Handwritten Character Recognition (HCR)

Telugu Language (తెలుగు భాష)

Suman AGK (గోపాల కృష్ణ సుమన్ అడుసుమిల్లి)

Capstone Project @ General Assembly (2020)

Index

- Problem Statement
- Background
- Acquiring Data
- Parse & Mine Data
- Refine Data
- Modelling & Results
- Conclusion & Future work

Problem statement

- Objective:
  - Recognize handwritten Telugu language characters and map them to their Unicode character set
- Features:
  - Handwritten Telugu character images are converted into binary matrix using their pixel level data
- Target:
  - Machine recognizable Unicode set Telugu characters
- Success rate (Accuracy score):
  - We will see in the end ☺

Background

- Handwritten Character Recognition(HCR)
    - Process of classifying hand written characters
    - Features extracted from each character
- Benefits of HCR
    - Mail sorting, processing of bank cheques, reading aid for blind, document reading and postal address recognition, form processing, digitalizing old manuscripts.
- Challenges
    - Varies from person to person with different style, speed, age, mood and even gender
    - Vast number of character classes
- Telugu Language
    - Dravidian language, predominantly spoken in the Indian states of Andhra Pradesh and Telangana and the Union Territory of Puducherry
    - Ranks 4th among languages with the highest number of native speakers in India
    - Ranks 15th in the list of most widely-spoken languages worldwide
    - 80-90 million Telugu speakers worldwide

Background (contd.)

- Telugu Language Character set:
  - 56 base alphabets
  - Including Vowels, Consonants and half characters.
  - Consonants combine with vowels to make new alphabets
  - Consonants also combine themselves and make more alphabets
- Vowels: 18 Vowels in total, 'అ','ఆ','ఇ','ఈ','ఉ','ఊ','ఋ','ౠ','ఌ','ౡ','ఎ','ఏ','ఐ','ఒ','ఓ','ఔ‘, 'అం','అః'
- Consonants: 35 Consonants, 'క', 'ఖ', 'గ', 'ఘ', 'ఙ', 'చ', 'ఛ', 'జ', 'ఝ', 'ఞ', 'ట', 'ఠ', 'డ', 'ఢ', 'ణ', 'త', 'థ', 'ద', 'ధ', 'న', 'ప', 'ఫ', 'బ', 'భ', 'మ', 'య', 'ర', 'ల', 'వ', 'స', 'ష', 'శ', 'హ', 'ళ', 'ఱ'
- Half characters (Special characters): 'ం','ః' ,'్'
- Combination characters
  - Consonants with vowels:
    - Example consonant, 'క‘, combination of 'క‘ with all vowels, 'కా', 'కి', 'కీ', 'కు', 'కూ', 'కృ', 'కౄ', 'కె', 'కే', 'కై', 'కొ', 'కో', 'కౌ', 'కం', 'కః', 'క్‘
  - Consonants with consonants
    - Example consonant, 'క‘, combination of 'క‘ with all consonants, 'క్క', 'క్ఖ', 'క్గ', 'క్ఘ', 'క్ఙ', 'క్చ', 'క్ఛ', 'క్జ', 'క్ఝ', 'క్ఞ', 'క్ట', 'క్ఠ', 'క్డ', 'క్ఢ', 'క్ణ', 'క్త', 'క్థ', 'క్ద', 'క్ధ', 'క్న', 'క్ప', 'క్ఫ', 'క్బ', 'క్భ', 'క్మ', 'క్య', 'క్ర', 'క్ల', 'క్వ', 'క్స', 'క్ష', 'క్శ', 'క్హ', 'క్ళ', 'క్ఱ'
- Numbers: '౦','౧','౨','౩','౪','౫','౬','౭','౮','౯'

