterização conhecidos.
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.

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, utilizando `min_cluster_size = 10`, que é o valor padrão para este parâmetro, e `min_samples = 1`, para diminuir a quantidade de pontos classificados como outliers. TODO: Por que utilizamos o HDBSCAN?

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.

Capítulo 3

Resultados

Capítulo 4

Conclusão

Índice remissivo