Sequence Labeling using Perceptron with viterbi decoding

Description

This project is about implementing an algorithm for training a tagging model. This model can be used in many information extraction (IE) or sequence labeling tasks, i.e. POS tagging, named-entity recognition, etc...

For example, let assume that our task is POS tagging. Given in input the sentence:

the man saw the dog

We would like the tagger to return the following output:

the/**D** man/**N** saw/**V** the/**D** dog/**N**

This model uses the perceptron algorithm with viterbi decoding to compute the probabilities of all the possible tag assignments to words in a sequence. Then, it returns the sequence with the maximum probability.

Your task is to implement the model described in the paper above and test the

The paper describing the algorithm can be downloaded here:
Discriminative Training Methods for Hidden Markov Models:Theory and Experiments with Perceptron Algorithms

Data

The data for training and testing the system is the CoNLL-2000 dataset for the text chunking task. More information about the data and task can be found here:
http://www.cnts.ua.ac.be/conll2000/chunking/

Evaluation

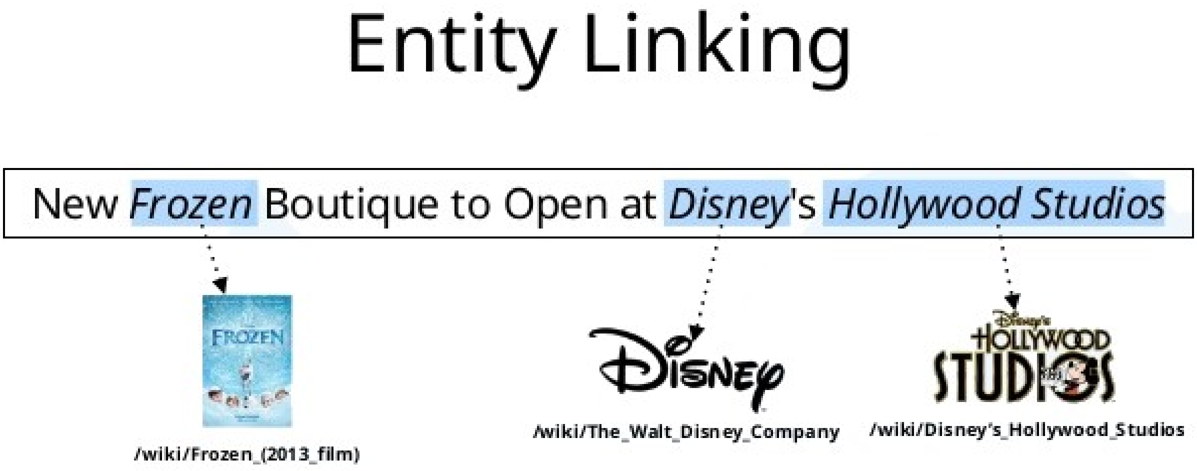
The system will be evaluated according to the standard metrics known in literature such as precision, recall and F-measure.

Entity Linking on Twitter data

Description

The aim of entity linking is to build a system that identifies mentions of entities (i.e. names of people, locations, organizations and products) in tweets (twitter messages) and link them to the corresponding pages present on popular knowledge-bases such as DBpedia.

Your task is to implement a system for named entity recognition and linking on twitter. The system must be multilingual.



Entity Linking

New *Frozen* Boutique to Open at *Disney*'s *Hollywood Studios*

/wiki/Frozen_(2013_film)

/wiki/The_Walt_Disney_Company

/wiki/Disney's_Hollywood_Studios

As a starting point you can read this paper and try to implement and improve the model described there:
Wikipedia-based WSD for multilingual frame annotation

Data

For evaluation purposes, we will use the dataset distributed for the NEEL-IT task at EVALITA 2016. Evalita is the 5th edition of the evaluation campaign for NLP and speech Tools for Italian.
Look here for more details about the data:
http://neel-it.github.io/data

