Escola de Letras, Artes
e Ciências Humanas
elach.uminho.pt

Renato Lopes Neto PG51178

Análise e visualização de dados

Mestrado em Humanidades digitais

Junho de 2023

RESUMO

Com o aparecimento da tecnologia e a crescente produção e disponibilidade de dados, com o tempo passou a ser muito importante a exploração de métodos e técnicas para extração, seleção e diferenciação de informação importante a partir destes volumes enormes de dados. Ainda assim durante muito tempo, os textos escritos foram negligenciados nesse aspeto, visto que a maioria das técnicas e métodos de análise dos dados estavam voltados para dados estruturados e quantitativos, deixando de lado a riqueza e particularidade das informações qualitativas presentas na linguagem.

Através da extração de entidades nomeadas, identificaremos nomes de pessoas, lugares e organizações, destacando personagens principais, sua coocorrência e categorização.
Faremos a extração de lemas e análise de frequência, permitindo a compreensão das palavras mais relevantes e sua distribuição em diferentes categorias gramaticais.
Ao considerar a dimensão temporal, analisaremos a distribuição de datas ao longo dos capítulos ou do livro. Isso nos permitirá compreender a evolução temporal dos eventos narrados.
Além disso, realizaremos análises de sentimentos, identificando palavras positivas, negativas e neutras. Essas análises proporcionarão uma compreensão mais profunda da carga emocional presente no texto.

Os resultados obtidos fornecerão insights significativos, contribuindo para uma melhor compreensão do conteúdo textual e possibilitando a tomada de decisões informadas.

Palavras-chave: tecnologia, extração, análise, frequência, sentimento, dados, qualitativos, quantitativos, categorização

Índice

1.INTRODUÇÃO

À medida que a tecnologia avança e os dados se tornam cada vez mais acessíveis, tornou-se fundamental explorar métodos e técnicas para extrair e analisar informações textuais. No entanto, por muito tempo, os textos escritos foram subestimados em comparação com os dados estruturados e quantitativos, apesar de serem fontes ricas e únicas de informações qualitativas.

1.1 Problema

O problema reside na falta de atenção dada aos textos escritos como fonte de informações valiosas. As técnicas e métodos tradicionais de análise de dados não levam em consideração as particularidades da linguagem e falham em aproveitar ao máximo os dados textuais disponíveis.

1.2 Enquadramento

Diante desse contexto, surge a necessidade de desenvolver abordagens eficazes para a extração de entidades nomeadas, análise de frequência, consideração da dimensão temporal, análise de sentimentos, bem como a visualização e interpretação dos dados obtidos nos textos escritos. Essas técnicas permitirão uma compreensão mais profunda do conteúdo textual e das informações relevantes presentes.

1.2 Objetivos

Pretendemos com esse projeto desenvolver métodos eficientes para a extração de entidades nomeadas em textos escritos, permitindo a identificação precisa de pessoas, lugares e organizações relevantes presentes nos documentos analisados. Isso contribuirá para uma melhor compreensão das relações e interações entre essas entidades, bem como a sua importância para o contexto do texto, também queremos fazer análises de frequência, explorando a distribuição das palavras ao longo dos textos bem como a identificação de padrões e tendências significativas. Intender a importância relativa das palavras e sua distribuição em diferentes categorias gramaticais e como estas que estas irão fornecer perceções sobre a estrutura e o estilo linguístico presentes nos textos, também temos como objetivo investigar a dimensão temporal dos eventos no texto, que envolve a análise da distribuição de datas ao longo do conteúdo, permitindo assim identificar a análise evolução temporal dos eventos e proporcionando uma visão cronológica dos acontecimentos narrados.

Outro objetivo principal é fazer análises dos sentimentos, com o intuito de identificar o sentimento global do texto, palavras entre outros presentes nos textos. Essa análise emocional irá nos ajudar a compreender a carga emocional do conteúdo, oferecendo assim uma perspetiva sobre o tom e a atitude do autor em causa.

