

# Lab Seven: RNNs

---

**Due** May 11 by 11:59pm**Points** 10**Submitting** a file upload

---

## Lab Assignment Seven: Recurrent Network Architectures

In this lab, you will select a prediction task to perform on your dataset, evaluate a recurrent architecture and tune hyper-parameters. If any part of the assignment is not clear, ask the instructor to clarify.

This report is worth 10% of the final grade. Please upload a report (**one per team**) with all code used, visualizations, and text in a rendered Jupyter notebook. Any visualizations that cannot be embedded in the notebook, please provide screenshots of the output. The results should be reproducible using your report. Please carefully describe every assumption and every step in your report.

### Dataset Selection

Select a dataset similarly to lab one: text. That is, the dataset should be text data (or a time series sequence). In terms of generalization performance, it is helpful to have a large dataset of similar sized text documents. It is fine to perform binary classification or multi-class classification. The classification can be "many-to-one" or "many-to-many" sequence classification, whichever you feel more comfortable with.

### Grading Rubric

- Preparation (30 points total)
  - [10 points] Define and prepare your class variables. Use proper variable representations (int, float, one-hot, etc.). Use pre-processing methods (as needed). Describe the final dataset that is used for classification/regression (include a description of any newly formed variables you created). **Discuss methods of tokenization in your dataset as well as any decisions to force a specific length of sequence.**
  - [10 points] Choose and explain what metric(s) you will use to evaluate your algorithm's performance. You should give a **detailed argument for why this (these) metric(s) are appropriate** on your data. That is, why is the metric appropriate for the task (e.g., in terms of the business case for the task). Please note: rarely is accuracy the best evaluation metric to use. Think deeply about an appropriate measure of performance.
  - [10 points] Choose the method you will use for dividing your data into training and testing (i.e., are you using Stratified 10-fold cross validation? Shuffle splits? Why?). **Explain why your chosen method is appropriate or use more than one method as appropriate.** Convince me that your cross validation method is a realistic mirroring of how an algorithm would be used in practice.
- Modeling (60 points total)
  - [20 points] Investigate at least two different recurrent network architectures (perhaps LSTM and GRU). Be sure to use an embedding layer (pre-trained, from scratch, or both). Adjust hyper-parameters of the networks as needed to improve generalization performance.

- **[20 points]** Using the best RNN parameters and architecture, add a second recurrent chain to your RNN. The input to the second chain should be the output sequence of the first chain. Visualize the performance of training and validation sets versus the training iterations.
- **[20 points]** Use the method of cross validation and evaluation criteria that you argued for at the beginning of the lab. Visualize the results of all the RNNs you trained. Use proper statistical comparison techniques to determine which method(s) is (are) superior.
- **Exceptional Work (10 points total)**
  - You have free reign to provide additional analyses.
  - One idea (**required for 7000 level students**): Use t-SNE (or SVD) to visualize the word embeddings of a subset of words in your vocabulary. Try to interpret what each dimension reflects (in your own words). That is, try to explain what aspect of the language is encoded in the reduced dimensionality embedding.
  - Another Idea (NOT required): Try to create a RNN for generating novel text.