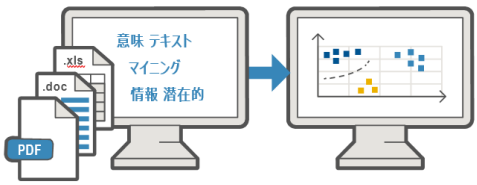
Text Analytics Toolbox のクイックスタートガイド



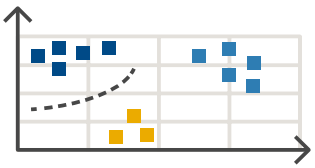
Text Analytics Toolbox™ は、テキストデータの前処理、解析、モデリングのためのアルゴリズムと可視化機能を提供します。ツールボックスを使用して作成されたモデルは、感情分析、予知保全、トピックモデリングなどのアプリケーションで使用できます。

関連情報: *mathworks.co.jp/products/text-analytics*

関数名	説明
`wordcloud`	bag-of-words あるいは LDA モデルからのワードクラウド作成
`wordCloudCounts`	ワードクラウド作成のための単語数カウント
`textscatter`	テキストの散布図
`textscatter3`	テキストの3次元散布図
`heatmap`	ヒートマップチャートの作成
`histcounts`	ヒストグラムのビンのカウント数
`discretize`	データをビンまたはカテゴリにグループ化

可視化

ワードクラウドとテキスト散布図を使用して、結果を要約して検証します。



モデリングと予測

Bag-of-words や学習済みの単語分散表現モデルを使用してテキストを数値表現に変換し、予測やトピックモデリングに特化した機械学習アルゴリズムを適用します。

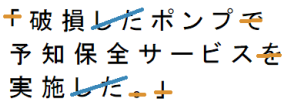
関数名	説明
`readWordEmbedding`	テキストファイルからの単語の分散表現の読み込み
`trainWordEmbedding`	単語の分散表現を学習
`word2vec/vec2word`	単語を分散ベクトルにマップ
`ldaModel`	潜在的ディリクレ配分法 (LDA) モデル
`lsaModel`	潜在意味解析 (LSA) モデル
`bagOfWords`	Bag-of-words モデル
`fitlda`	潜在的ディリクレ配分法 (LDA) モデルの学習
`fitlsa`	潜在意味解析 (LSA) モデルの学習
`predict`	文書内の主要な LDA トピックを予測
`fitdist`	データへの確率分布オブジェクトの近似
`fitrlinear`	高次元データに対する線形回帰モデルのあてはめ
`fitclinear`	高次元データに対する線形分類モデルのあてはめ
`fitcecoc`	サポート ベクター マシンまたはその他の分類器向けのマルチクラスモデルの近似

MathWorks®
Accelerating the pace of engineering and science

関数名	説明
`extractFileText`	PDF, Microsoft Word, テキストファイルからのデータ読み込み
`textscan`	テキストファイルまたは文字列からの書式付きデータの読み込み
`readtable`	ファイルからのテーブルの作成
`compose`	データを書式設定された string 配列に変換
`xlsread`	Microsoft Excel スプレッドシート ファイルの読み取り
`webread`	RESTful Web サービスからのコンテンツの読み取り
`TabularTextDatastore`	表形式テキスト ファイルのデータ ストア
`FileDatastore`	カスタム ファイル リーダーを使用するデータ ストア
`SpreadsheetDatastore`	スプレッドシート ファイルのデータ ストア

読み込み

Microsoft® Word® ファイル、 PDF、テキストファイル、スプレッドシートからテキストを読み込みます。



前処理

ありきたりな単語、句読点、および URL などの深い意味を持たない単語を取り除き、語幹抽出により単語の正規化を行います。

関数名	説明
`tokenizedDocument`	文書を単語の集合に分割
`normalizeWords`	Porter stemmer を使用して語形変化で使用される語尾を単語から削除
`bagOfWords`	Bag-of-words モデル
`stopWords`	ストップワードリスト
`context`	文書内の単語の出現を検索
`removeWords`	文書あるいは bag-of-words から特定の単語を削除
`removeLongWords`	文書あるいは bag-of-words から長い単語を削除
`removeShortWords`	文書あるいは bag-of-words から短い単語を削除
`removeInfrequentWords`	bag-of-words モデルから頻度の低い単語を削除
`erasePunctuation`	テキストや文書から句読点の削除

関数名	説明
`str = "Hello,world"`	string 変数を定義
`str = ["Hello", "World"]`	string 配列を定義
`str = string( C )`	キャラクタ配列 Cを string 型に変換
`str2double`	文字列のdouble型への変換
`strlength`	文字列の長さ
`isstring`	入力が string 配列かどうかを判別
`join`	文字列の結合
`split`	文字列の分割
`splitlines`	文字列を改行文字の位置で分割
`replace`	文字列内の部分文字列を検索して置換
`contains`	文字列内に指定したパターンがあるかどうかの判別
`erase`	文字列内の部分文字列の削除
`extractBetween`	部分文字列の開始と終了を示す指定子の間の部分文字列を抽出
`extractAfter`	指定された位置より後の部分文字列の抽出
`extractBefore`	指定した位置より前の部分文字列を抽出
`strcmp`	文字列の比較
`regexp`	正規表現の一致 (大文字小文字を区別)

```
"Hello,world"
```

String 配列

テキストデータを効率的に操作、比較、保存します。
(各言語に対応)