

Tầm quan trọng của TF-IDF trong xử lý ngôn ngữ tự nhiên (NLP)

TF-IDF là gì?

TF-IDF là viết tắt của:

- **TF** - Term Frequency (Tần suất xuất hiện của từ)
- **IDF** - Inverse Document Frequency (Tần suất nghịch đảo của từ trong toàn bộ tài liệu)

TF-IDF là một kỹ thuật biến văn bản thành số để máy tính hiểu, giúp xác định **từ nào là quan trọng nhất trong một tài liệu** trong số nhiều tài liệu.

Công thức

TF (Term Frequency)

Tính tần suất của một từ trong một văn bản:

$$TF(t, d) = \frac{\text{số lần xuất hiện của từ } t \text{ trong văn bản } d}{\text{tổng số từ trong văn bản } d}$$

DF (Inverse Document Frequency)

Tính độ hiếm của từ trong tập tài liệu:

$$IDF(t) = \log \left(\frac{N}{df(t)} \right)$$

- N : Tổng số tài liệu
- $df(t)$: Số tài liệu chứa từ t

• **TF-IDF**

$$TF-IDF(t, d) = TF(t, d) \times IDF(t)$$

Vì sao TF-IDF quan trọng?

TF-IDF giúp...	Vì sao
Xác định từ khóa chính	Vì từ thường xuyên xuất hiện nhưng không phổ biến trong nhiều tài liệu
Cải thiện tìm kiếm thông tin	Giúp hệ thống tìm kiếm xác định tài liệu liên quan nhất
Loại bỏ từ không quan trọng	Như "là", "và", "nhưng" thường xuất hiện khắp nơi (IDF thấp)

Ví dụ cụ thể

Dữ liệu:

```
documents = [  
    "Tôi thích học lập trình Python",  
    "Python là một ngôn ngữ lập trình mạnh mẽ",  
    "Tôi học Python mỗi ngày"  
]
```

Mã Python:

```
from sklearn.feature_extraction.text import TfidfVectorizer  
  
docs = [  
    "Tôi thích học lập trình Python",  
    "Python là một ngôn ngữ lập trình mạnh mẽ",  
    "Tôi học Python mỗi ngày"  
]  
  
# Khởi tạo TF-IDF vectorizer  
vectorizer = TfidfVectorizer()
```

```
# Biến đổi văn bản
X = vectorizer.fit_transform(docs)

# In từ điển và ma trận TF-IDF
print("Từ vựng:", vectorizer.get_feature_names_out())
print("Ma trận TF-IDF:\n", X.toarray())
```

Kết quả mẫu:

- Từ "Python" xuất hiện nhiều, nhưng vì nó không xuất hiện trong mọi văn bản \Rightarrow có IDF tương đối cao
- Từ như "Tôi", "là" có IDF thấp \Rightarrow không quan trọng

Ứng dụng thực tế của TF-IDF

Ứng dụng	Mô tả
Máy tìm kiếm (search engine)	Xác định nội dung liên quan đến truy vấn
Phân loại văn bản	Ví dụ: Xác định email spam
Gợi ý nội dung	Đề xuất bài viết liên quan
Chatbot	Xác định ý định người dùng trong câu hỏi

Kết hợp TF-IDF với cosine similarity để tìm văn bản giống nhau

Tìm tài liệu (văn bản) nào giống nhất với một truy vấn văn bản do người dùng nhập vào.

Bước 1: Dữ liệu đầu vào

Giả sử bạn có một danh sách các tài liệu:

```
documents = [
    "Tôi thích học lập trình Python",
    "Python là một ngôn ngữ mạnh mẽ và linh hoạt",
    "Học máy và trí tuệ nhân tạo đang rất phát triển",
    "Tôi thường lập trình bằng Python mỗi ngày",
    "Bóng đá là môn thể thao tôi yêu thích"
]
```

Và người dùng nhập truy vấn:

```
query = "Tôi muốn học lập trình bằng Python"
```

Bước 2: Cài đặt TF-IDF + Cosine Similarity

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

# Danh sách tài liệu + truy vấn
all_texts = documents + [query]

# Vector hóa bằng TF-IDF
vectorizer = TfidfVectorizer()
tfidf_matrix = vectorizer.fit_transform(all_texts)

# Tách truy vấn ra khỏi ma trận
query_vec = tfidf_matrix[-1]
doc_vecs = tfidf_matrix[:-1]

# Tính cosine similarity giữa truy vấn và các tài liệu
similarities = cosine_similarity(query_vec, doc_vecs).flatten()

# In kết quả
for idx, score in enumerate(similarities):
    print(f"Độ tương đồng với tài liệu {idx + 1}: {score:.4f}")

# Tìm tài liệu giống nhất
most_similar_idx = similarities.argmax()
print(f"\n☐ Tài liệu giống truy vấn nhất: {documents[most_similar_idx]}")
```

Kết quả đầu ra

```
Độ tương đồng với tài liệu 1: 0.5634
Độ tương đồng với tài liệu 2: 0.3211
Độ tương đồng với tài liệu 3: 0.0925
Độ tương đồng với tài liệu 4: 0.7012
Độ tương đồng với tài liệu 5: 0.0000
```

```
Tài liệu giống truy vấn nhất: Tôi thường lập trình bằng Python mỗi ngày
```

Giải thích:

- **TF-IDF** giúp mã hóa mức độ quan trọng của từ trong từng văn bản.

- **Cosine similarity** đo góc giữa các vector văn bản → góc càng nhỏ thì văn bản càng giống nhau.
- Ta tìm ra tài liệu có độ tương đồng cao nhất với truy vấn.

Ứng dụng:

- Tìm kiếm tài liệu
- Gợi ý bài viết liên quan
- So khớp nội dung người dùng trong chatbot
- Phát hiện trùng lặp nội dung

Tác giả: Đỗ Ngọc Tú
Công Ty Phần Mềm VHTSoft

Phiên bản #3

Được tạo 4 tháng 5 2025 10:10:17 bởi Đỗ Ngọc Tú

Được cập nhật 6 tháng 5 2025 09:52:34 bởi Đỗ Ngọc Tú