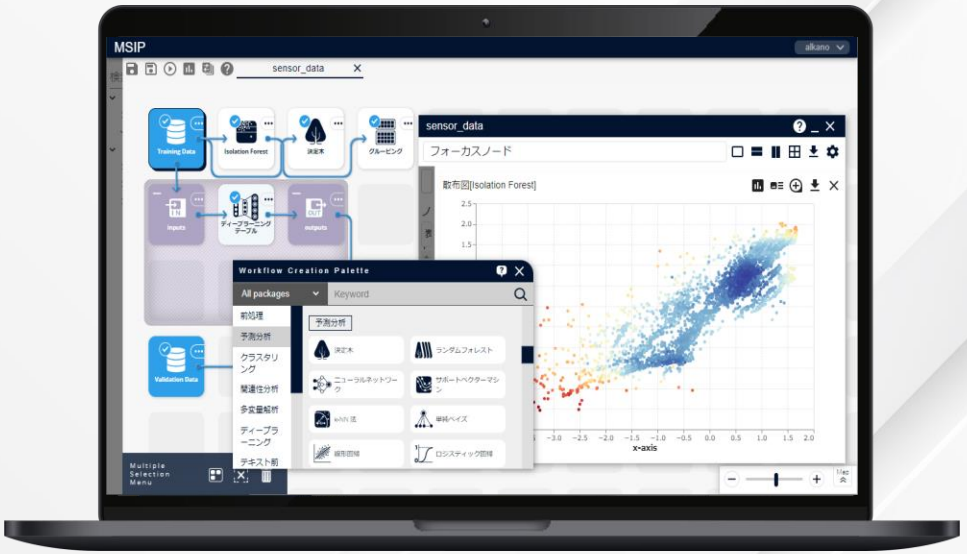
【アルカノ】
Alkano

テクニカルサンプルプロジェクト

テキストデータの可視化・類似検索



株式会社 NTTデータ数理システム

NTT DATA Mathematical Systems Inc.

このプロジェクトについて

※p.13 の「本文書・プロジェクトファイルのご利用にあたって」をお読みのうえ、ご利用ください。

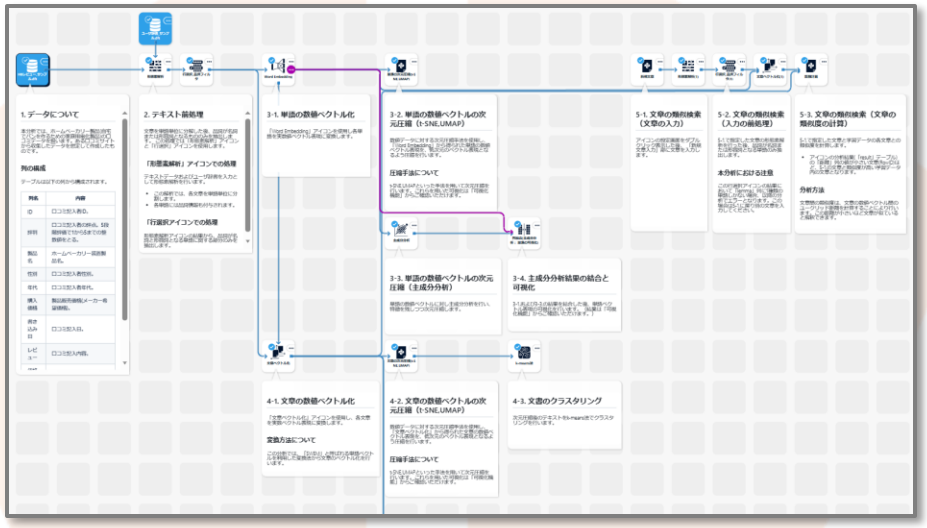
こんな方におすすめします

膨大なテキストデータを効率的に分析したい方

類似検索（特定データと似たデータの抽出）をしたい方

何をするプロジェクト？

膨大なテキストデータの分析を行う際にテキスト全てに目を通すことは、実務上・時間上難しい場合があります。
そのため、特定のデータやそれに似たデータだけを抽出し、効率的に分析を行うことが必要になってきます。
このプロジェクトでは、次元圧縮を用いたテキストデータの可視化・クラスタリングを行うことで**似た傾向のデータをグループ分け**し、新規データが既存データ群のどれに近いかの**類似検索**をベクトルの距離計算で行っています。
テキストの次元圧縮については、古典的に利用されてきた主成分分析の他、t-SNEやUMAPなどの近年注目を集めている手法を利用しています。