- http://www.learningtelugu.org/vowels-consonants-and-combinations.html

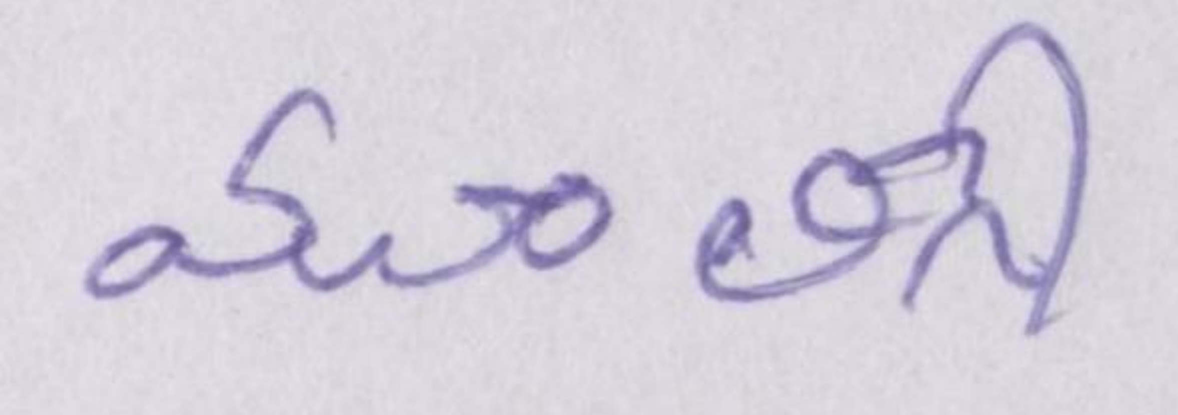
Data source

- Center for Visual Information Technology,
  International Institute of Information Technology (IIIT),
  Gachibowli, Hyderabad - 500 032,
  Telangana, INDIA.

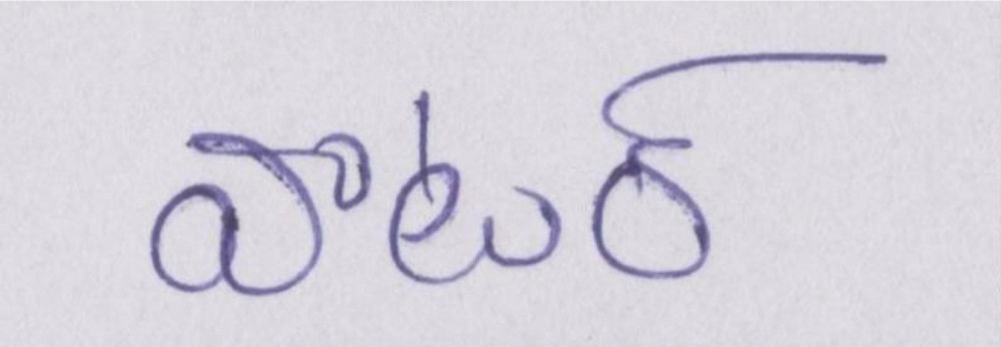
- https://cvit.iiit.ac.in/research/projects/cvit-projects/indic-hw-data

- A Telugu dataset comprising of over 120K handwritten words

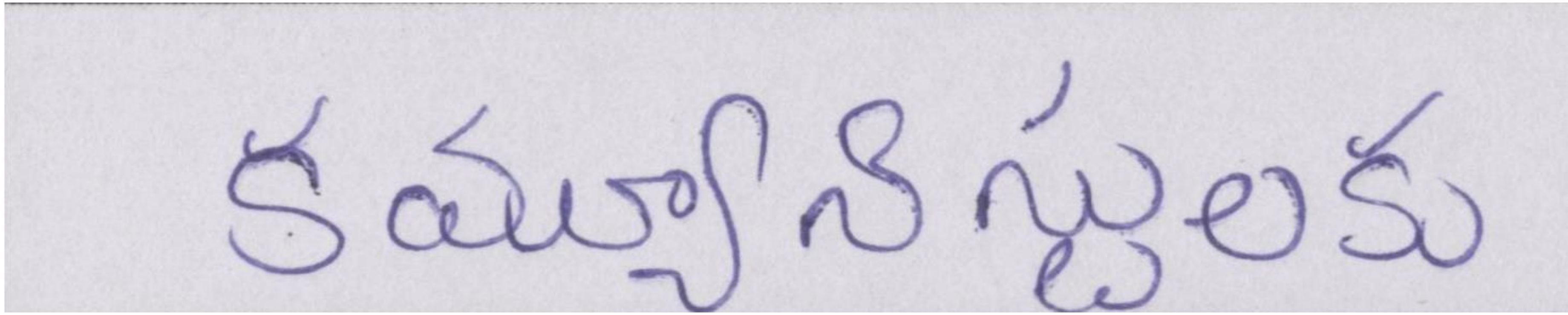
Data source (sample images)



