Artifical Intelligence

Exercise 6: Project Proposal

December 19, 2017

Team

Sebastian Bek, Marvin Klaus, Daniela Schacherer

Problem Definition

Nowadays, we get constantly more used to having AI systems ease our every day lives. In particular in the context of electronic devices we use the available features for facilitated text-processing. This can for instance be observed on mobile phone or search engines.
In this context we propose the development of an text autocompletion and prediction system. For that purpose, we will build a model suggesting possible word-endings given a prefix. In advance, it will predict the following word relative to the cursor position.

Dataset and Agent Environment

By means of collection of data (sets), we will try to design a text bot gathering multiple english text sources in diverse text files. We will use several datasets originating from e.g.:
https://nlp.stanford.edu/links/statnlp.html#Corpora
We will divide these data into a sufficiently large training set and a test set.

Approach

We are planning to adress this task by using an machine learning approach. We will implement a neural networks and train it using the selected set of training data. More specifically, we would like to use a long short-term memory (LSTM) network - a special type of recurrent neural networks (RNN) - as they are commonly used in language recognition tasks, according to literature. Through the recurrent connections a short term memory is created for every layer of the neural network which can be interpreted as memory. Since we don't consider ourselves experts in AI, we can't assess how well-suited our approach is for the given task. However, as we are interested in neural networks, we would like to try this method in order to gain a deeper insight into the topic.

Possible Challenges:
A possible challenge arises from words, which are occurring much more frequently than others, e.g. "in, the, that", since these do not indicate any hint about possibly following words. We have to cope with this task by finding an appropriate solution. Other than that, we would have

to deal with spelling mistakes of the user. We could think of a SpellCheck routine, suggesting the correctly spelled word and additionally the predicted next word.

Evaluation and expected results

Firstly we would access the accuracy (amount of reasonable predictions) of our network over time for the training set as well as for the test set. We aim at minimizing the difference between the accuracy for the training set and the accuracy for the test set. We will classify a prediction as reasonable if the input text together with the predicted word appear in the test set.

Hardware

We do not think we will require any specific hardware system, yet our approach's (execution and evaluation) requirements will not be especially demanding.