プロジェクト 解説

プロジェクト 解説 (1/3)

1. 対象データ

右に示すようなホームベーカリーの
レビュー（口コミ）データを利用します。
今回の分析ではテキスト部分である、
「レビュー」列のみを対象とします。

HBレビュー_サンプル.dft-data　列数: 10 行数: 100

No.	ID CATEGORY	評価 INT	製品名 CATEGORY	性別 CATEGORY	年代 CATEGORY	購入価格 INT	書き込み日 DATE	レビュー CATEGORY
1	15I6073	5	leipa ホームベーカリー	女性	40代	19,220	2015/03/21	もともと焼き立てのパンを食べたい
2	15I8963	5	leipa ホームベーカリー	女性	60代	19,220	2015/03/23	これまでずっと手ゴネでパンを作っ
3	15I8413	5	leipa ホームベーカリー	女性	40代	19,220	2015/04/05	購入したばかりでほぼ毎日パンを焼
4	15I5073	5	leipa ホームベーカリー	男性	30代	19,220	2015/04/23	全粒粉パンやメロンパンなど手軽に
5	15I8193	5	leipa ホームベーカリー	男性	40代	19,220	2015/04/26	母が欲しいというので、母の誕生日
6	15I4341	5	leipa ホームベーカリー	男性	30代	19,220	2015/04/29	毎日おいしいパンを食べられます。
7	15I8719	5	leipa ホームベーカリー	男性	30代	19,220	2015/05/10	ホームベーカリーはこれで２台目で
8	15I5763	5	leipa ホームベーカリー	男性	20代	19,220	2015/05/16	これまでのホームベーカリーが、壊
9	15I6742	4	leipa ホームベーカリー	女性	20代	19,220	2015/05/24	高さのあるぱんが焼けます。前はフ
10	15I1583	4	leipa ホームベーカリー	男性	30代	19,220	2015/05/24	コスパは最高ながら，少々音が気に
11	15I7354	2	leipa ホームベーカリー	女性	20代	19,220	2015/06/01	米粉パンが作れるというので購入。
12	15I5032	5	leipa ホームベーカリー	女性	40代	19,220	2015/04/02	パン好きの私としては、毎日焼き立

例.

