Visualization as an aid to Text Processing

Honours Project 2015

Nikola Nikolov

s1245947

**Food for thought**

Bookeh + Ipython vs Flask + D3  
  
Which one is a better way to create a visual representation of dependency parsing.

How to extract data from each step of the algorithm to be able to visualize it step by step?

How to evaluate which implementation is better?

Is bokeh more ‘static’ than d3 in terms of animation?

Presentation, executable plan (timetable).

Added Werkzeug python library to check if the file uploaded by the user (the ConLL file) has a secure name. (Pass it a filename and it will return a secure version of it. This filename can then safely be stored on a regular file system and passed to [**os.path.join()**](http://docs.python.org/dev/library/os.path.html#os.path.join). The filename returned is an ASCII only string for maximum portability.

On windows systems the function also makes sure that the file is not named after one of the special device files.

>>> secure\_filename**(**"My cool movie.mov"**)**

'My\_cool\_movie.mov'

)

What is Dependency Parsing?  
Dependency Parsing is basically the process of parsing natural language to a Dependency Tree. There are two ways of achieving this goal – by using constituency grammar or dependency grammar. The main difference between the two is that in dependency grammar for every word or morph in the sentence there is exactly one single node corresponding to it, where as in constituency grammar we can have one or more nodes corresponding to the same word or morph