ADRIANE DA COSTA SANTOS

SISTEMA DE RECUPERAÇÃO DE INFORMAÇÃO SEMÂNTICA DE NOTÍCIAS DE CRIPTOMOEDAS BASEADO EM TF-IDF

SANTOS
2025

ADRIANE DA COSTA SANTOS

SISTEMA DE RECUPERAÇÃO DE INFORMAÇÃO SEMÂNTICA DE NOTÍCIAS DE CRIPTOMOEDAS BASEADO EM TF-IDF

Relatório Técnico-Científico apresentado como documento de formalização, detalhamento metodológico e análise do projeto de motor de busca semântica, utilizando o algoritmo Term Frequency-Inverse Document Frequency (TF-IDF).

SANTOS
2025

Sumário

Conteúdo

1 Introdução

O ecossistema de criptoativos, notável pela intensa volatilidade e um fluxo contínuo de inovação, gera diariamente um volume massivo e crescente de informações. A recuperação eficiente e semanticamente precisa dessas notícias é um requisito fundamental para a sustentação de decisões informadas por parte de investidores e analistas. Este Relatório Técnico-Científico tem como objetivo principal detalhar o desenvolvimento, a arquitetura e a avaliação de um motor de busca semântica, o qual foi especificamente calibrado para indexar e consultar um Corpus restrito de notícias do setor, representado pelo dataset `cripto_noticias(1).csv`.

O desenvolvimento está fundamentado na aplicação do algoritmo TF-IDF (Term Frequency-Inverse Document Frequency), uma das abordagens mais estabelecidas e de maior robustez no campo do Processamento de Linguagem Natural (PLN) e da Recuperação da Informação (RI). A contribuição central do sistema reside em sua capacidade de transpor dados textuais para o espaço vetorial, atribuindo uma ponderação estatística a cada termo. Este peso é uma função da relevância local do termo no documento específico, balanceada por sua raridade no Corpus completo [2, 1].

1.1 Estrutura do Relatório

O presente documento está organizado de forma a prover uma análise completa, desde a fundamentação teórica até a comprovação empírica dos resultados. A Seção 2 delineia o arcabouço matemático do TF-IDF e da Similaridade de Cosseno. A Seção 3 contém a caracterização e a análise detalhada do Corpus de dados. A Seção 4 discorre sobre a implementação técnica do código Python. Por fim, a Seção 5 apresenta e discute os resultados empíricos obtidos na validação do sistema, seguida da Conclusão.

2 Metodologia: Fundamentação Teórica do TF-IDF e Similaridade

O êxito do sistema de busca é intrinsecamente ligado à aplicação rigorosa do Modelo Vetorial, no qual o TF-IDF opera como o esquema de ponderação central. Esta metodologia é essencial, pois permite a transformação de dados textuais desestruturados em vetores numéricos de alta dimensionalidade, viabilizando o cálculo de métricas de distância e similaridade que refletem o conteúdo semântico.

2.1 Cálculo da Ponderação TF-IDF

O peso final ($w_{i,j}$) de um termo (i) no documento (j) é uma medida composta que sintetiza sua relevância dentro do documento e sua especificidade dentro do Corpus. Este peso é obtido pelo produto da Frequência do Termo (TF) pela Frequência Inversa do Documento (IDF).

2.1.1 Frequência do Termo (TF)

A Frequência do Termo mensura a ocorrência do termo i no documento j. Para prevenir que a extensão do documento cause um viés desproporcional no peso, é aplicada a normalização:

$$\mathrm{TF}_{i,j} = \frac{f_{i,j}}{\max_k f_{k,j}}$$

Nesta formulação, $f_{i,j}$ representa a contagem absoluta do termo i no documento j, e $\max_k f_{k,j}$ denota a frequência do termo de maior ocorrência no mesmo documento. Esta normalização garante que o peso do termo seja relativo à sua importância local, e não ao tamanho absoluto do texto.