もともと焼き立てのパンを食べたいときはオーブンで焼いていたのですが、オ
これまでずっと手ゴネでパンを作っていて、初めてのホームベーカリーを購入

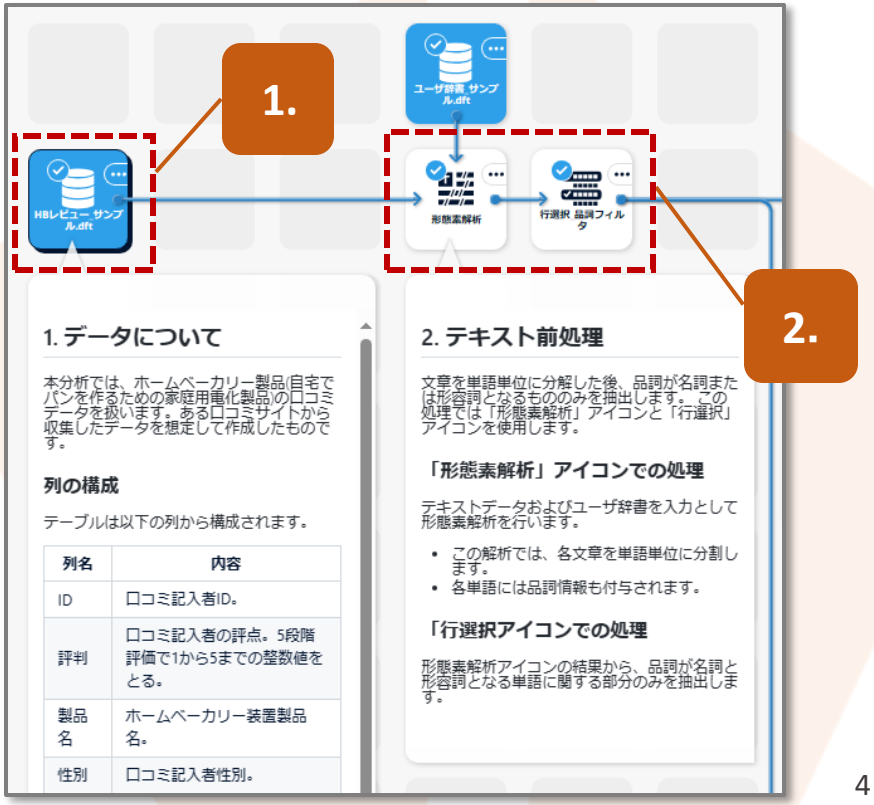
2. テキスト前処理

テキストデータを形態素解析して、単語を抽出します。名詞と形容詞に着目しフィルタリングを行うことで、文章に含まれている意味のある情報を抽出します。

行選択_品詞フィルタ

対象列

列名	列型	条件式
pos	category	in [名詞,形容詞]



プロジェクト 解説 (2/3)

3. 単語の数値ベクトル化・可視化

