### tfidf

文字列から重要な単語を抽出します。出現頻度が低く、相対的に特異性の高い単語により高いスコアを付与し、有意な情報を抽出します。本コマンドは各単語の頻度（TF）と逆文書頻度（IDF）を組み合わせてTF-IDFスコアを算出します。

#### 構文

フィールドリストで指定した文字列から単語頻度を計算

tfidf [delimiter=STR\_DELIMITER] [numeric=t] [op=build] [threshold=INT] FIELD\_LIST

フィールドリストで指定した文字列から単語ごとのTF-IDFスコアを計算

tfidf [delimiter=STR\_DELIMITER] [numeric=t] [op=query] [threshold=INT] FIELD\_LIST

サブクエリで単語頻度データを取得し、フィールドリストで指定した文字列から単語ごとのTF-IDFスコアを計算

tfidf [delimiter=STR\_DELIMITER] [numeric=t] [op=query] [threshold=INT] FIELD\_LIST [ SUBQUERY ]

**delimiter=STR\_DELIMITER**

文字列の区切り文字リスト（デフォルト：空白文字）。区切り文字をセミコロン（;）やスラッシュ（/）などに変更することで、単語をより細かく分割したり、特定のパターンで処理できます。区切り文字として認識させたい文字をスペースなしで入力してください。

例えば、以下のように指定できます。

delimiter=" ;&/:-=\_.,[](){}\\n\\t"

1. 上記文字列の先頭の特殊文字は空白文字です。
2. タブ（\t）や改行（\n）などの特殊文字は "\n"、"\t" のようにエスケープして入力してください。

**numeric=t**

同じ長さの数字文字列トークンを同一として扱うよう指定します。例えば "123" と "456" は長さが同じため同一として処理されます（デフォルト：f）。

一定長の数字が繰り返し出現する場合、それらを同一として扱いたい場合に使用します。副次的にパフォーマンスの向上効果もあります。

**op={build|query}**

コマンドの動作を指定します（デフォルト：query）。

1. query：単語の頻度と逆文書頻度を計算します。
2. build：単語の頻度のみを計算します。

op=buildで計算した単語頻度は自動的には保存されません。後続のクエリでimportコマンドを利用して計算結果をテーブルに保存してください。定期的な更新が必要な場合は、行動プロファイルやスケジュールされたクエリの利用を検討してください。

**threshold=INT**

単語の頻度が指定値以下の場合、その単語を無視します（デフォルト：0。0の場合は無視しません）。例えば、threshold=5の場合、頻度が5以下の単語は無視されます。

**FIELD\_LIST**

複数フィールドを1つの文字列として扱い、分析します。例えば、フィールドaが "abc def"、フィールドbが "ghi jkl" の場合、4つのトークン（"abc", "def", "ghi", "jkl"）を持つ文字列（"abc def ghi jkl"）として扱われます。

**[ SUBQUERY ]**

単語頻度を保存したテーブルや行動プロファイルから単語頻度データを取得するクエリ文を角括弧（[ ]）内に入力します。このサブクエリはop=query時に利用可能です。

#### 説明

単語頻度（TF, Term Frequency）は、文書内で特定単語が繰り返し出現する回数を測定する指標です。文字列データにおいて単語の重要度を算出する際、その単語が文書内でどれだけ頻繁に現れるかを評価します。単語が文書内で多く出現するほどTF値は大きくなります。文書が長くなると単純な頻度だけでは比較が難しくなるため、文書内の総単語数で割って正規化します。

逆文書頻度（IDF, Inverse Document Frequency）は、文字列データにおける単語の重要度を測定する指標です。ある単語が多くの文書で頻繁に出現する場合、その単語は情報量が少なく重要度が低いと見なされます。例えば、英語の "the" や "is" のような一般的な単語はほぼ全ての文書に現れるため、IDF値は小さくなります。一方、特定の文書でのみ主に出現する単語は情報量が多いとされ、高いIDF値を持ちます。

TF-IDFは単語頻度と逆文書頻度を組み合わせて、文字列データ内の単語重要度を算出する手法です。単語の頻度（TF）と逆文書頻度（IDF）を掛け合わせて計算します。TFが高い単語は文書内で頻繁に使われる単語、IDFが高い単語は文書集合内で特異性の高い単語として強調されます。これにより、よく出現する単語（ストップワード）は重要度が低く、文書ごとに固有の単語は重要度が高く評価されます。

韓国語のような膠着語では、助詞（例："〜은"、"〜는"）や語尾（例："〜합니다"、"〜하는"）など様々な接辞が含まれます。接辞は単語の意味を決定する上で重要な要素ではありませんが、不要な単語（ストップワード）が多いとTF-IDF計算結果に悪影響を及ぼすため、形態素解析によって「語幹」のみを抽出したり、ストップワードを除去する前処理が必要です。

パフォーマンス指標

以下はtfidfコマンドのパフォーマンス指標です。

1. **ビルド**：1秒あたり10万〜20万件処理
2. **サブクエリ未使用時**：1秒あたり5万〜10万件処理
3. **サブクエリ使用時**：1秒あたり10万〜20万件処理

ストリームクエリとして利用する場合は、事前にビルドしたデータをサブクエリとして活用してください。あらかじめビルドしたデータを利用することで、クエリ実行時の計算負荷を軽減し、リアルタイム処理速度を向上させることができます。

出力フィールド

出力フィールドは以下の通りです。

|  |  |  |
| --- | --- | --- |
| フィールド | 型 | 説明 |
| term | 文字列 | 単語 |
| \_idf | 64ビット整数 | 単語の逆文書頻度 |
| \_tfidf | 64ビット整数 | 単語のTF-IDFスコア。op=build時は出力されません。 |

#### 使用例

基本的な使用例

各文字列行に対して\_tfidfフィールドにTF-IDFスコアを付与します。

table iis\_result | tfidf numeric=t delimiter=" ;&/:-=\_.,\\n[](){}\\t" op=query line

頻度テーブルのビルド

各文字列のIDFスコアを\_idfフィールドに計算し、iis\_idfテーブルに保存します。頻繁に出現するほどスコアは低くなります。

table iis\_result | tfidf numeric=t delimiter=" ;&/:-=\_.,\\n[](){}\\t" op=build line | import iis\_idf

サブクエリの利用

事前にビルドしたIDFテーブルをサブクエリで読み込み、スコアを計算します。パフォーマンスが向上します。

table iis\_result | tfidf numeric=t delimiter=" ;&/:-=\_.,\\n[](){}\\t" op=query line [ table iis\_idf ]