テキストデータのベクトル化

江崎　菜々

富山県立大学　情報基盤工学講座

1. はじめに
2. ベクトル化の手法の種類
3. 他「メリットなど」

October 11, 2019

はじめに

莫大に増大し続けているネット内のテキストをベクトル化するべく、ベクトル化の手法を調べてみる。
今回は用途に応じて４つの手法について紹介する。

調べたベクトル化の手法

1、back of Word (BOW)
2、Keras
3、Word2vec
4、Skip-thought

調べた手法その 1　BOW

手法の中でも単純で、文章から単語が何回出てきたかカウントするのみ。
-例題文-
The sun is shining
The weather is sweet
The sun is shining, the weather is sweet, and one and is two

BoW は大文字小文字の区別はしないので「the」と「The」は同じ単語扱いになる。
そうすると、単語数は 9 個、文章は 3 個になる。
文字カウントには scikit-learn という機械学習のライブラリの CountVectorizer を使う。

調べた手法その 1　BOW

まずアルファベット順に単語ごとに番号を割り振る。
出力結果
'is': 1, 'one': 2, 'sweet': 5, 'two': 7, 'shining': 3, 'sun': 4, 'the': 6, 'weather': 8, 'and': 0
つづいて各行に 9 個の単語が何個あるか調べる
出力結果
010110100　　1 行目
and が 0 個、is が 1 個、one が 0 個、shining が 1 個、sun が 1 個、sweet が 0 個、the が 1 個、two が 0 個、weather が 0 個
010100101　　2 行目
232111211　　3 行目

調べた手法その２　Keras

keras は自らかくソースコードが他のデープラーニングよりも簡単にかけることから初心者むけと言われている。他のクラスを使えば手書き文字の識別も可能らしい。
テキストのベクトル化は Keras の Tokenizer クラスでやる。3 つの方法で得られる情報が違う

-例題文-
I am a student. He is a student, too.
She is not a student

調べた手法その２　Keras

まず BoW 同様文章の数、単語ごとの出現回数、単語に割り当てられたばんごうを出力する。
与えられた文章の数　：　2
与えられた文章内の単語ごとの出現回数　　：'i', 1, 'am', 1, 'a', 3, 'student', 3, 'he', 1, 'is', 2, 'too', 1, 'she', 1 'not', 1
単語ごとに割り振られた番号　：　'a': 1, 'student': 2, 'is': 3, 'i': 4, 'am': 5, 'he': 6, 'too': 7, 'she': 8, 'not': 9

調べた手法その２　Keras

○バイナリ表現

　回数問わず文章中に出現したら１になる。先頭は０番目で、０番目に割り振られた番号はないからいつだって０

0. 1. 1. 1. 1. 1. 1. 1. 0. 0.　　１行目

0. 1. 1. 1. 0. 0. 0. 0. 1. 1.　　２行目

○カウント表現

各単語を番号順に並べ、各行で出現した回数を表示する。これも先頭は０

0. 2. 2. 1. 1. 1. 1. 1. 0. 0.　　１行目

0. 1. 1. 1. 0. 0. 0. 0. 1. 1.　　２行目

調べた手法その２　Kares

○ TF-IDF 表現
単語の重要度を表す。出現頻度が高い　＋　いくつもの文章で出現しない単語ほど数値は高くなる。
この文章で一番高いのは「a」次は「student」
1 行目
0. 0.86490296 0.86490296 0.51082562 0.69314718 0.69314718 0.69314718 0.69314718 0. 0.
2 行目
0. 0.51082562 0.51082562 0.51082562 0. 0. 0. 0. 0.69314718 0.69314718

調べた手法その３　Word 2 vec

言葉の引き算により関係性を求めることが可能
具体的には単語の意味を加えたり、抜いたりできる
単語同士の意味の近さをベクトル化により数値化できる。
⇒進化するニュートラルネットワークの中で自然言語を演算処理できる。
例：国王ー男＋女＝女王
例：ゴリラ≒マウンテンゴリラ←勝手な予想

例題：
The sun is shining

この文章のみで考えると、各単語のベクトル表記は
the=1000, sun=0100, is=0010, shining=0001 、と文章に出てきた順に番号がつく

調べた手法その４　Skip-thought

word2vec と特徴は似ているが、Skip-thought は長文に対して、類似文との比較ができるということである。

Word ２ vec の課題点、「文章中の、ある単語の前後５単語に対して関連性を覚える。それにより離れていると関連性がつかみにくなる。」をクリアする
例：「星とは宇宙の中でまさに LED のように光り輝くとても大きな岩である。 」　　の「星」と「岩」

○ Woed2vec との違い。
Word2vec ⇒入力に対し、周辺に位置する単語を予測して単語の共起関係を学ぶ。
Skip-thoug ⇒入力単語の系列をエンコード、前後の文の単語を出力として順番に予測。エンコード結果をベクトル化する。

使用例：類似した文の検索、長文もいける。長文同士の比べることも可能

その４

具体例
「研究会もいいですけど、研究もしたらどうですか。」と比較

思想的に世の中を変えたと思っているじゃないですか⇒類似度 0.803
本当に批判するんだったら、ぐうの音も出ない批判をすればいいじゃないですか。類似度⇒ 0.781
類似度は MAX 1

おわりに

まとめ

1. いろいろあった。
   手法でできることが違うが word2vec のような演算はコードが多すぎるし、Skip-thought は数値的に類似でも実際は結構違っていたりしていた。

今後の課題

1. 簡単なやつ Keras とかのコードが解読できるようになること