Word Embeddingアイコンで各単語をベクトル表現に変換します。ただし、Word Embeddingの結果そのものでは可視化を行う際に次元が多すぎるので、主成分分析やt-SNE,UMAPといった手法により次元圧縮を行い、可視化に適した次元数に減らします。

4. 文章の数値ベクトル化・可視化

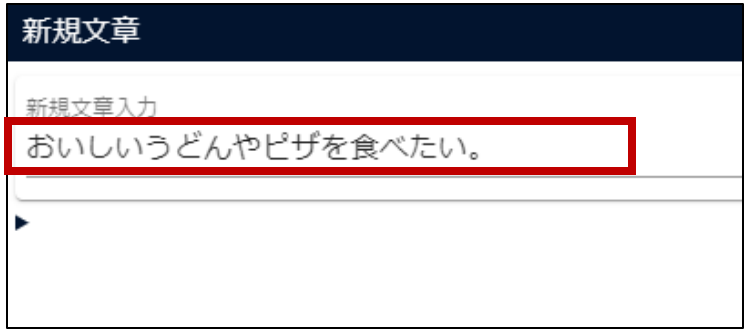
文章ベクトル化アイコンで各文章をベクトル表現に変換し、3. 単語の数値ベクトル化・可視化と同様に可視化のための次元圧縮を行います。
文章ベクトル化では単語ベクトルを利用するSWEMという手法を選択しています。

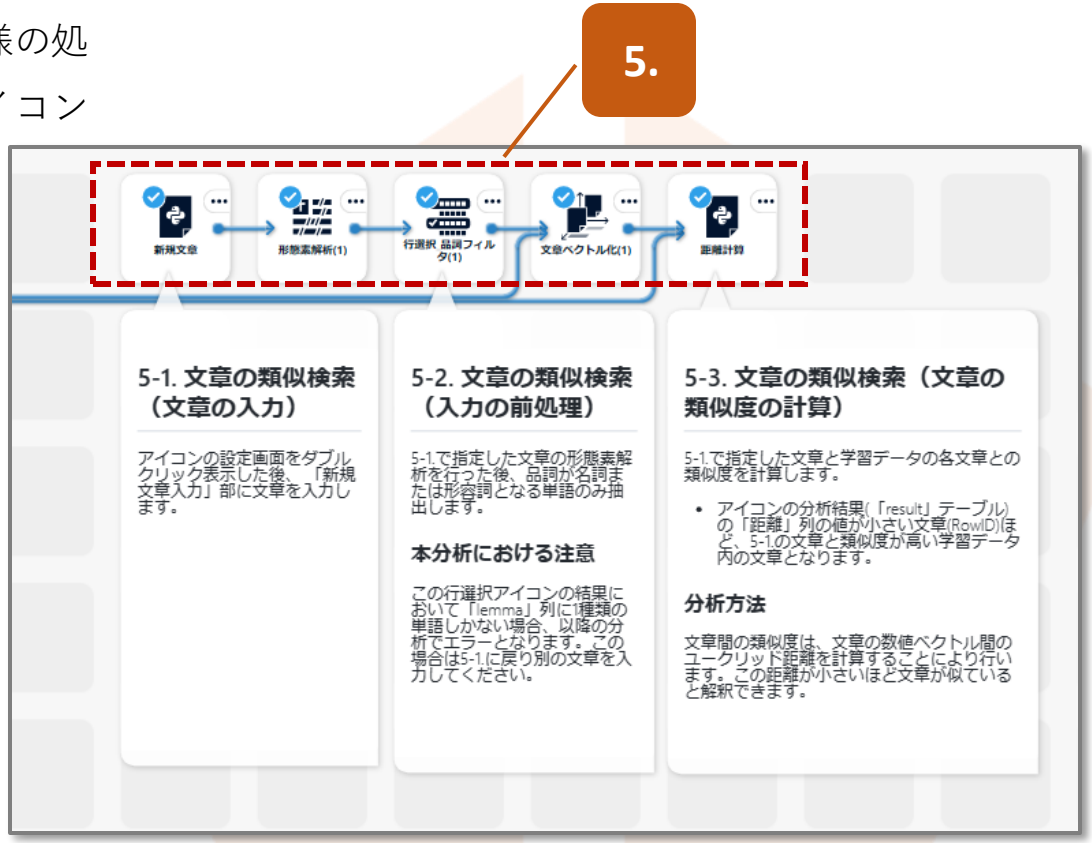
プロジェクト 解説 (3/3)