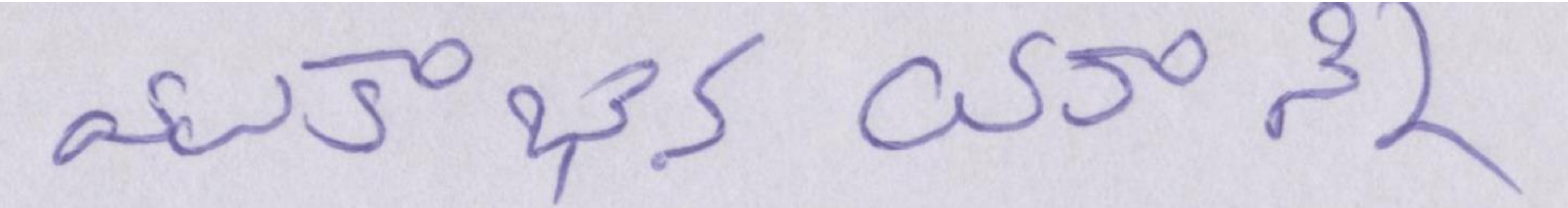
[illegible]



వాటర్



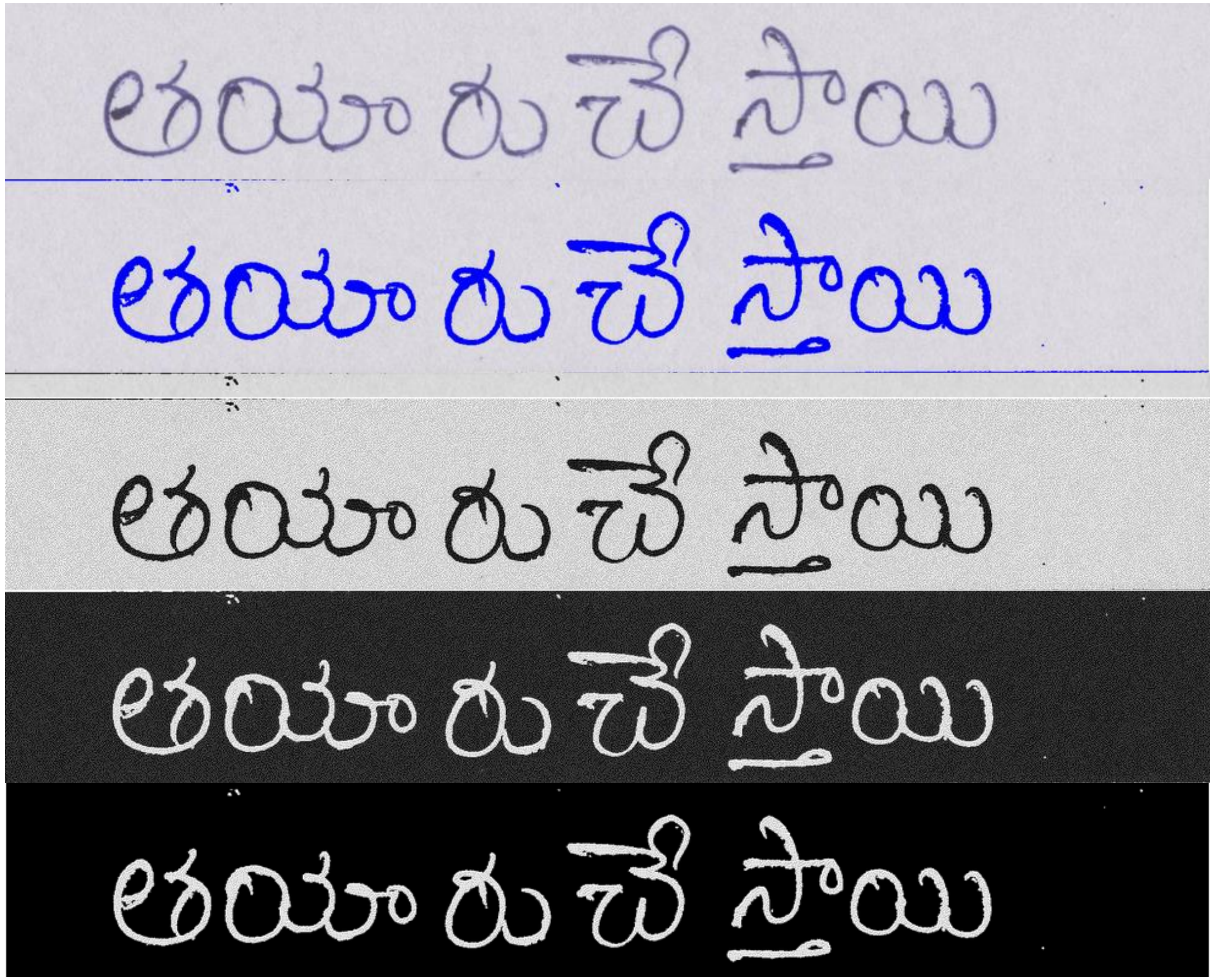
కమ్యూనిస్టులకు



[illegible]

Cleansing data (EDA)

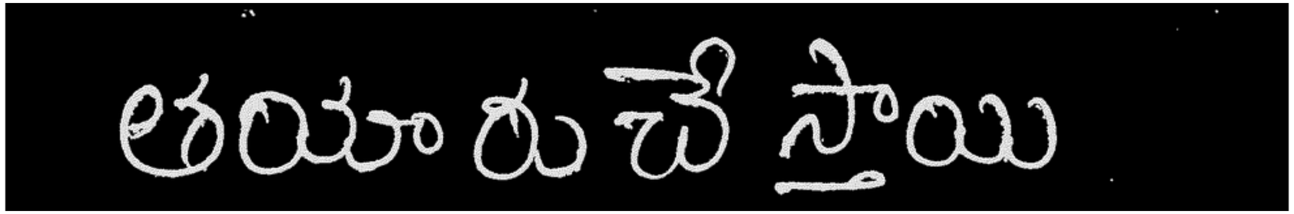
- Stage 0: Original Image
- Stage 1: Enhance the intensity of the blue colour
- Stage 2: Mono-chrome image
- Stage 3: Black & white image
- Stage 4: White & Black image
- Stage 5: Remove noise

