Labo 2 - Mise en œuvre et évaluation de «POS taggers» pour le français

Nathan Gonzalez et Vincent Guidoux

1.Télécharger des données en français annotées avec les POS tags : trois fichiers de données

Quel est le format de ces 3 fichiers ?

C'est le format CoNLL-U Des lignes de mots séparés en dix colonne par un charactère de tabulation

https://universaldependencies.org/format.html (https://universaldependencies.org/format.html)

Dans quelles colonnes se trouvent les mots et leur POS tags ?

- Colonne 0 : le mot
- Colonne 2 : le POS tags

Pouvez-vous trouver sur le Web la liste des POS tags du projet Universal Dependencies ?

https://universaldependencies.org/u/pos/ (https://universaldependencies.org/u/pos/)

2.Évaluer le Stanford POS tagger pour le français avec les modèles fournis

b.Téléchargez les modèles pour le francais et testez-les sur les fichiers « dev » et « test », en utilisant le modèle `french-ud.tagger`. Quels sont les scores obtenus ?

`french-ud.tagger.props` composé respectivement de

```
model = models/french-ud.tagger
testFile = format=TSV,wordColumn=1,tagColumn=3,../data/fr-ud-dev.conllu3
```

et

```
model = models/french-ud.tagger
testFile = format=TSV,wordColumn=1,tagColumn=3,../data/fr-ud-test.conllu3
```

en faisant la commande :

```
java -mx1000m -classpath stanford-postagger.jar
edu.stanford.nlp.tagger.maxent.MaxentTagger -p
rop models/french-ud.tagger.props
```

fr-ud-test.conllu3 nous donne : Unkown words right : 487 *(69.87%)*

fr-ud-dev.conllu3 nous donne : Unkown words right : 2232 *(73.20%)*

3.Entraîner le Stanford POS tagger sur les données UD en français

c.Quel modèle est meilleur, le vôtre ou celui téléchargé en 2 ?

`french-ud.tagger.props` composé de

```
model = models/french-ud.tagger
trainFile = format=TSV,wordColumn=1,tagColumn=3,../data/fr-ud-train.conllu3
```

en l'entraînant avec la commande :

```
java -mx1000m -classpath stanford-postagger.jar
edu.stanford.nlp.tagger.maxent.MaxentTagger -p
rop models/french-ud.tagger.props
```

et en faisant la même manipulation qu'au point `2b`

fr-ud-test.conllu3 nous donne : Unkown words right : 489 *(81.364%)*

fr-ud-dev.conllu3 nous donne : Unkown words right : 2243 *(83.04%)*

Notre modèle entraîné est plus efficace

4. Entraîner un POS tagger dans NLTK pour qu'il fonctionne sur le français.

a.Importez les données en francais dans NLTK

In [1]:

```
from nltk.corpus.reader.conll import ConllCorpusReader
from nltk.tag.perceptron import PerceptronTagger

#source : http://techstuffbrazil.blogspot.com/2017/03/quick-tutorial-to-nltk-corpus-reader-of.html

root = './data/'
test = 'fr-ud-test.conllu3'
train = 'fr-ud-train.conllu3'
COLUMN_TYPES = ('ignore',
                'words',
                'ignore',
                'pos',
                'ignore',
                'ignore',
                'ignore',
                'ignore',
                'ignore',
                'ignore')

trainFile  = ConllCorpusReader(root=root,
                               fileids=train,
                               columntypes=COLUMN_TYPES,
                               encoding='utf8',
                               separator="\t",
                               tagset='universal')

testFile  = ConllCorpusReader(root=root,
                               fileids=test,
                               columntypes=COLUMN_TYPES,
                               encoding='utf8',
                               separator="\t",
                               tagset='universal')

train_words = trainFile.tagged_sents()
test_words = testFile.tagged_sents()
```

c1.Entraînez ce module sur les données train

In [2]:

```
tagger = PerceptronTagger(load=False)

tagger.train(train_words)
```

c2.puis testez-le sur les données test grâce à la méthode evaluate

In [3]:

```
tagger.evaluate(test_words)
```

Out[3]:

```
0.9596038065643814
```

Comment se compare-t-il avec les deux modèles du POS tagger MaxEnt de Stanford ?

Il est beaucoup plus performant que les deux modèles des points 2 et 3