PythonとKerasによるディープラーニング

6章 テキストとシーケンスのためのディープラーニング

6.1 テキストデータの操作

17T4028N 菊田尚樹

テキストのベクトル化

テキストはそのまま用いるのではなく数値テンソルに変換する。テキストの数値テンソルへの変換プロセスをベクトル化と呼ぶ。

◇ベクトル化の複数の手法

・テキストを単語に分割し、各単語をベクトルに変換

・テキストを文字に分割し、各文字をベクトルに変換

・Nグラムの単語または文字を抽出し、各Nグラムをベクトルに変換

(Nグラム：複数の連続する単語または文字のグループ)

テキストのベクトル化

テキストの分割に用いる単位(単語、文字、Nグラム)のことを<u>トークン</u>と呼ぶ。

テキストをベクトル化するプロセスはすべて何らかのトークン化手法を適用し、生成されたトークンを数値ベクトルに関連付けるという手順で構成される。

テキストのベクトル化

トークンを数値ベクトルに関連付ける方法はいくつかある。

・one-hotエンコーディング

・トークン埋め込み(単語埋め込み)

one-hotエンコーディング

one-hot表現：一つが１でその他が0のベクトル

例　(0,0,1,0,0,0)

「私はカレーが好き」

私：　(1,0,0,0,0,0,0,0)

は：　(0,1,0,0,0,0,0,0)

カレー：(0,0,1,0,0,0,0,0)

が：　(0,0,0,1,0,0,0,0)

好き：　(0,0,0,0,1,0,0,0)

単語数に応じてベクトルの次元が多くなってくる

one-hotエンコーディング

one-hot ハッシュトリック

トークンの数が膨大の場合に使う手法

メリット：ベクトル長を固定でき、メモリの節約になる

デメリット：ハッシュ衝突の可能性あり

1．単語をハッシュ関数にかけ整数を作る

2．ベクトル(配列)の添え字に上記の整数を利用する

◇ベクトル長を1,000までに制限する場合

index = abs(hash(word)) % 1000

results[i, j, index] = 1　i番目の文書 j番目の単語

単語数が1000以上のときはハッシュ衝突の可能性が高い

トークン埋め込み(単語埋め込み)

one-hotエンコーディングとの比較

one-hot　:

二値(0,1)

高次元

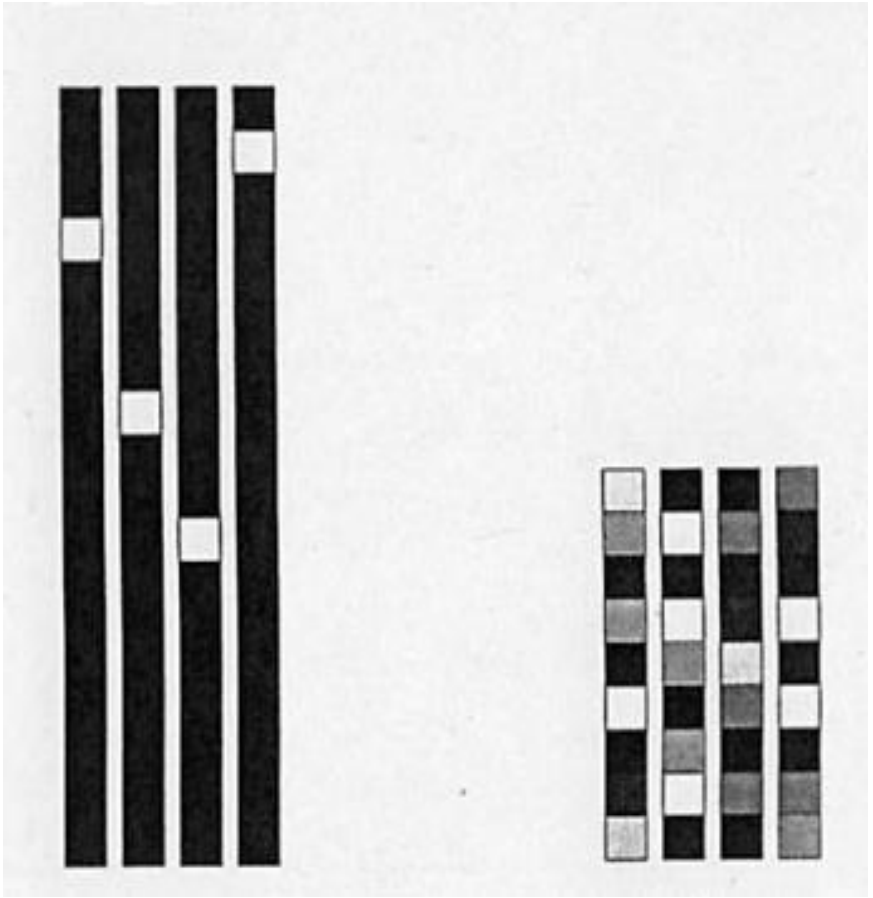
疎ベクトル

単語埋め込み　:

浮動小数点型

低次元

密ベクトル



トークン埋め込み(単語埋め込み)

単語埋め込みとは？

自然言語を幾何学的な空間へマッピングするもの

ベクトル同士の距離を計算することで類似度・関連性がわかるようにする

代表的なツール

word2vec

GloVe