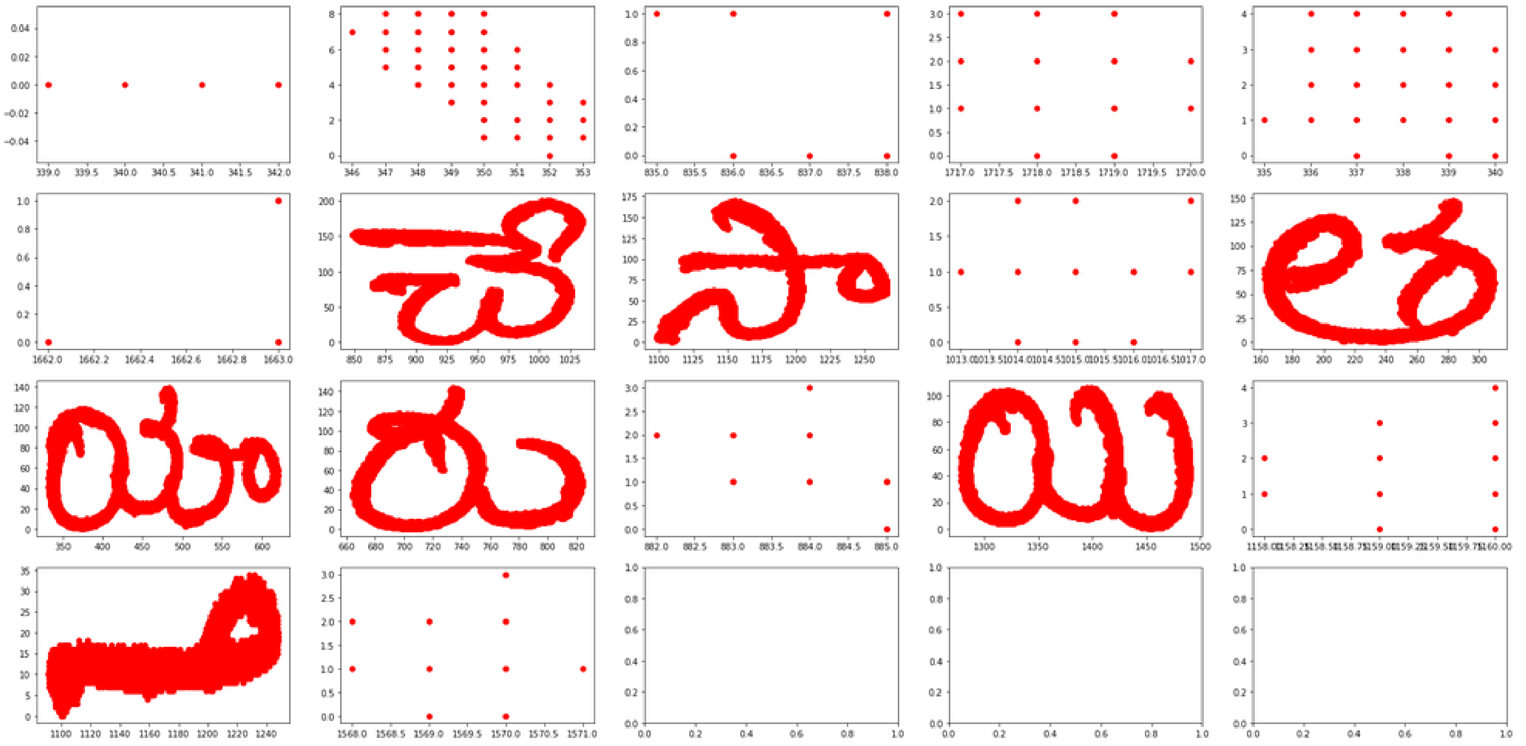


Labels are split into character labels.
Source Label : 'తయారుచేస్తాయి‘
Character Labels: 'త', 'యా', 'రు', 'చే', 'స్తా', 'యి'

Mining Data (EDA)

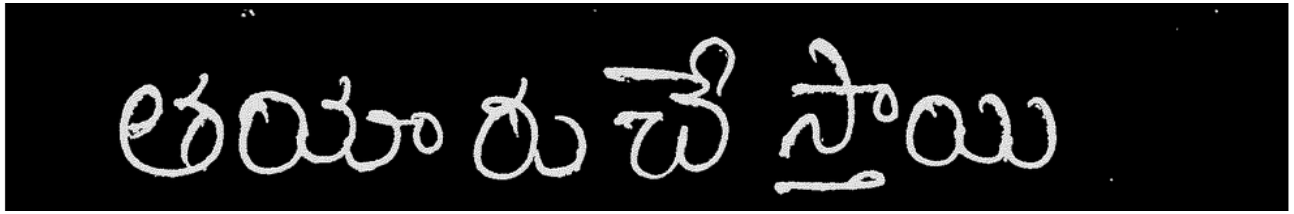
- DB-Scan is used to find clusters and thus the characters of the word given

