**Poetic License:**

**Insights from Machine Classification, Clustering, and Generation of Poetry**

Cassandra Engstrom

Hunter College - Seminar in Machine Learning CSCI 795 - Fall 2021

**Project Motivation and Description**

Poetry is a notoriously enigmatic form of language that frequently ignores standardized compositional criteria while rigorously upholding others. Poets enjoy what is known as “poetic license” when formulating their works, a policy that allows them to develop a highly idiosyncratic style characterized by semantic and syntactic trademarks and patterns of rule-breaking/rule-following. Advances in Natural Language Processing (NLP) and text mining have made it possible for machines to categorize, compare, and emulate samples from different textual sources. Due to the highly heterogeneous nature of poetic language, however, it is unclear how successfully Machine Learning Techniques can truly learn and derive patterns, even within individual poets or specific schools/eras of poetic output.

The aim of this study is therefore severalfold:

1. To compare the performance of different Supervised Classifiers in predicting the category (time period/style) of poems
2. To utilize unsupervised clustering techniques to examine the relative homogeneity within poetic groupings as well as examine the textual similarity between different poets
3. To generate novel poetry using a word-based RNN

This analysis is largely inspired by the approach taken in [1], which compares several religious texts from Eastern and Western cultures to understand potential textual (and therefore perhaps belief-based) similarities.

**Team Member’s Role and Contribution**

*Cassandra Engstrom:*

1. **Data Cleaning processing:** Remove stop words using NLTK; remove punctuation
2. **Feature Vectorization:** Create Bag of Words (BoW) Document Term Matrix (DTM) AND Term Frequency – Inverse Document Frequency (TF-IDF) tables. Eventually, the efficacy of these two vectorization approaches will be contrasted in prediction and clustering approaches.
3. **Prediction Model Comparison:** Compare the predictions of different supervised classification methods (K-nearest neighbor, Random Forest, Support Vector Machine, Multinomial Logistic Regression, Naïve Bayes) on the prediction of both poem type, poem age, and poem author, for both kinds of feature vectorization above.
4. **Similarity Matrices:** Use similarity matrices to understand how similar poems belonging to each author, type(subject) and age(era) are to one another. Report which groupings include poems with the greatest and least similarity to one another.
5. **Clustering (compare to ground truth):** Use unsupervised clustering techniques (K-Means, Spectral Clustering) to cluster poems based on K (cluster number), which is determined by ground truth. In other words, compare the output of the clustering techniques to true labels, wherein each label is a unique combination of poem age and style (6 classes total). The K-means clustering implementation will be carried out using different distance measures (Jaccard, Cosine, Manhattan, and Euclidean), eventually reporting and selecting the best one.
6. **Clustering (examine similarity of authors):** Utilize clustering technique to reveal which authors are most similar to one another. This will be done by incrementally iterating through the K means clustering procedure from K=2 to K=N-1 where N is the total number of authors in the dataset. Note which authors get grouped together in intermediate steps.
7. **Novel Poem Generation**: Train word-based RNN on one or several of the poem type-age groupings (6 classes) to generate novel samples of text based on each category.

**CSCI 795 Related Topics**

* Feature Vectorization
  + Bag of Words (BoW)
  + Term Frequency – Inverse Document Frequency (TF-IDF)
* Supervised Techniques
  + K-nearest neighbor
  + Naïve Bayes
  + Multinomial Logistic Regression
  + Random Forest
  + Support Vector Machine
* Unsupervised Techniques
  + K-Means Clustering
  + Spectral Clustering
* Model Evaluation
  + Precision, recall, F1
  + Similarity Matrices

**Dataset**

We will use a poetry dataset downloadable on Kaggle (<https://www.kaggle.com/ultrajack/modern-renaissance-poetry/version/1>) which pulls examples of Modern and Renaissance poetry from poetryfoundation.org.

This dataset, in the form of .csv file, has 574 entries and 5 columns: author, content (the actual poem), poem name, age (Renaissance or Modern), and type (Nature, Love, Mythology & Folklore)

**Demo**

We hope to present an extensive comparison of Classifier Models/Clustering Techniques and their performance on this dataset by generating appropriate plots and providing metrics. We will also generate visualizations that communicate the findings of the K-means clustering analysis designed to probe textual similarity amongst authors. Finally, we will provide samples of text generated by the RNN alongside examples of some of the samples each model was trained on.

**Project Timeline**

|  |  |
| --- | --- |
| **10/19-**10/25 | Prepare data for analysis. Remove Stop Words. Vectorize each entry in dataset according to BoW and TF-IDF procedures (separately) |
| **10/26-**11/01 | Train Classifier Models: KNN, Multinomial Logistic Regression, Naïve Bayes on author, poem type, and poem age |
| **11/02-**11/08 | Train Classifier Models: Support Vector Machine, Random Forest on author, poem type, and poem age |
| **11/09-**11/15 | Generate similarity matrices Implement K-Means & Spectral Clustering Procedures to compare to ground truth and examine inter-author similarities. |
| **11/16-**11/22 | Train RNN based on each poem type |
| **11/23-**11/29 | Train RNN based on each poem type |
| **11/30-**12/06 | Generate plots, visualizations based on findings |
| **12/07-**12/15 | Prepare presentation / demo |

**Evaluation**

**CLASSIFICATION**: Prediction models/classifiers will be evaluated on based on accuracy scores, as well as precision, recall, and F1 scores, and the corresponding AUC. We may also choose to evaluate the confusion matrices for the prediction exercises of both author and poem type to understand error patterns.

**CLUSTERING:** Firstly, the output of the clustering procedures for comparison to ground truth will be evaluated in terms of the [unadjusted & adjusted] Rand Index, the [normalized & adjusted] mutual information (NMI & AMI), as well as homogeneity and completeness scores, and their harmonic mean ‘V-measure’.

Secondly, the output of the incremental K-means clustering procedure to compare authors will be evaluated in terms of expert field knowledge. For example, were the poets clustered at a near distance all alive during a similar period or are they known to be part of the same stylistic ‘school’ of poets? (<https://en.wikipedia.org/wiki/List_of_poetry_groups_and_movements>)?

**GENERATION:** The poems generated by the RNN will be evaluated based on the frequency of (normalizing for era) grammatical and spelling errors relative to sample length, as well as (potentially) a Turing-style test where humans pick out which poems were likely ML-generated. Unfortunately, we cannot use ‘perplexity’ as a measure, because there is no test set to compare these novel generated samples against.

**References**

[1] Sah, P. & Fokoue, E (2019). ‘What do Asian Religions Have in Common? An Unsupervised Text Analytics Exploration’, *arXiv: 1912.10847v1.*

**Resources: Clustering**

<https://scikit-learn.org/stable/modules/clustering.html>

**Resources: Word- & Char- based RNNs**

<https://www.kaggle.com/ultrajack/poetry-analysis-using-ai-machine-learning/notebook#Prediction-result>

<https://www.tensorflow.org/text/tutorials/text_generation>

<http://karpathy.github.io/2015/05/21/rnn-effectiveness/>

<https://gist.github.com/karpathy/d4dee566867f8291f086>

<https://github.com/jcjohnson/torch-rnn>