E por fim, buscamos desenvolver propostas de visualização dos dados obtidos a partir das análises feitas. Através de gráficos diagramas e outras representações visuais, pretendendo assim facilitar a interpretação e análise dos resultados, proporcionado uma visão clara perspicaz das informações textuais extraídas.

No conjunto, esses objetivos irão contribuir para uma melhor compreensão do conteúdo textual, irão fornecer perceções que possibilitarão uma visualização eficiente e interpretativa dos dados textuais analisados, com um grande foco principalmente na automatização e criação de uma experiência “user-friendly”, para que a ferramenta final possa ser disponibilizada e usada em outros contextos por outras pessoas.

2.ABORDAGEM METODOLÓGICA

2.1 Extração dos dados

Para a extração dos dados e sua consequente utilização foi desenvolvido um código que começa importando as bibliotecas necessárias, como **re** para expressões regulares, **string** para operações com **strings**, pandas para manipulação de dados, **streamlit** para criar uma interface web, **spacy** para processamento de linguagem natural, **nltk** para tarefas de PLN, **altair** para visualização de dados e "requests" para fazer requisições HTTP.

Em seguida, o código carrega um modelo pré-treinado do **Spacy** para processar texto em português.

Cleaning

Duas listas são criadas, **excluded_people** e **excluded_places**, para especificar as entidades que devem ser excluídas da análise.
A função **clean_text** é definida para limpar o texto fornecido. Ela remove palavras e caracteres específicos, espaços em excesso, "tokeniza" o texto em palavras, remove "stopwords" (palavras comuns como "e", "o", etc.) e remove sinais de pontuação. O texto limpo é retornado como saída.
A função **extract_entities** é definida para extrair entidades do texto limpo. Ela recebe o texto limpo e um tipo de entidade como entrada. A função usa o modelo do **Spacy** para extrair entidades do tipo especificado (por exemplo, pessoa ou localização) do texto. A função filtra as entidades excluídas e conta a frequência de cada entidade. Ela retorna um dicionário contendo os nomes das entidades como chaves e suas frequências como valores.

2.1.2 O Método de visualização das entidades e dados.

O código cria uma interface **Streamlit** com o título "Análise de Texto". O usuário pode selecionar um arquivo de texto por meio de um menu suspenso. Os nomes dos arquivos disponíveis são fixos no código que, porém, podem ser alterados para outro caso desejado.

Um "DataFrame" do **pandas** é criado com base nas entidades de pessoa extraídas e suas frequências, as entidades de pessoa são filtradas para incluir apenas aquelas com frequências maiores que 2.

O "DataFrame" filtrado é ordenado em ordem decrescente de frequência depois é criado um gráfico de barras usando a biblioteca **altair** para visualizar as frequências das entidades de pessoa e igualmente outro "DataFrame" é criado para as entidades de localização e suas frequências, as entidades de localização são filtradas para incluir apenas aquelas com frequências maiores que 1.
O "DataFrame" filtrado é ordenado em ordem decrescente de frequência e é criado um gráfico de barras para visualizar as frequências das entidades de localização.

Os gráficos são exibidoss na interface **Streamlit**, mostrando as frequências das entidades de pessoa e localização.

De uma forma resumida, o código lê um arquivo de texto selecionado, limpa o texto, extrai entidades de pessoa e localização usando um modelo pré-treinado do **Spacy**, filtra e ordena as entidades extraídas com base em suas frequências e visualiza as frequências usando gráficos de barras em uma interface **Streamlit**.

2.2 A extração e análise de sentimento dos textos

Para a parte de análise de sentimento depois de um refinamento coletivo na sala de aula com os professores e colegas, o código em Python fornece um conjunto de funções e algoritmos que nos permitem analisar o sentimento de um texto com base em um léxicon de palavras, ficheiro esse denominado de "sentilexjj.txt". Esse arquivo contém palavras associadas a um valor de polaridade, indicando se a palavra possui um sentimento positivo ou negativo. O código lê esse arquivo e armazena as palavras e seus respetivos valores de polaridade em um dicionário.