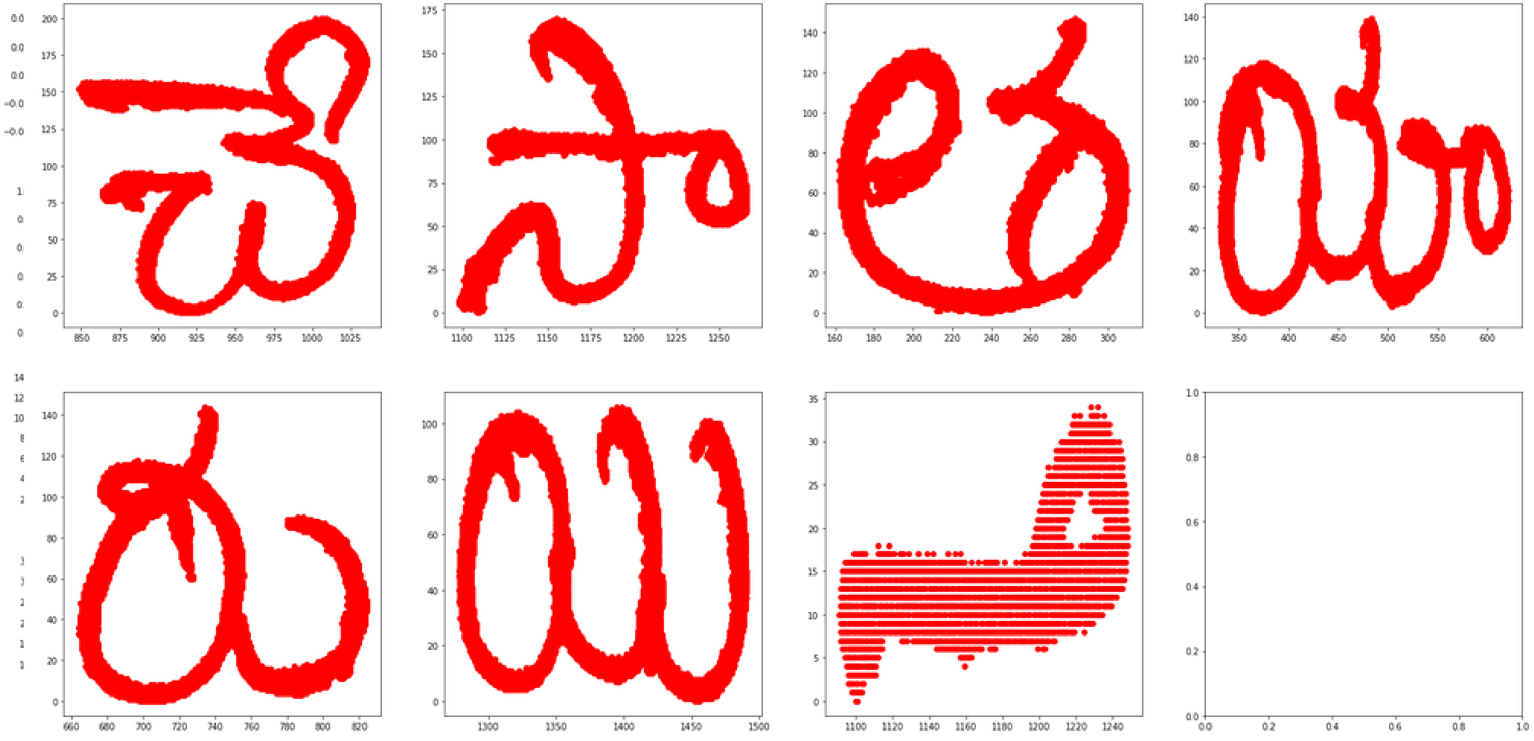




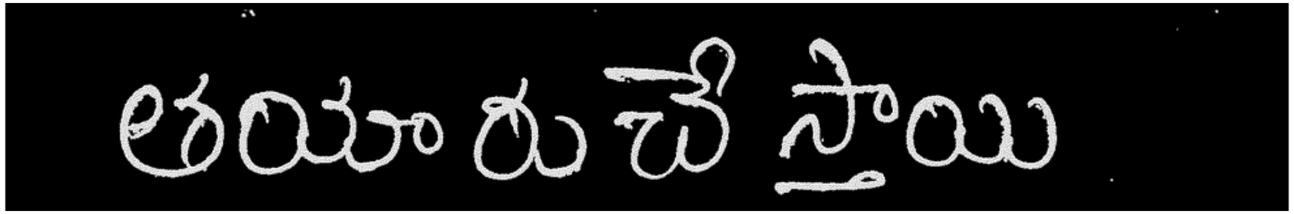
Mining Data (EDA)

- DB-Scan is used to find clusters and thus the characters of the word given
- Remove small clusters

