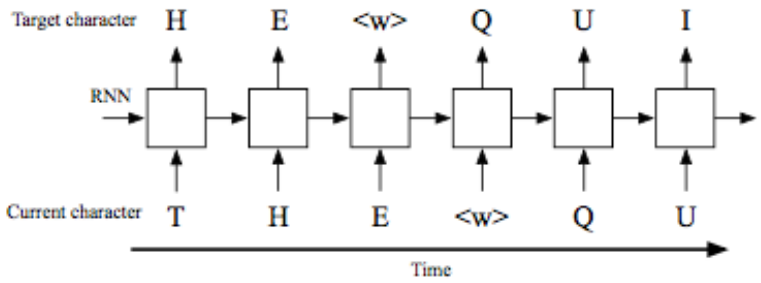
Homework 03

Neural Networks and Deep Learning

Dr. Alberto Testolin
Dr. Federico Chiariotti
A.A. 2019-20

Sequence modeling with RNNs



- Download some text corpora from Gutenberg: http://www.gutenberg.org/browse/scores/top
- Train, validate and test your recurrent neural network using PyTorch
- Write the report (as always, 3 pages length + Appendix for additional figures/tables)
- To create the rankings, the quality of the text generated from your model will be evaluated using a peer-review procedure

Don’t worry, rankings are just for fun!

(and, in this specific case, they are qualitative rather than quantitative)

Sequence modeling with RNNs

Homework objectives:

1. Extend the available PyTorch script (Lab04) or create a brand-new one, in order to:
   - Train a recurrent neural network on one (or more) text dataset:
     - Using proper methods for tuning the hyperparameters / avoiding overfitting (however, due to the computational complexity of RNNs a systematic search is not required)
     - Implementing a seq2seq task, where the goal is to predict the next character given some initial characters
       - » Some text pre-processing is required, for example for reducing the size of the alphabet (you can delete rare characters and punctuation symbols, or group them together)
       - » The initial characters and generation length should be provided as input parameters to the script
       - » At each timestep, the predicted character should be produced either using the maximum (as in Lab04) or, better, by sampling from a softmax distribution
   - Analyze the trained network by generating sequences of characters, providing different contexts as input

2. Write a short report describing your work and the results achieved (figures are appreciated)

3. Send the Homework through the Moodle platform:
   - the script **MUST** work by running the following command:

```
python trained_model.py --seed "Initial context" --length n
```

which should print as output a sequence of *n* characters produced by the model, given the string specified in the parameter "initial context"

Sequence modeling with RNNs

Homework objectives:

2. [Optional]

- Rather than working at the character-level, convert entire words into vectors (word embedding) by using some standard libraries (e.g., word2vec). Note that some word embedding tools are available through PyTorch. Note also that if you plan to use this approach for the competition, you should save all word vectors into a separate file and load them during the generation (i.e., we won't run word2vec)
- Implement more advanced architectures (e.g., LSTM with additional layers, encoder-decoder architectures)

- Train the RNN over different datasets (for example, containing musical sequences):
  - http://deeplearning.net/tutorial/rnnrbm.html
  - http://www.piano-midi.de
  - https://magenta.tensorflow.org/datasets

  (If you choose this optional point, you can just focus on the music domain and avoid the NLP task; however, please note that the ranking are produced only for the NLP task)