5. 文章の類似検索

文章の類似検索をベクトル間の距離計算により行います。
新規文章アイコンの新規文章入力で（既存のテキスト群と比較したい）文章を入力し、2. テキスト前処理と同様の処理を行います。ベクトル化を行った後、距離計算アイコンで類似度計算を行います。

今回は、「おいしいうどんやピザを食べたい。」という新規の文章が入力されたときに、今までの文章のうちどれが意味的に最も似ているかを計算しています。





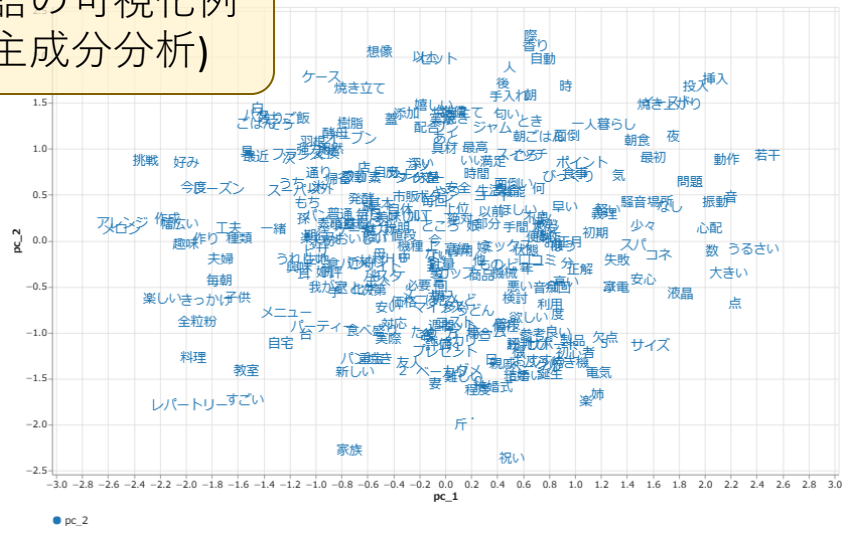
アウトプットの説明

アウトプット（可視化）

単語や文章をベクトル化した後、主成分分析やt-SNE・UMAPなどの次元圧縮法でそれらを可視化しやすい次元数に圧縮し、散布図で可視化しています。

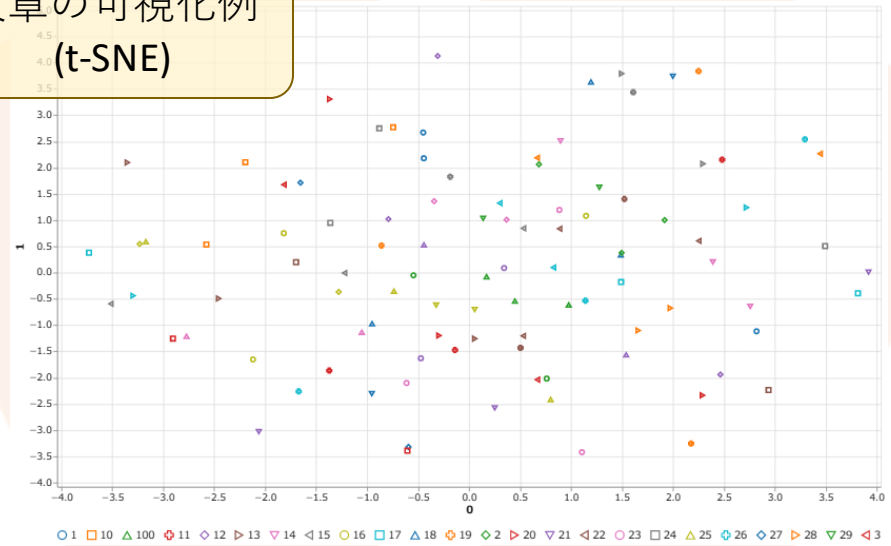
データが豊富に存在しベクトル化が上手く学習できていれば、これらの可視化では、単語の場合には似た単語同士が、文章の場合には似た文章同士が近くにプロットされます。

単語の可視化例
(主成分分析)



可視化画面：グラフ一覧 -
[テキスト散布図[列結合(主成分分析：単語の可視化)-resultx(pc_1)y(pc_2)]

文章の可視化例
(t-SNE)



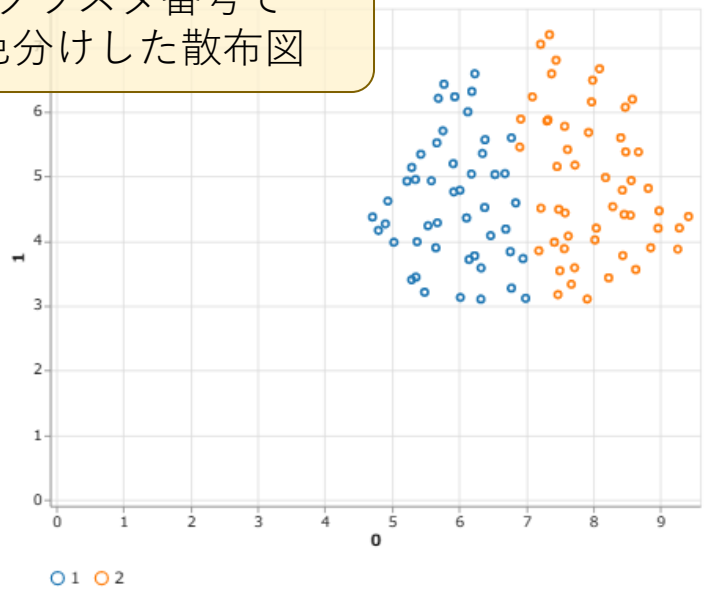
可視化画面：グラフ一覧 - 散布図[文章の次元圧縮(t-SNE,UMAP)-tsne_docx(0)y(1)]

アウトプット（クラスタリング）

クラスタリングの一種である「k-means法」アイコンでは、次元圧縮後のデータをグループ分けしています。
データのクラスタリングをどの段階で行うか、実際の分析では試行錯誤が必要ですが、今回のプロジェクトでは「可視化のために次元圧縮したデータは、人の解釈が行いやすい状態になっており、クラスタリングの実行に適した状態になっているはずだ」という予想のもと実行しています。

クラスタリングによって、データ傾向の解釈を行ったり、データにラベルを付与して、予測モデル構築のためのデータ整備に活用したりすることができます。
例えば、クラスタ番号を説明変数の一つとして元データに追加し、この文章を書くような人は○○だ、といった結果を導くこともできます。

