知識処理論 第二回 レポート課題

工学部システム創成学科 C コース B3 合田拓矢

Introduction

自然言語処理の問題を解決する上でボトルネックの 1 つに, タスクごとにアノテーション付きデータを用意する必要があることが挙げられる. また, せっかくラベル付きデータを用意しても, タスクの問題設定が変わると必要なアノテーションが異なることも多く, 教師付き学習を行うコストは高い. 例えば, 文章の分類問題を考えると, 予め用意していたカテゴリ以外のカテゴリを追加するたびに正解データを作成する必要がある.

このレポートでは, 分布仮説に基づいて単語の潜在的な意味を捉えたベクトル表現を学習する word2vec を利用して, 教師付きデータを用意せずに文書分類を行うことができないか考える.

Methods

文書分類で予めカテゴリが設定されていないような場合に, 文書を分類する方法を考える. より具体的には, 「[文章]は〇〇か否か」の 2 値分類を行うことを考える. 〇〇には辞書に含まれる任意の単語が入る(例: いい/悪い, 優しい/厳しい など)
Wikipedia のダンプデータを用いて学習した word2vec モデルを利用して, 各文章の「〇〇度合」を以下のような手法で数値化する.

1. 文章を MeCab で形態素解析し, 名詞・動詞・形容詞・形容動詞のみを抽出する.
2. 文章中の各単語に対して, そのベクトル表現と〇〇のベクトル表現との間の cos 類似度を取る.
3. その平均を文章の「〇〇度合」と定義する.
4. 〇〇度合から閾値を定めて分類する.

データセットは livedoor ニュース(https://www.rondhuit.com/download.html#ldcc) を利用した. また word2vec のモデルはこちらの学習済みモデルを使用した.
(http://aial.shiroyagi.co.jp/2017/02/japanese-word2vec-model-builder/)
今回書いたコードはこちらにおいてある.
https://github.com/tkyaaida/knowledge_processing/blob/master/report2/report2.ipynb

Results

各文章の「怒り度合」を計算し, 上位 10 位を抜粋すると以下のような結果を得た. (ただし, sentence は元の文章, token_seq は抽出された形態素のリスト, score は上記のスコアである)

```
[{'sentence': '本当に驚きました。', 'token_seq': ['驚き'], 'score': 0.773547554150178},
 {'sentence': 'でも不安だ。', 'token_seq': ['不安'], 'score': 0.6254362622572959},
 {'sentence': 'と大変お怒りの様子。',
  'token_seq': ['大変', '怒り', '様子'],
  'score': 0.6157276879175946},
 {'sentence': 'たまらないね。', 'token_seq': ['たまらない'], 'score': 0.5464817954724311},
 {'sentence': 'すごく、謝りました。',
  'token_seq': ['すごく', '謝り'],
  'score': 0.5382587800835938},
 {'sentence': '恥を知ってほしいね。',
  'token_seq': ['恥', '知っ', 'ほしい'],
  'score': 0.5379403460838145},
 {'sentence': '不安だらけだ。',
  'token_seq': ['不安', 'だらけ'],
  'score': 0.5346223609270333},
 {'sentence': '正直に、はっきりと聞きます。',
  'token_seq': ['正直', '聞き'],
  'score': 0.531903227676507},
 {'sentence': 'その顛末はいかに―。',
  'token_seq': ['顛末', '―。'],
  'score': 0.5109663268328465},
 {'sentence': '私もごめんなさいって言いますし。',
  'token_seq': ['私', '言い'],
  'score': 0.5069577043037405}]
```