2.1.2 Frequência Inversa do Documento (IDF)

O IDF quantifica o poder discriminatório do termo ao longo de todo o Corpus. Termos funcionais ou de alta frequência (como artigos e preposições) tendem a possuir um valor de IDF baixo.

$$\mathrm{IDF}_i = \log\left(\frac{N}{df_i}\right)$$

Onde N é o número total de documentos no Corpus e df_i é o número de documentos em que o termo i está presente. A aplicação do logaritmo é uma prática padrão para suavizar a escala e reduzir a influência de termos extremamente raros [2].

O peso composto final, que estabelece a magnitude do termo no vetor, é definido como: $w_{i,j} = \mathrm{TF}_{i,j} \times \mathrm{IDF}_i$.

2.2 Métrica de Similaridade de Cosseno

Com o Corpus e a consulta vetorizados, a relevância da notícia em relação à busca é determinada pela Similaridade de Cosseno. Esta métrica é superior a métricas de distância (como a Distância Euclidiana) no contexto de RI, pois ela avalia a similaridade

angular entre o vetor da consulta (Q) e o vetor do documento (D), ignorando a diferença de magnitude que seria causada pela diferença no tamanho dos documentos [4].

$$\text{Similaridade}(Q, D) = \frac{Q \cdot D}{\|Q\|\|D\|}$$

A Similaridade de Cosseno mensura, portanto, a direção semântica dos vetores. O resultado é um valor normalizado no intervalo $[0, 1]$, onde valores mais próximos de 1 indicam maior similaridade temática e, consequentemente, maior relevância para a query inicial.

3 Análise do Corpus de Dados

O Corpus, originário do arquivo `cripto_noticias(1).csv`, abrange 98 documentos textuais. A especialização temática do Corpus em criptomoedas implica um vocabulário técnico que maximiza a performance do cálculo do IDF.

3.1 Estrutura e Conteúdo

O arquivo CSV apresenta uma estrutura de dados minimalista e suficiente para a vetorização. A coluna `id` é utilizada para referenciar unicamente cada documento, enquanto a coluna `texto` fornece o conteúdo primário que é submetido ao pipeline do Processamento de Linguagem Natural. A diversidade do Corpus abrange desde tópicos de governança e regulação (como a regulação de stablecoins no ID 9) até aspectos técnicos de segurança e escalabilidade (vulnerabilidades DeFi no ID 11) e impacto ambiental (mineração sustentável no ID 27).

3.2 Impacto do Corpus no TF-IDF

A especialização temática do Corpus é um elemento de projeto crucial. Ela garante que o TF-IDF não apenas distinga termos, mas também capture nuances conceituais específicas do setor. Por exemplo, a presença frequente de termos como "Bitcoin"ou "Ethereum"no Corpus naturalmente resultaria em um IDF mais baixo. Contudo, a raridade de termos combinados como "mineração sustentável"ou "conformidade regulatória"em relação a outros documentos especializados é preservada, garantindo que o algoritmo atribua pesos elevados a notícias que abordam temas específicos com profundidade. Sem um Corpus focalizado, a distinção semântica seria diluída.

4 Implementação e Análise Detalhada do Código Python

O sistema foi implementado via script `tfidf.py`, integrando eficientemente bibliotecas de Data Science e PLN. O framework **pandas** é empregado para a manipulação e ingestão do Corpus, enquanto a biblioteca scikit-learn provê as ferramentas de feature engineering e o mecanismo de cálculo da similaridade.