Em seguida, o texto de entrada é processado, O código lê o texto a partir de um arquivo fornecido como argumento de linha de comando e o armazena em uma variável chamada "txt". Esse texto será analisado para determinar o sentimento associado a cada parte separada, nesse caso os capítulos.

A função "sentimento(frase)" é responsável por realizar a análise de sentimento de uma frase. Primeiro, a frase é dividida em uma lista de palavras utilizando expressões regulares. Em seguida, para cada palavra na lista, verifica-se se ela está presente no léxicon de palavras (POL). Se a palavra estiver no léxicon, seu valor de polaridade é verificado. Se o valor for positivo, é incrementado o contador de sentimentos positivos (ptotalpos) e o contador de palavras positivas (qp). Se o valor for negativo, é incrementado o contador de sentimentos negativos (ptotalneg) e o contador de palavras negativas (qn).

Após a análise de sentimento de cada parte separada do texto, os resultados são armazenados em um arquivo de saída no formato CSV. O código cria um arquivo com o mesmo nome do arquivo de entrada, acrescentando a extensão ".csv". O arquivo de saída contém informações como o número de caracteres, o total de palavras positivas e negativas, além do fator de equilíbrio (Factor), que é a razão entre os sentimentos positivo e negativo.

Além disso, o código gera um gráfico de barras que ilustra o sentimento de cada parte separada do texto. A função "criagraf(xl, y)" é responsável por criar esse gráfico utilizando a biblioteca Matplotlib. O gráfico é exibido na tela e também salvo como um arquivo de imagem de nome "g.png".

2.2.1 O método de visualização da análise de sentimentos

Após a realização da análise de sentimento com base no código mencionado anteriormente, foi desenvolvido um aplicativo de visualização de dados para explorar os resultados de forma interativa. O aplicativo foi implementado utilizando a biblioteca Streamlit e a visualização dos dados é feita com o auxílio da biblioteca Altair.

O aplicativo começa com a opção de carregar um arquivo CSV contendo os resultados da análise de sentimento. O usuário pode fazer o upload do arquivo por meio de uma interface simples e intuitiva. O arquivo é lido utilizando a biblioteca Pandas e os dados são armazenados em um DataFrame. Após o carregamento do arquivo, os dados são exibidos na seção correspondente do aplicativo. O usuário pode analisar o DataFrame e verificar os resultados da análise de sentimento para cada parte separada do texto. Em seguida, o aplicativo permite que o usuário selecione as colunas para os eixos X e Y dos gráficos interativos. Ele pode escolher as colunas relevantes no DataFrame, o que proporciona flexibilidade na visualização dos dados.

	Cap	Nº carateres	totpos	quanpos	totneg	quanneg	palavras	rationeg
0	0	189	0	0	0	0	25	1.0777
1	1	4,800	26	27	31.5	32	774	1.0777
2	2	9,245	72	72	44.5	49	1,571	1.0777
3	3	3,557	27	28	13	13	612	1.0777
4	4	11,526	59	62	58.5	63	1,929	1.0777
5	5	9,442	54.5	58	50.5	55	1,648	1.0777
6	6	7,716	45	47	44.5	47	1,281	1.0777
7	7	21,969	129.5	143	98.5	108	3,944	1.0777

