概要

1. NEologd に含まれていない単語の収集

2. NEologd に含まれる単語のベクトルの学習

Github: https://github.com/reiyw/intern-line

1. NEologd に含まれていない単語の収集

- 収集された単語例:
  - 言語批判論集　ゲンゴヒハンロンシュウ
  - 言語技術　　ゲンゴギジュツ
  - 言語の数理　ゲンゴノスウリ
- リソース: 日本語版 Wikipedia ダンプデータ (20160601)
  - https://dumps.wikimedia.org/jawiki/20160601/
  - jawiki-20160601-pages-articles.xml.bz2 2.2 GB
- 方針:
  - できるだけ固有表現のみを抽出
    - 特に，NEologd には作品名が足りてなさそう
  - 正規表現だけでラクに集めてくる
    - 数が必要である場合に API は叩きたくない

手順

1. WikiExtractor (https://github.com/attardi/wikiextractor) で
   ダンプされた xml をプレーンテキスト化
   - マークアップの削除など
2. 括弧 “『』” で囲まれている単語を抽出
   - 『』は作品名，書籍名を表すために用いられる
3. 英数字だけからなる文字列を削除
   - 日本語形態素解析のための辞書なので，英単語は念のため除外
4. 文字数が 2 以下または 31 以上である文字列を削除
   - 2 以下: 作品名である場合もあるが，ほとんどはノイズ
   - 31 以上: 『』が引用文のために用いられる場合がある
5. 半角の記号類を含む文字列を削除
6. NEologd との重複単語削除
   - LevelDB 使用
7. NEologd を用いた MeCab で読みの付与

作成したプログラム・データ

- extract-name-of-work.sh: 前頁の 2-5 を処理
- new_words.txt.bz2: 収集した単語と読み (174,013 単語)

2. NEologd に含まれる単語ベクトルの学習 | 手順

1. WikiExtractor (https://github.com/attardi/wikiextractor) で ダンプされた xml をプレーンテキスト化
2. 1 行 1 文に変換
3. 行をランダムに並び替える
4. NEologd を用いた MeCab で形態素解析
5. 学習したい単語の基本形を抽出
   - 名詞・動詞・形容詞だけを残す
   - ただし，非自立な動詞・名詞，ストップワードは削除する
6. word2vec (https://github.com/dav/word2vec) で学習
   - 300 次元とした以外はデフォルトのパラメタを使用
7. NEologd との重複単語削除

作成したプログラム・データ

- make_corpus.sh: 前頁の 2-3 を処理
- mecab2words.py: 前頁の 5 を処理
- vec.txt.a{a,b,c,d,e}.bz2: 学習された単語ベクトルの内，NEologd に存在するもの (535,173 単語)

自己 PR

- 現在私が取り組んでいる研究についてはエントリーシートに記載した通りです．最近人工知能学会で発表をしましたので，参考までにそのときのスライドを同レポジトリにアップロードしておきます(jsai2016.pptx)．現状では，国際会議に投稿するために，モデルの改善などに取り組んでいるところです．

- OSS に関する活動は，主に自信の無さが原因で，残念ながらこれまでにしたことがありません．必要だと感じたツールを自分で作ることはあまり厭わないし，それをするだけの能力はそれなりにあると自負しているのですが，外部に発信していく，あるいは誰かのために貢献するというところに気持ちが向かわないのだと思います．そこで貴社へのインターンで実際のプロの開発現場に飛び込んで様々な刺激を受けてみて，良い方向に向かうか悪い方向に向かうか分かりませんが，「狭い世界に閉じこもって研究のためのコードを書くだけ」な現状を変えるきっかけが得られたらと考えています．