TF-IDF TRABALHO - Enzo Ura

Enzo Shinji Sugano Ura

Novembro 2025

1 Introdução

Sistema de Recomendação Odontológica com TF-IDF
Aluno: **Enzo Shinji Sugano Ura**
4 de Novembro de 2025

2 Introdução

O trabalho tem como objetivo desenvolver um sistema de recomendação de procedimentos odontológicos utilizando a técnica de TF-IDF (Term Frequency-Inverse Document Frequency) e a métrica de similaridade do cosseno. A proposta é permitir que o usuário digite um procedimento odontológico ou uma breve descrição, e o sistema retorne os tratamentos mais semelhantes com base em um banco de dados textual. A escolha do tema visa demonstrar a aplicação prática de técnicas de mineração de texto na área da saúde.

3 Descrição do Dataset

O conjunto de dados é composto por 10 procedimentos odontológicos, com suas respectivas descrições resumidas:

- **Limpeza dental** — Limpeza dental: remoção de placa bacteriana e tártaro, polimento e aplicação de flúor.
- **Restauração dental** — Restauração dental: reparar dentes danificados por cáries, fraturas ou desgaste.
- **Extração dental** — Extração dental: remoção de dentes comprometidos por cáries profundas, infecção ou falta de espaço.
- **Clareamento dental** — Clareamento dental: clarear o tom dos dentes com agentes clareadores.
- **Canal (endodontia)** — Canal (endodontia): remover a polpa dentária infectada e selar o interior do dente.
- **Implante dentário** — Implante dentário: colocação de pino de titânio no osso para substituir dentes ausentes.

- **Aparelho ortodôntico** — Aparelho ortodôntico: dispositivo fixo ou móvel para corrigir o alinhamento dos dentes.
- **Profilaxia infantil** — Profilaxia infantil: limpeza preventiva em crianças para evitar cáries e gengivite.
- **Raspagem periodontal** — Raspagem periodontal: remoção de tártaro abaixo da gengiva em casos de periodontite.
- **Prótese dentária** — Prótese dentária: substituição de dentes perdidos por próteses fixas ou removíveis.

4 Metodologia

O sistema foi implementado em Python utilizando a biblioteca scikit-learn. Cada descrição textual é convertida em um vetor numérico com base no modelo TF-IDF.

4.1 Pré-processamento de Texto

Foram implementadas as seguintes etapas de pré-processamento:

- **Normalização de texto**: Conversão para minúsculas e remoção de acentos
- **Remoção de stop words**: Utilização de lista extensa de palavras irrelevantes em português
- **Processamento com n-gramas**: Consideração de sequências de 1 e 2 palavras (unigramas e bigramas)

4.2 Cálculo do TF-IDF

O TF-IDF mede a relevância de uma palavra dentro de um documento e em relação ao conjunto total de documentos. A técnica foi aplicada considerando:

- Frequência dos termos nos documentos individuais
- Frequência inversa nos documentos do corpus completo
- Ponderação para termos mais relevantes e distintivos

5 Análise das Semelhanças e dos Ângulos

A análise dos ângulos mostra como o modelo identifica relações semânticas entre os procedimentos. Por exemplo:

- **Limpeza dental e Profilaxia infantil** — ambas envolvem remoção de placa e prevenção, portanto apresentam alta similaridade (ângulo pequeno, cerca de 25° a 30°).
- **Raspagem periodontal e Limpeza dental** — possuem objetivos semelhantes, mas uma é mais profunda. Similaridade moderada (ângulo de 40° a 50°).

- **Clareamento dental e Implante dentário** — tratam de áreas distintas (estética e cirurgia), logo o ângulo entre os vetores é grande (próximo de 80°), indicando baixa similaridade.
- **Canal e Extração dental** — embora sejam tratamentos diferentes, ambos envolvem dentes danificados e processos de remoção de tecido, com similaridade média (ângulo de 45° a 55°).

Essas comparações mostram que o TF-IDF, mesmo sendo um método simples, consegue capturar padrões relevantes de contexto. O "ângulo"funciona como uma medida geométrica de semelhança textual: quanto mais próximas as descrições, mais "paralelos"são os vetores no espaço vetorial.

6 Interação com o Usuário

O programa é interativo: o usuário informa o nome ou uma descrição do procedimento e o sistema retorna os três mais semelhantes com suas descrições completas e scores de similaridade. Para encerrar a execução, basta digitar **sair**.