Figura 1: Interface da app de visualização

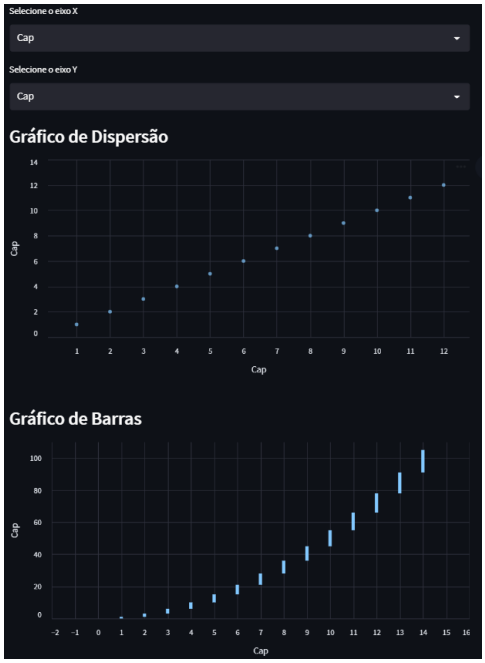


Figura 2: Gráficos disponíveis

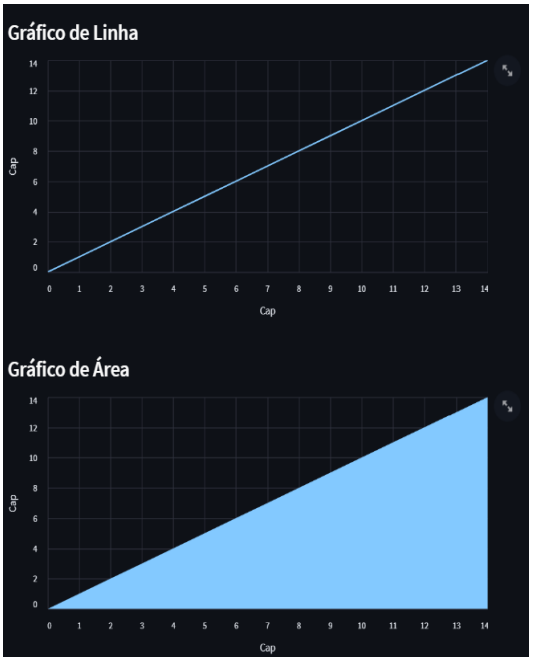


Figura 3: Gráficos disponíveis

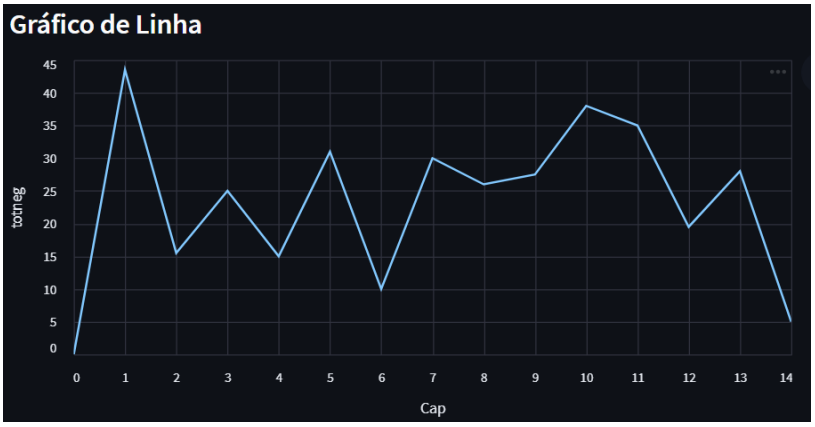
2.2.2 Possíveis perguntas e interpretações?

Usando a nossa aplicação que perguntas podemos fazer para tentar interpretar os nossos textos e que conclusões podemos tirar com essas visualizações?

Optei por gráfico simples e diretos que usam principalmente o princípio do desenvolvimento do da obra, neste caso capítulo por capítulo entre outros.

