An Unsupervised Clustering Model for Ancient Chinese Poems

JIAXIN GUAN

Motivation

- Ancient Chinese poems have high literature values and are widely studied by scholars around the world

- The way that people mostly study these poems are rather subjective:
  - No quantitative analysis
  - Purely empirical
  - Lots of personal and biased deduction

- By building an unsupervised clustering model for these poem, we hope to answer the following questions:
  - Which poet has the most consistent style across all his poems?
  - Which poets have similar style in their writing?
  - Is there a trend in style that develops through time?

The Dataset

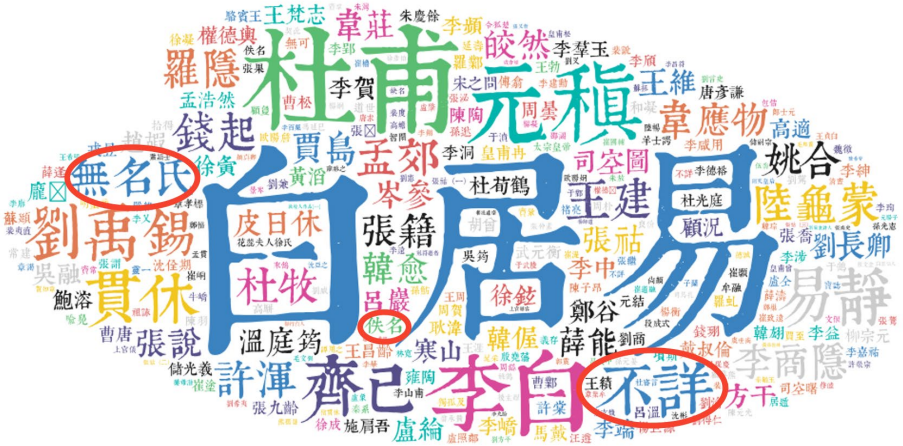
- Chinese Poetry repo on Github:
  - https://github.com/chinese-poetry/chinese-poetry

- 55,000 Tang dynasty poems in JSON format
  - Strain
  - Author
  - Paragraph
  - Title


```
[
 {
  "strains": [
   "平平平仄仄，平仄仄平平。",
   "仄仄平平仄，平平仄仄平。",
   "平平平仄仄，平仄仄平平。",
   "平仄仄平仄，平平仄仄平。"
  ],
  "author": "太宗皇帝",
  "paragraphs": [
   "秦川雄帝宅，函谷壯皇居。",
   "綺殿千尋起，離宮百雉餘。",
   "連甍遙接漢，飛觀迴凌虛。",
   "雲日隱層闕，風煙出綺疎。"
  ],
  "title": "帝京篇十首 一"
 }
]
```

A Failed Attempt…

- Initially, aim to construct *supervised classification* model for the poems
  - Classes are the poets
  - Goal: Help identify poems with unknown authors
- Issues:
  - The number of classes is too large: (3719 poets in total)
  - A lot of the poets have very small number of poems (datapoints)
- Solution:
  - Limit us to the top 50 authors?
  - Doesn't make sense, since these author's poems are well maintained



Word cloud for authors of poems from the Tang dynasty. The three circled authors are "noname", "anonymous", and "unknown", from left to right respectively.

Feature Extraction (Paragraph)

- No clear definition of “words”
  - Break the text into characters
  - Use an initial segmentation module to break the text into “words”
    - The number of Chinese characters is large
    - The number of “words”, or character tuples, would be huge
- The Poems are well structured and concise
  - Mostly 20 or 28 characters
  - 4 verses, with 5 or 7 characters per verse
  - “炼字” Lianzi (“Character Curating”)：Try to find the best character to put in the spot
    - The characters bear more meanings than modern Chinese language
- Breaking into characters might be the best bet

Feature Extraction (Strains)

- Strain
  - Rules that the tones of the character in the poem must follow
  - “平” Ping: Level Tone
  - “仄” Ze: Oblique Tone
  - “〇” Wildcard: Either level or oblique

- ping_percentage: The percentage of Level Tones in the whole poem
- ze_percentage: The percentage of Oblique Tones in the whole poem
- wildcard_percentage: The percentage of Wildcard Tones in the whole poem

- end_ping_percentage: The percentage of verses that end with a level tone
- end_ze_percentage: The percentage of verses that end with a oblique tone
- end_wildcard_percentage: The percentage of verses that end with a wildcard tone

- symmetry_score: For each verse (x1, x2), count the number of strains in x1 that is the opposite of the strain in x2 at the corresponding position

Unsupervised Clustering Model (In Progress)

- We plan to use the following models for unsupervised clustering and compare the results:
  - K-Means
  - Spectral Clustering
  - Ward Hierarchical Clustering
  - DBSCAN
  - OPTICS

Evaluation

- Purity score with the classes being the authors
- The Silhouette Score
- The Calinski-Harabasz Index
- Davies-Bouldin Index
- Examining the clustering of the poems of a single author

Thank you!