NLP 課題: Dependency Parsing

廣瀬　惟歩

1　はじめに

自然言語処理タスクの一つに，Dependency Parsing (係り受け解析) というものが存在する．これは構文解析の一種であり，各単語の依存関係に着目して文章の構造を解析するタスクである．

本課題では，特徴量を組み込んだ Shift Reduce を利用して各単語の依存関係を学習，推定し，実験を通して精度を確認した．

2　手法

本課題で用いる手法は Graham Neubig 氏の NLP チュートリアル[*1]を基にした．本課題では以下の関数を要するプログラムを構築した:

■def train_parse(train_file): train データに対して依存関係の学習を行うための関数．この時後述の def shift_reduce に deque(save_list) を入力として 2 つ渡しているが，前者は学習中に予測した依存関係を記録するためのもの，後者は正解の依存関係を記録するためのものである．

■def shift_reduce(queue, queue_for_corr, mode="train"): 与えられた各文章に対し，特徴量を基に重みを学習，更新するための関数．特徴量の計算には後の def makefeats を活用する．予測と正解が違った場合，予測に対応する重みには負の，正解に対応する重みには正の値を加算する．

■def makefeats(stack, queue): 特徴量を計算し，値を返すための関数.

■def unproc_word(queue): 各単語の未処理の子供の数を把握するための関数．正解の依存関係を記録するのに用いる．

■def test_parse(test_file): 学習によって得られた重みパラメータを用いて，test データに対して依存関係の予測を行うための関数.

■def accuracy(heads, heads_answ): テスト結果の精度 (accuracy) を計算するための関数.

[*1] https://github.com/neubig/nlptutorial/blob/master/download/11-depend/nlp-programming-ja-11-depend.pdf

3 実験

3.1 実験データ

train 用データには mstparser-en-train.dep[*2]を，test 用データには mstparser-en-test.dep[*3]を使用した．これらのデータでは各文章に対し，単語ごとに ID，単語の表層形，品詞ラベル，係り受け先などが 1 行ずつ記載されている (例: 1 in in IN IN _ 43 PP)．そのため，def train_parse と def test_parse では各行を split で分割し，ID，単語の表層形，品詞ラベル，係り受け先を取り出して学習，推測するようにした．

3.2 実験方法

train データと test データを用意したのち，コマンドファイルから python parse.py を実行して学習とテストを行った．精度の確認方法は以下の通りである．

1. def test_parse において各行から ID，単語の表層形，品詞ラベル，係り受け先を取り出し，それらを shift_reduce の入力として，返された推測と正解の依存関係をそれぞれ heads_answ と heads_pred リストに追加する．
2. heads_answ リストと heads_pred リストを def accuracy の入力とし，2 つのリストの係り受け先の一致数を全体で割ったものを精度とする．

4 結果

実験によって得られた精度は 0.07 であった．これは明らかに低い数値であり，特徴量の組み込み方法及び重みパラメータの更新方法が不適切であったためと考えられる．

[*2] https://github.com/neubig/nlptutorial/blob/master/data/mstparser-en-train.dep
[*3] https://github.com/neubig/nlptutorial/blob/master/data/mstparser-en-test.dep