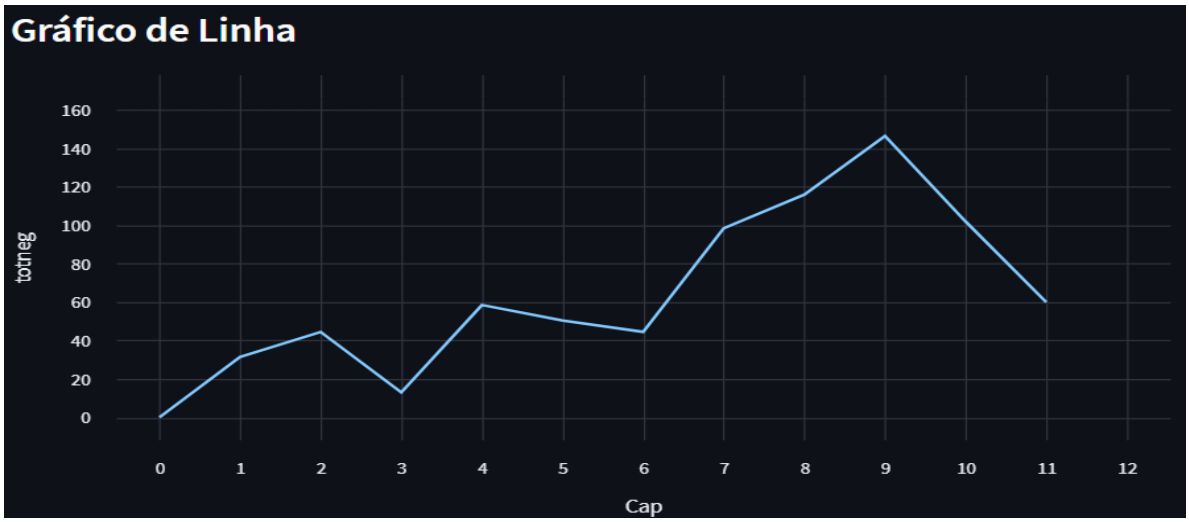
Pergunta:

Existe alguma relação entre o avançar dos **Cap** e o **totneg?**

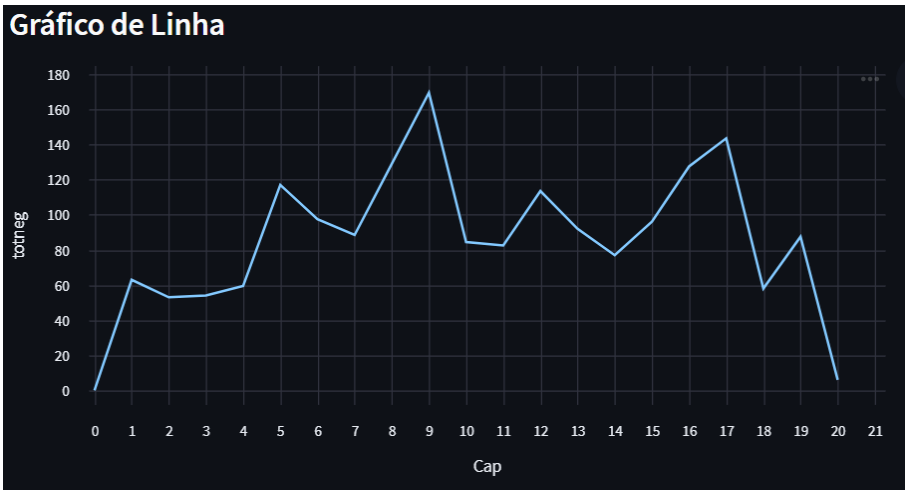


Para a obra: **A_Gratidao** observou-se o seguinte: A maior quantidade observa-se nos primeiros capítulos equilibrando nos capítulos de meio e tendo uma queda nos últimos capítulos, será que nas outras obras isso é verdade?

Vejamos a obra **A_Infanta_Capelista,** como podemos observar os dois gráficos parece bem distintos um do outro



Agora ao analisar a obra **Carlota_Angela,** nesse caso podemos encontrar algum padrão em comum com a primeira obra, um padrão de estabilidade nos capítulos do meio.



A conclusão que podemos tirar é que ao ver assim, essa informação em nada nos acresce para tirar conclusões sobre o autor ou essas três obras que foram analisadas. Se a quantidade da amostra fosse maior talvez poderíamos identificar alguns padrões e assim tirar conclusões.