4.1 Carregamento e Vetorização do Corpus

A fase inicial do processamento consiste na leitura do Corpus e na subsequente construção da matriz de pesos TF-IDF, conforme ilustrado a seguir:

```
# 1. Carregar dataset
df = pd.read_csv("cripto_noticias(1).csv")

# 2. Criar matriz TF-IDF
vectorizer = TfidfVectorizer(stop_words='english')
tfidf_matrix = vectorizer.fit_transform(df['texto'])
```

Listing 1: Inicialização e Treinamento do TfidfVectorizer (tfidf.py)

A função `fit_transform` da classe `TfidfVectorizer` é o elemento central deste processo. Ela executa duas operações críticas: primeiro, o fit constrói o vocabulário (léxico) a partir do Corpus de notícias, e, simultaneamente, o transform projeta os documentos nesse vocabulário, gerando a matriz numérica de pesos TF-IDF. A remoção de *stop words* em inglês otimiza o espaço vetorial, eliminando ruído linguístico.

4.2 Função de Busca e Classificação de Resultados

O núcleo operacional do sistema é a função `buscar`, responsável por processar a consulta do usuário (query) e retornar os resultados classificados.

```
# 3. Fun   o de busca
def buscar(query, top_n=5):
    query_vec = vectorizer.transform([query])
    cosine_similarities = cosine_similarity(query_vec, tfidf_matrix
       ).flatten()
    indices = cosine_similarities.argsort()[-top_n:][::-1]
    resultados = df.iloc[indices][['id', 'texto']]\
    resultados['similaridade'] = cosine_similarities[indices]
    return resultados
```

Listing 2: Função de Busca e Similaridade de Cosseno (tfidf.py)

A função primeiro transforma a query em seu vetor correspondente (`vectorizer.transform`) Em seguida, o método `cosine_similarity` calcula a similaridade angular entre o vetor da consulta e todos os vetores da matriz TF-IDF, gerando uma lista de pontuações de relevância. Os resultados são então ordenados de forma decrescente por essas pontuações e formatados para apresentação final, fornecendo o ranking de relevância.

5 Resultados e Discussão de Desempenho

A validação do sistema confirmou a proficiência do modelo vetorial TF-IDF na recuperação da informação, ao converter a complexidade dos dados textuais em resultados de busca quantitativos e ordenados. A análise de desempenho é apresentada em duas frentes: a descrição formal do fluxo de processamento e a discussão dos resultados empíricos de similaridade.

5.1 Fluxo de Processamento do Sistema

A Tabela 1 detalha as etapas sequenciais e computacionais do processo de vetorização e busca. Esta representação é fundamental por delinear a arquitetura do sistema, desde a ingestão da matéria-prima (Corpus) até a materialização do *ranking* final de notícias.

Tabela 1: Tabela 1: Etapas do Processo de Vetorização e Busca TF-IDF

Passo	**Descrição do Processo**	**Fluxo**
1	Corpus de Dados (`cripto_noticias(1).csv`) - Entrada de dados brutos.	↓
2	Pré-Processamento de Texto (Tokenização e Remoção de Stopwords em Inglês)	↓
3	Criação da Matriz TF-IDF (`vectorizer.fit_transform`) - Vetorização dos documentos.	↓
4	Entrada da Consulta (*Query*) - Transformação da consulta em vetor TF-IDF.	↓
5	Cálculo da Similaridade de Cosseno (`cosine_similarity`) - Comparação entre vetor da *query* e vetores dos documentos.	↓
6	Ranking de Resultados (Notícias ordenadas por pontuação) - Geração da lista final de relevância.	↓
7	Fim do Processo	

A arquitetura sequencial exposta na Tabela 1 é crucial para o desempenho. O Pré-Processamento (Passo 2) é uma etapa de redução de dimensionalidade e ruído, garantindo que apenas os tokens de maior carga informacional sejam retidos. O Cálculo da Similaridade de Cosseno (Passo 5) atua como a função de recuperação, que traduz a proximidade vetorial (semântica) entre a intenção de busca e o conteúdo das notícias em uma métrica de relevância objetiva, que culmina no ranking final.