Evaluation

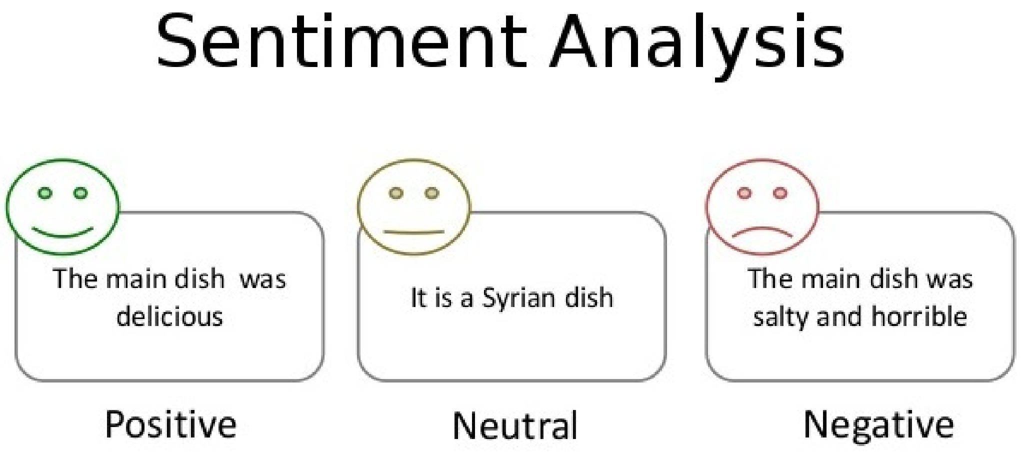
The system evaluation will take into account the number of entities correctly identified and linked. More information about the evaluation here:
http://neel-it.github.io/

Sentiment Analysis on twitter dataset

Description

Sentiment analysis is the task of identifying positive and negative opinions, emotions and evaluations in text.
A system for sentiment analysis annotates messages from twitter with sentiment polarity. Sentiment polarity defines the attitude of a speaker when writing a piece of a document. Each document is classified as positive, negative or neutral based on its content.

Your task is to implement a model for Sentiment Analysis of tweets.


Sentiment Analysis
The main dish was delicious
It is a Syrian dish
The main dish was salty and horrible
Positive
Neutral
Negative

As a starting point you can implement the model described here:
Predicting tweet sentimentpolarity combining micro-blogging, lexicon and semantic features; or
the state-of-the-art model using **Neural Networks** described here:
Training Deep Convolutional Neural Network for TwitterSentiment Classification

Data

The resulting system will be evaluated on:

- the SemEval-2015 Task 10 Dataset (English)
  http://alt.qcri.org/semeval2015/task10/index.php?id=data-and-tools
- the SENTIPOLC dataset released for EVALITA 2016 (Italian).
  http://www.di.unito.it/~tutreeb/sentipolc-evalita16/index.html

Evaluation

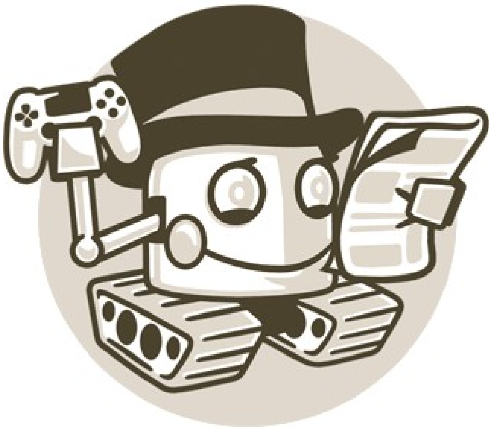
The system will be evaluated by considering the number of tweets correctly classified as positives, negatives and neutrals.
More information about the evaluation here: http://www.di.unito.it/~tutreeb/sentipolc-evalita16/index.html

Telegram bot

Description

The goal of this project is to build a bot for the instant messaging app Telegram. A bot is a computer program that interacts with users in a chat. Users can interact with bots by sending them messages, commands and inline requests.

Bot can answer with text or by playing music, showing images, etc..



For more information about telegram bots, see here:
https://core.telegram.org/bots

This year, telegram is granting one million dollars to people developing bots for the telegram messaging app. You can find more information here:
https://telegram.org/blog/botprize

Your task is to implement a bot that performs some non-trivial task.

The bot must use the **UIMA** framework and **DKPro** annotators to process the text input by the users.

For example, you can build a bot that:
(i) reads messages input by the users in a chat; and
(ii) prints information about the entities mentioned in a text such as places, people, etc...

… or you can work with people doing the project on sentiment analysis and build a bot detecting sentiment polarity of telegram chat messages on the fly.

For this project, you are free to use any API you want, like MediaWiki, Foursquare APIs, etc...