クラスタ番号で
色分けした散布図



可視化画面：グラフ一覧 - 散布図[k-means法-resultx(0)y(1)]

k-means法-result　列数: 4 行数: 100

No.	cluster_id Category	RowID Integer	0 Float	1 Float
1	1	1	6.235187	6.588500
2	1	2	5.375085	3.990037
3	2	3	8.460635	4.413394
4	1	4	6.989886	3.115825
5	2	5	9.418335	4.379376
6	1	6	6.226283	5.431502
7	1	7	6.129481	6.000703
8	2	8	8.022105	4.018936
9	2	9	7.445384	6.797503
10	2	10	7.378325	6.588400

グループ分けの番号
（クラスタ番号）

入力したデータ

可視化画面：k-means法 ノード - resultテーブル

アウトプット（距離計算）

「距離計算」アイコンの結果は、「新規文章」アイコンで入力した文章と、既存データである「HBレビュー_サンプル」データの近さ（類似度）を表しています。
距離列を値が小さい順に昇順ソートすることで、新規文章と距離の近い（類似度の高い）順にデータを並び替えることができます。

この例では、RowID:19に対応する文書が、新規文章に対して一番類似度が高い文書です。

この考え方を用いて、実際の業務では以下のような活用方法が考えられます。

- 類似文書検索（欲しい文書と似た文書群の抽出）
  - 例：調査文献の絞り込み・スクリーニング
- 文書推薦（入力文書と似た文書を推薦）
  - 例：FAQ検索

(RowID:19 の文章)
パンに加え、ピザ生地やケーキも焼けるということでこの機種にしました。レシピ通りにケーキを焼いてみたところ、しっとりおいしいケーキができました。ピザ生地ももっちりしていておいしいですね。もう宅配ピザは頼まなくてもいいかもしれません。料理のレパートリーが増えますので、すごくお勧めです。

距離計算-result　列数: 2 行数: 100

No.	RowID Integer	距離 Float
1	19	1.654161
2	75	1.933700
3	6	2.037984
4	72	2.108748
5	31	2.112271
6	57	2.128740
7	61	2.140654
8	8	2.141308
9	23	2.159706
10	83	2.172319

可視化画面：距離計算 ノード - result テーブル

補足情報

技術的な情報や利用規約について

技術情報：類似度について

今回の分析では、類似度をユークリッド距離で定義しています。

d次元ベクトルx,yに対するユークリッド距離は以下の形で計算できます。

$$similarity(x,y) = euclidean(x,y) = \sqrt{\sum_{i=1}^{d}(x_i - y_i)^2}$$

この他、類似度としてよく使われるものにコサイン類似度があります。

d次元ベクトルx,yに対するコサイン類似度は以下の形で計算できます。

$$similarity(x,y) = cosine(x,y) = \frac{\sum_{i=1}^{d} x_i y_i}{\sqrt{\sum_{i=1}^{d} x_i^2}\sqrt{\sum_{i=1}^{d} y_i^2}}$$

ユークリッド距離は2点間の距離であり、コサイン類似度は原点から見た2点のなす角度です。

BoW(Bag-of-Words)によるベクトル化など、値のスケールが一定ではない場合には、

コサイン類似度を使う場合もあります。

本文書・プロジェクトファイルのご利用にあたって

本文書ならびにプロジェクトファイルは、（株）NTT データ数理システム （以下「弊社」）が開発・販売する分析プラットフォーム Alkano についての情報提供として弊社が作成を行ったものです。弊社による事前の許可なしに、本文書の再配布や引用の範囲を超える複製といった行為、およびリバースエンジニアリングを禁じます。

本文書ならびにプロジェクトファイルのご利用に際して、ご利用者様および第三者に損害が発生したとしても、弊社は責任を負わないものとします。

プロジェクトファイルは、その中に同梱されているデータを利用し、本文書内で解説している設定可能なパラメータで動作させた場合についてのみ、弊社にて動作の検証を行っております。これを超えるような状況における動作は保証いたしません。

本プロジェクトファイル は、MSIP1.8.2 および Alkano1.2.2 にて動作確認を行っております。

【アルカノ】

Alkano

データ活用の確かなパートナー

お問い合わせ: 株式会社NTTデータ数理システム　営業部
Tel: 03-3358-6681
E-mail: alkano-info@ml.msi.co.jp
WEB: https://www.msi.co.jp/alkano/

株式会社 NTTデータ数理システム

NTT DATA NTT DATA Mathematical Systems Inc.