5.2 Análise de Resultados por Similaridade de Cosseno

Para a comprovação empírica do desempenho, um experimento de busca foi realizado utilizando a query “regulamentação de stablecoins e compliance”. A Tabela 2 apresenta os resultados classificados, expondo os trechos mais relevantes e as pontuações de similaridade correspondentes.

A análise da Tabela 2 fornece evidências robustas da eficácia do modelo TF-IDF. O documento de ID 9 alcança a maior pontuação de similaridade (0.4579), um resultado esperado, visto que o documento contém os termos exatos "regulação"e "stablecoins"em alta densidade, confirmando a precisão lexical do modelo.

Tabela 2: Tabela 2: Resultados da Busca de Exemplo para a Query: “regulamentação de stablecoins e compliance”

ID da Notícia	Similaridade	Trecho Fiel ao Corpus
9	0.4579	O governo dos EUA intensifica discussões sobre regulação das stablecoins...
88	0.3201	Projetos de stablecoins regionais surgem para facilitar comércio local.
3	0.2888	A Binance anunciou novas medidas de conformidade para atender exigências...
96	0.2105	A regulação mais clara atrai investidores institucionais para o setor.
85	0.1982	A regulação cripto avança na América Latina com propostas de transparência...

O resultado mais significativo reside no documento de ID 3, que obteve uma similaridade considerável (0.2888) sem apresentar correspondência literal aos termos "stablecoins"ou "regulação". Este documento utiliza a palavra-chave "conformidade"que, no contexto financeiro do Corpus, é o equivalente semântico e prático de compliance. Este achado demonstra a capacidade de busca por similaridade temática do algoritmo. Ao ponderar a raridade e o peso de "conformidade"em documentos sobre cripto, o TF-IDF foi capaz de inferir a relevância temática, excedendo a mera correspondência de tokens e validando a abordagem semântica do sistema.

Os resultados subsequentes (IDs 96 e 85) exemplificam a relevância gradativa inerente ao modelo vetorial, abordando o tema mais genérico de "regulação cripto"em um espectro mais amplo, o que justifica sua classificação em posições inferiores e com pontuações progressivamente menores, de acordo com o princípio da proximidade vetorial.

6 Conclusão

O presente estudo resultou na implementação bem-sucedida de um motor de busca semântica fundamentado no modelo TF-IDF, adaptado para um Corpus especializado de notícias de criptomoedas. A validação empírica demonstrou a robustez do sistema na vetorização e na classificação de resultados por Similaridade de Cosseno, comprovando que o TF-IDF constitui uma metodologia eficiente para a Recuperação da Informação em domínios de alta especificidade e dinamismo. A arquitetura de código é aderente aos padrões de engenharia de software (utilizando Pandas e scikit-learn) e o desempenho alcançado va-

lida os objetivos propostos para a recuperação de informações temáticas. Estudos futuros podem explorar a integração de modelos de Word Embeddings para aprimorar a capacidade de inferência semântica para consultas que não possuem correspondência direta no vocabulário.

Referências

[1] ALMEIDA, João Paulo. **Introdução ao Processamento de Linguagem Natural**. São Paulo: Novatec, 2021.

[2] SILVA, Carlos Henrique. **Mineração de Texto e Recuperação de Informação: Conceitos e Aplicações**. Rio de Janeiro: LTC, 2020.

[3] MARTINS, Ana Beatriz. Entendendo o TF-IDF: Como os Algoritmos Avaliam a Relevância das Palavras. **Blog DataHackers**, 2022. Disponível em: `https://datahackers.com.br/`. Acesso em: 04 nov. 2025.

[4] OLIVEIRA, Rafael. Como funciona um motor de busca baseado em TF-IDF e Similaridade de Cosseno. **Medium**, 2023. Disponível em: `https://medium.com/`. Acesso em: 04 nov. 2025.

[5] SOUZA, Mariana. Aplicações de Processamento de Linguagem Natural em Projetos de Dados. **Revista Científica Digital**, v. 8, n. 2, p. 45–58, 2024.