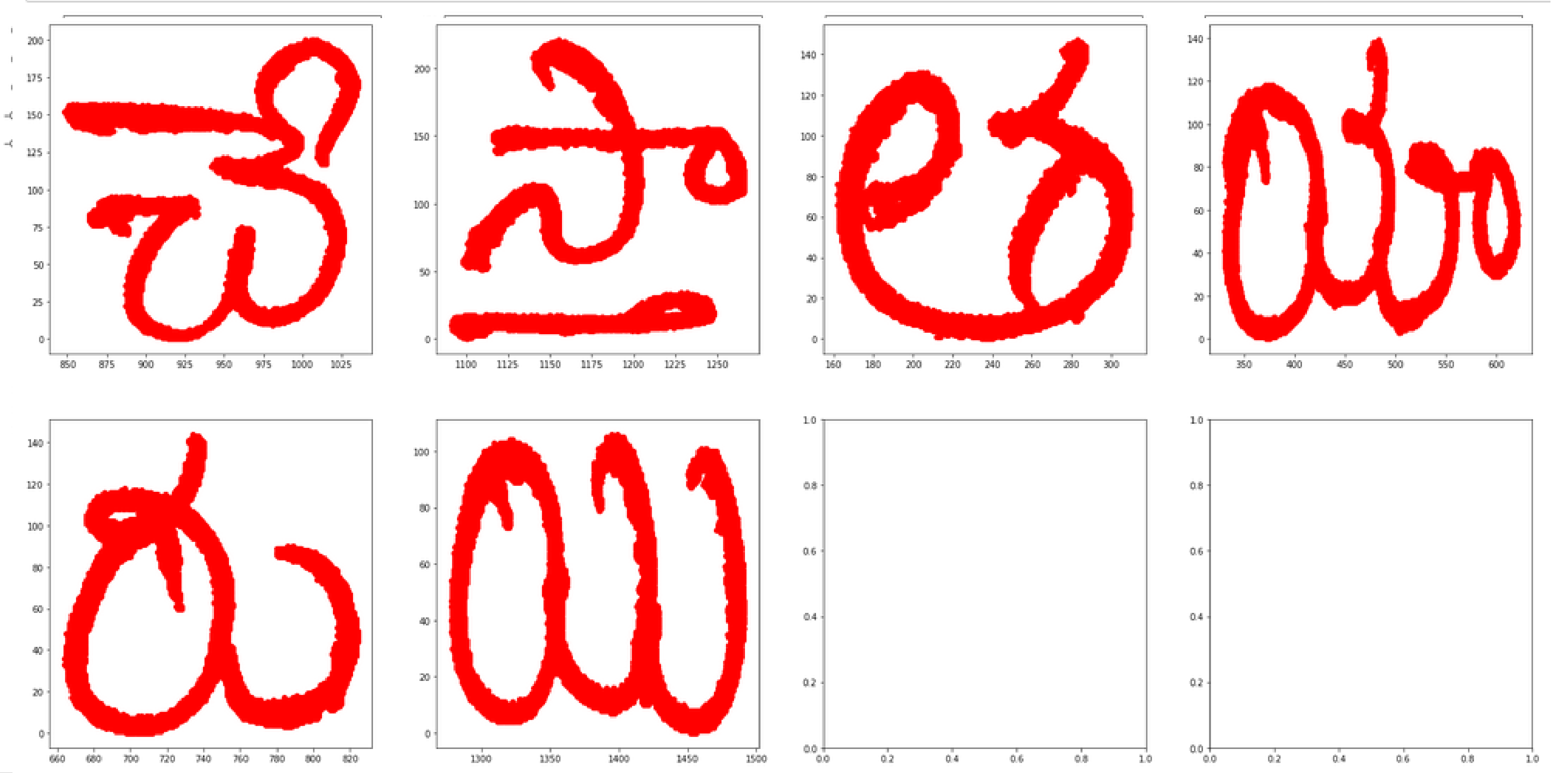




Mining Data (EDA)