7 Código em Python

```
import unicodedata
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity

def normalizar(texto):
    """
    Normaliza o texto: converte para minúsculas e remove acentos (diacríticos).
    """
    texto = texto.lower()
    # Remove acentos (diacríticos)
    texto = ''.join(c for c in unicodedata.normalize('NFD', texto)
                    if unicodedata.category(c) != 'Mn')
    return texto

procedimentos = {
    "Limpeza dental": "Limpeza dental: remoção de placa bacteriana e tártaro,
                       polimento e aplicação de flúor.",
    "Restauração dental": "Restauração dental: reparar dentes danificados por
                           cáries, fraturas ou desgaste.",
    "Extração dental": "Extração dental: remoção de dentes comprometidos por
                       cáries profundas, infecção ou falta de espaço.",
    "Clareamento dental": "Clareamento dental: clarear o tom dos dentes com
                           agentes clareadores.",
    "Canal (endodontia)": "Canal (endodontia): remover a polpa dentária infectada
                           e selar o interior do dente.",
    "Implante dentário": "Implante dentário: colocação de pino de titânio no osso
                         para substituir dentes ausentes.",
```

```
    "Aparelho ortodôntico": "Aparelho ortodôntico: dispositivo fixo ou móvel
                            para corrigir o alinhamento dos dentes.",
    "Profilaxia infantil": "Profilaxia infantil: limpeza preventiva em crianças
                           para evitar cáries e gengivite.",
    "Raspagem periodontal": "Raspagem periodontal: remoção de tártaro abaixo da
                            gengiva em casos de periodontite.",
    "Prótese dentária": "Prótese dentária: substituição de dentes perdidos por
                        próteses fixas ou removíveis."
}

nomes = list(procedimentos.keys())
# Aplica a normalização nas descrições para o vetorizador
descricoes = [normalizar(texto) for texto in procedimentos.values()]

# Lista de stop words em Português
portuguese_stop_words = [
    'de', 'a', 'o', 'que', 'e', 'é', 'do', 'da', 'em', 'um', 'uma', 'para',
    'com', 'não', 'uma', 'os', 'as', 'dos', 'das', 'pelo', 'pela', 'pelos',
    'pelas', 'ao', 'aos', 'à', 'às', 'dele', 'dela', 'deles', 'delas', 'aquele',
    'aquela', 'aqueles', 'aquelas', 'isto', 'aquilo', 'este', 'esta', 'estes',
    'estas', 'isso', 'esse', 'essa', 'esses', 'essas', 'no', 'na', 'nos', 'nas',
    'por', 'mais', 'mas', 'ao', 'tempo', 'se', 'depois', 'quando', 'como', 'qual',
    'ser', 'ter', 'ir', 'vir', 'estar', 'fazer', 'dizer', 'poder', 'ver', 'saber',
    'querer', 'chegar', 'dar', 'falar', 'comer', 'beber', 'cantar', 'dançar',
    'andar', 'correr', 'nadar', 'voar', 'dormir', 'acordar', 'levantar', 'sentar',
    'cair', 'subir', 'descer', 'entrar', 'sair', 'abrir', 'fechar', 'ligar',
    'desligar', 'começar', 'terminar', 'continuar', 'parar', 'mudar', 'achar',
    'pensar', 'sentir', 'ouvir', 'ver', 'olhar', 'gostar', 'amar', 'odiar',
    'precisar', 'usar', 'ter', 'haver', 'ser', 'estar', 'ir', 'vir', 'dar',
    'fazer', 'dizer', 'poder', 'ver', 'saber', 'querer', 'chegar', 'dar',
    'falar', 'comer', 'beber', 'cantar', 'dançar', 'andar', 'correr', 'nadar',
    'voar', 'dormir', 'acordar', 'levantar', 'sentar', 'cair', 'subir', 'descer',
    'entrar', 'sair', 'abrir', 'fechar', 'ligar', 'desligar', 'começar',
    'terminar', 'continuar', 'parar', 'mudar', 'achar', 'pensar', 'sentir',
    'ouvir', 'ver', 'olhar', 'gostar', 'amar', 'odiar', 'precisar', 'usar'
]

# Inicializa o vetorizador TF-IDF com as stop words em português e n-gramas de 1 e
vetorizador = TfidfVectorizer(stop_words=portuguese_stop_words, ngram_range=(1, 2))
# Ajusta (fit) e transforma (transform) as descrições em uma matriz TF-IDF
matriz_tfidf = vetorizador.fit_transform(descricoes)

print("=== Sistema de Recomendação Odontológica (TF-IDF) ===")
print("Digite o nome ou descrição do procedimento que deseja encontrar.")
print("Quando quiser sair, digite 'sair'.\n")

while True:
    # Coleta a entrada do usuário
```

```
entrada = input("Qual procedimento você procura? ").strip().lower()

if entrada == "sair":
    print("Encerrando o sistema... até logo!")
    break

# Normaliza a entrada
entrada = normalizar(entrada)

# Transforma a entrada em vetor TF-IDF (usa apenas transform, pois já foi fitad
entrada_tfidf = vetorizador.transform([entrada])

# Calcula a similaridade do cosseno entre a entrada e todos os procedimentos
similaridades = cosine_similarity(entrada_tfidf, matriz_tfidf)[0]

# Obtém os índices ordenados por similaridade (do maior para o menor)
indices = similaridades.argsort()[::-1]

print("\nProcedimentos mais semelhantes:")
# Exibe os 3 procedimentos mais semelhantes
for i in indices[:3]:
    print(f"→ {nomes[i]} - Similaridade: {similaridades[i]:.2f}")
    print(f"  Descrição: {procedimentos[nomes[i]]}\n")
print("-" * 60)
```

8 Conclusão

O uso do TF-IDF aliado à similaridade do cosseno mostrou-se eficaz na criação de um sistema de recomendação textual simples e funcional. Mesmo com um conjunto de dados pequeno, a técnica conseguiu identificar relações reais entre os procedimentos odontológicos com base na linguagem natural.

A interpretação dos ângulos entre vetores reforça a importância do modelo: ângulos pequenos indicam forte correlação entre termos, enquanto ângulos grandes revelam diferenças semânticas significativas.

As melhorias implementadas no código atual incluem:

- **Pré-processamento mais robusto** com normalização de texto e remoção de acentos
- **Otimização do processamento linguístico** com lista extensa de stop words em português
- **Melhoria na análise contextual** com uso de n-gramas (unigramas e bigramas)
- **Interface mais informativa** com exibição de descrições completas e scores de similaridade

Como aprimoramentos futuros, pode-se incluir:

- Análise semântica com embeddings (Word2Vec, BERT);

- Expansão do dataset com descrições mais detalhadas;
- Interface gráfica para facilitar a interação do usuário;
- Integração com banco de dados para armazenamento dinâmico de procedimentos.