3.LIMITAÇÕES E PERSPETIVAS DE DESENVOLVIMENTOS FUTUROS

Ao longo do desenvolvimento do projeto deparei-me com muitos problemas e erros e limitações da minha própria máquina e scripts desenvolvidos. No final o resultado apresentado possui algumas limitações e precisa e também oferece espaço para futuros desenvolvimentos e aprimoramentos.

3.1 Limitações

Dependência do modelo pré-treinado: O código carrega um modelo pré-treinado do Spacy para realizar a extração de entidades. Essa dependência implica que as entidades são identificadas apenas com base no conhecimento e nas capacidades do modelo em uso. Portanto, se o modelo não estiver atualizado ou não possuir um bom desempenho para um determinado domínio, as entidades podem não ser extraídas com precisão.

Exclusão manual de entidades: O código possui uma lista de entidades pré-definidas que devem ser excluídas da análise, como nomes de pessoas e lugares específicos. Essas exclusões são realizadas manualmente no código. Isso pode ser trabalhoso e propenso a erros, além de exigir atualizações frequentes para lidar com novas entidades a serem excluídas.

Sensibilidade ao contexto: O código atual não leva em consideração o contexto das entidades ao realizar a extração. Isso significa que as entidades podem ser extraídas independentemente de seu contexto ou significado no texto. Considerar o contexto pode melhorar a precisão e relevância das entidades extraídas.

3.2 Perspetivas de desenvolvimentos futuros

Utilização de modelos mais avançados: A substituição do modelo do **Spacy** por modelos mais avançados de processamento de linguagem natural, como **BERT** (Bidirectional Encoder Representations from Transformers) ou **GPT** (Generative Pre-trained Transformer), poderia melhorar a qualidade da extração de entidades, bem como possibilitar outras tarefas de análise de texto, por exemplo sumarização de conteúdos.

Expansão das funcionalidades: Além da extração de entidades, o código pode ser expandido para incluir outras funcionalidades, como deteção de tópicos ou identificação de padrões específicos no texto, isso tornaria o código mais versátil e útil para uma variedade de tarefas de análise de texto.

Melhorias na interface: A interface **Streamlit** pode ser aprimorada para oferecer recursos adicionais, como filtragem interativa das entidades, exibição de contextos relevantes das entidades extraídas ou até mesmo a integração com outras ferramentas de visualização de dados.

Essas são apenas algumas das limitações e possibilidades de desenvolvimento futuro para o projeto, com a disponibilidade de mais tempo pesquisa e aprimoramento contínuo das técnicas de PLN, espera-se que essas limitações sejam superadas e muitas outras e novos recursos sejam adicionados para melhorar ainda mais a análise de texto e sua aplicação em mais áreas.

4 MATERIAL PARA USO DA FERRAMENTA E MAIS

Para o acesso completo da ferramenta completa (códigos, ficheiros, README e outros Encontre aqui disponibilizado o repositório utilizado na unidade Curricular onde tem igualmente os exercícios feitos durante o semestre como a padronização dos “markdowns” e ficheiros das obras, exercícios do amor de perdição, e do Harry potter, as propostas de visualização entre outros.

Aqui ficara a saber também como utilizar melhor a ferramenta e fazer sugestão para o melhorar ou corrigir a ferramenta:

https://github.com/rlopesneto/AVD2023-Renato

5 BIBLIOGRAFIA

OpenAI. (2023). ChatGPT. Chat.openai.com. https://chat.openai.com/

spaCy · Industrial-strength Natural Language Processing in Python. (2015). SpaCy. https://spacy.io/

NLTK. (2009). Natural Language Toolkit — NLTK 3.4.4 documentation. Nltk.org. https://www.nltk.org/

Welcome to Streamlit 👋. (2022, September 11). GitHub. https://github.com/streamlit/streamlit

Bad charts - Newsletter. (n.d.). DataJournalism.com. Retrieved June 4, 2023, from https://datajournalism.com/read/newsletters/bad-charts

Rosling, H. (2017). The best stats you've ever seen. Ted.com; TED Talks. https://www.ted.com/talks/hans_rosling_the_best_stats_you_ve_ever_seen