この手法では, 対象の単語に対して, 余計な単語が含まれていない方が高スコアとなるため, 短い語数の文章が上位に来る傾向がある. そこで, 5 語以内の文章は除外するようにしたところ以下のような結果を得た.

```
[{'sentence': '思ったこと『あのブスめ！』と、怒り心頭だ。',
  'token_seq': ['思っ', 'こと', 'ブス', 'め', '怒り', '心頭'],
  'score': 0.4767247590137995},
 {'sentence': '菜々緒が杉村太蔵のセクハラ発言に嫌悪の表情。',
  'token_seq': ['菜', '緒', '杉村', '太蔵', 'セクハラ', '発言', '嫌悪', '表情'],
  'score': 0.4156870290483231},
 {'sentence': 'のメンバーは笑いつつも「ひどい」「びっくり」とさんまの発言に異議を唱えた。',
  'token_seq': ['メンバー', '笑い', 'ひどい', 'びっくり', 'さんま', '発言', '異議', '唱え'],
  'score': 0.41488492371010827},
 {'sentence': 'まあ、ファンの方に迷惑をかけてしまったので、心配かけてしまったので、申し訳ないなって。',
  'token_seq': ['ファン', '方', '迷惑', 'かけ', 'しまっ', '心配', 'かけ', 'しまっ', '申し訳'],
  'score': 0.41136217118679524},
 {'sentence': '私たちはGhaddafi（カダフィ）のことを忘れない、安らかに。',
  'token_seq': ['私', 'たち', 'Ghaddafi', 'カダフィ', 'こと', '忘れ', '安らか'],
  'score': 0.408436657026593},
 {'sentence': 'すると、店屋のおじさんが、「おい、ガキ、ちょっと待てや。」とかなり乱暴な口調で私を引き留めました。',
  'token_seq': ['店屋', 'おじさん', 'ガキ', '待て', '乱暴', '口調', '私', '引き留め'],
  'score': 0.39957401628787415},
 {'sentence': 'マツコは即座に「今のセリフちょうだい!」と褒めるくらい、この言葉を気に入ったという。',
  'token_seq': ['マツコ',
   '即座',
   '今',
   'セリフ',
   'ちょうだい',
   '!」',
   '褒める',
   '言葉',
   '気に入っ',
   'いう'],
  'score': 0.3953983121067167},
 {'sentence': 'そういう人が、いつか反省してくれるとありがたいですなぁ。',
  'token_seq': ['人', 'いつか', '反省', 'し', 'くれる', 'ありがたい'],
  'score': 0.3787018449096587},
 {'sentence': 'デヴィ夫人のブログに共感のコメントが殺到。',
  'token_seq': ['デヴィ', '夫人', 'ブログ', '共感', 'コメント', '殺到'],
  'score': 0.37548823355629496},
 {'sentence': '次に気を取り直した有吉が、改めて夏目に「自分とマツコ、どっちがいじりにくい？」と質問を投げたが、夏目は「有吉さん」と返答。',
  'token_seq': ['気',
   '取り直し',
   '有吉',
   '夏目',
   '自分',
   'マツコ',
   'どっち',
   'いじり',
   'にくい',
   '質問',
   '投げ',
   '夏目',
   '有吉',
   'さん',
   '返答'],
  'score': 0.37478133783503875}]
```

一部,「怒り」とは関係ない文章も含まれるが, 全体的には怒りっぽい内容の文章が上位にランクインしている. 特に, 2 番目の文章は「怒り」という文字が文章中に含まれていないにもかかわらず,「嫌悪」という類似語から怒り度合が高いと判定できている. 同様に 6 番目の『すると、店屋のおじさんが、「おい、ガキ、ちょっと待てや。」とかなり乱暴な口調で私を引き留めました。』という文章に対しても, 怒り度合が高いと判定できている.

Discussions

提案手法を用いることで, 完全な教師なしでクエリの単語に対してその度合が高い文章を抽出することが可能となる. 今回使用した word2vec モデルは誰でも取得可能な Wikipedia データから作成したものであるため, 分析対象のデータさえあれば, この手法を適用することが可能である.

しかし, いくつかの欠点もある. 今回の実験では「怒り」以外のいくつかの単語(例: かっこいい, 嬉しい, etc)に対しても試してみたが,「怒り」ほどうまくいかなかった. 今回用いたデータセットがニュース記事であるため, 否定的な内容が多かったためだと考えられる.

また, 根本的な課題として, word2vec モデルでは単語が空間上の 1 点 で表現されているため, クエリの単語が指し示す概念全体を捉えきれていない可能性があることが挙げられる. 文章中の各単語とクエリ単語の距離の平均が, 文章の単語系列とクエリ単語が意味する概念全体との類似度を捉えられているかは疑問が残る. 人間は言語を用いることで概念を伝達可能な形式に落とし込んでいると考えると, この手法を発展させるためには何らかの方法で概念をモデルに組み込む必要があると考える.