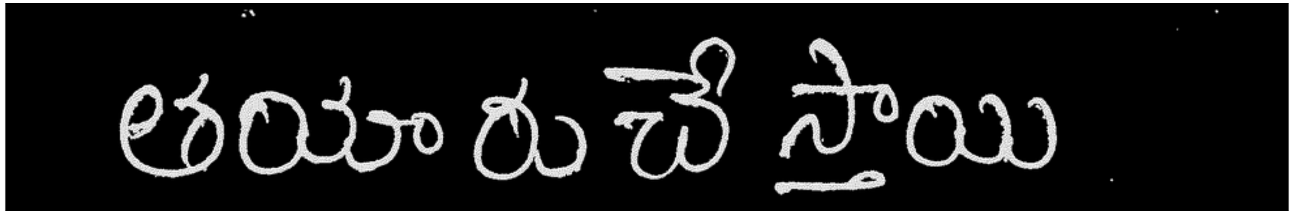


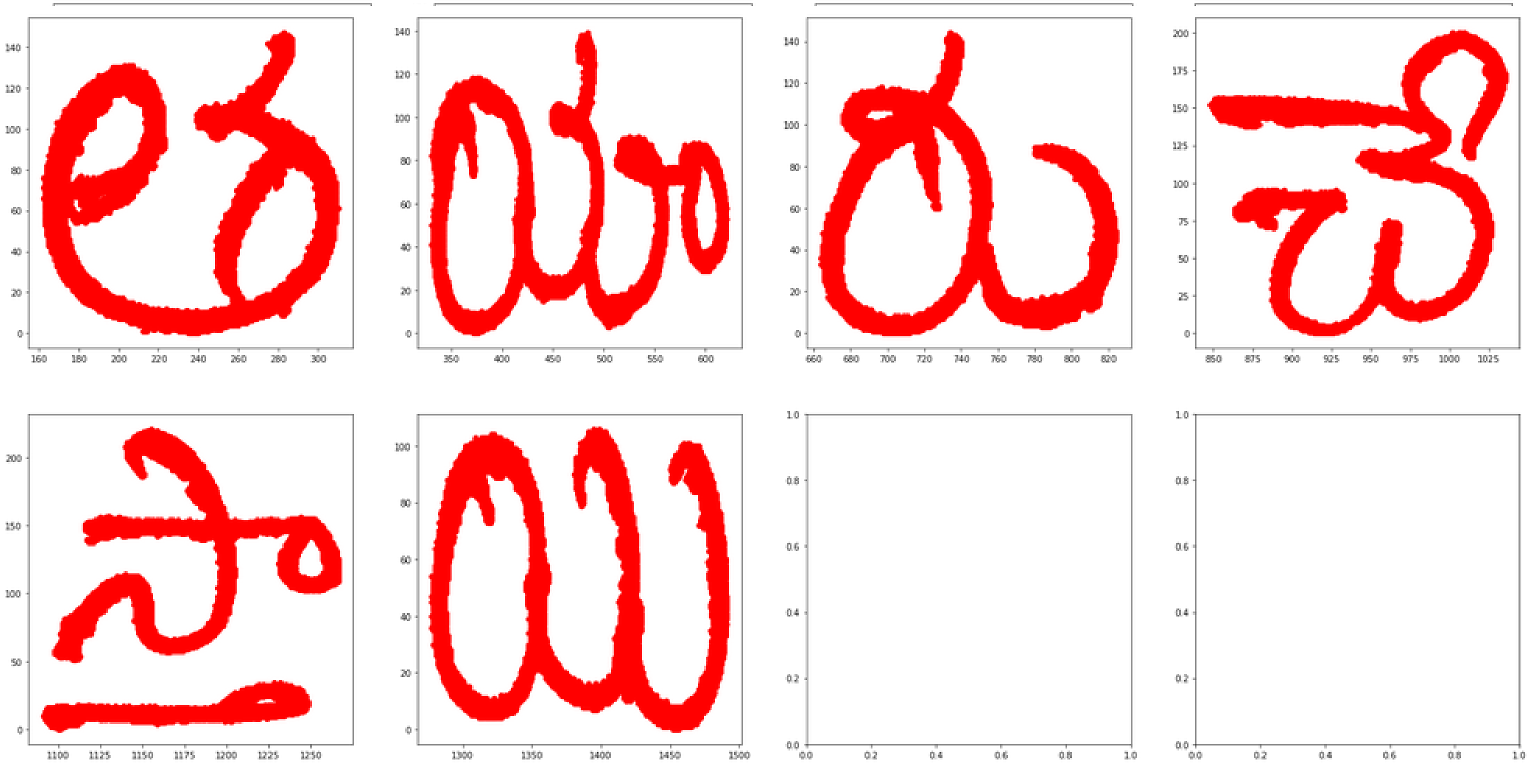
- DB-Scan is used to find clusters and thus the characters of the word given
- Remove small clusters
- Combine or Split clusters as required



Mining Data (EDA)

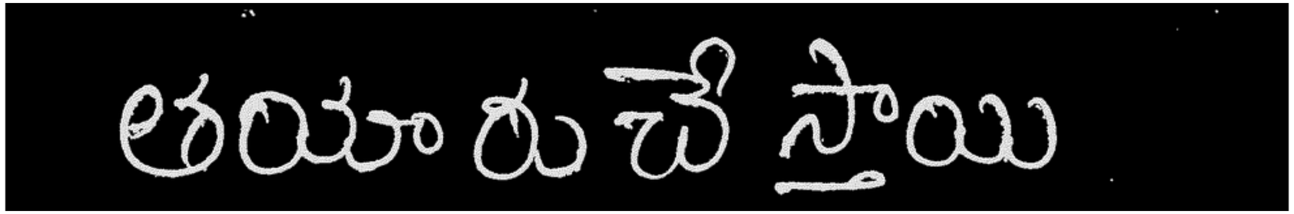
- DB-Scan is used to find clusters and thus the characters of the word given
- Remove small clusters
- Combine or Split clusters as required
- Sort the characters as they are written in image

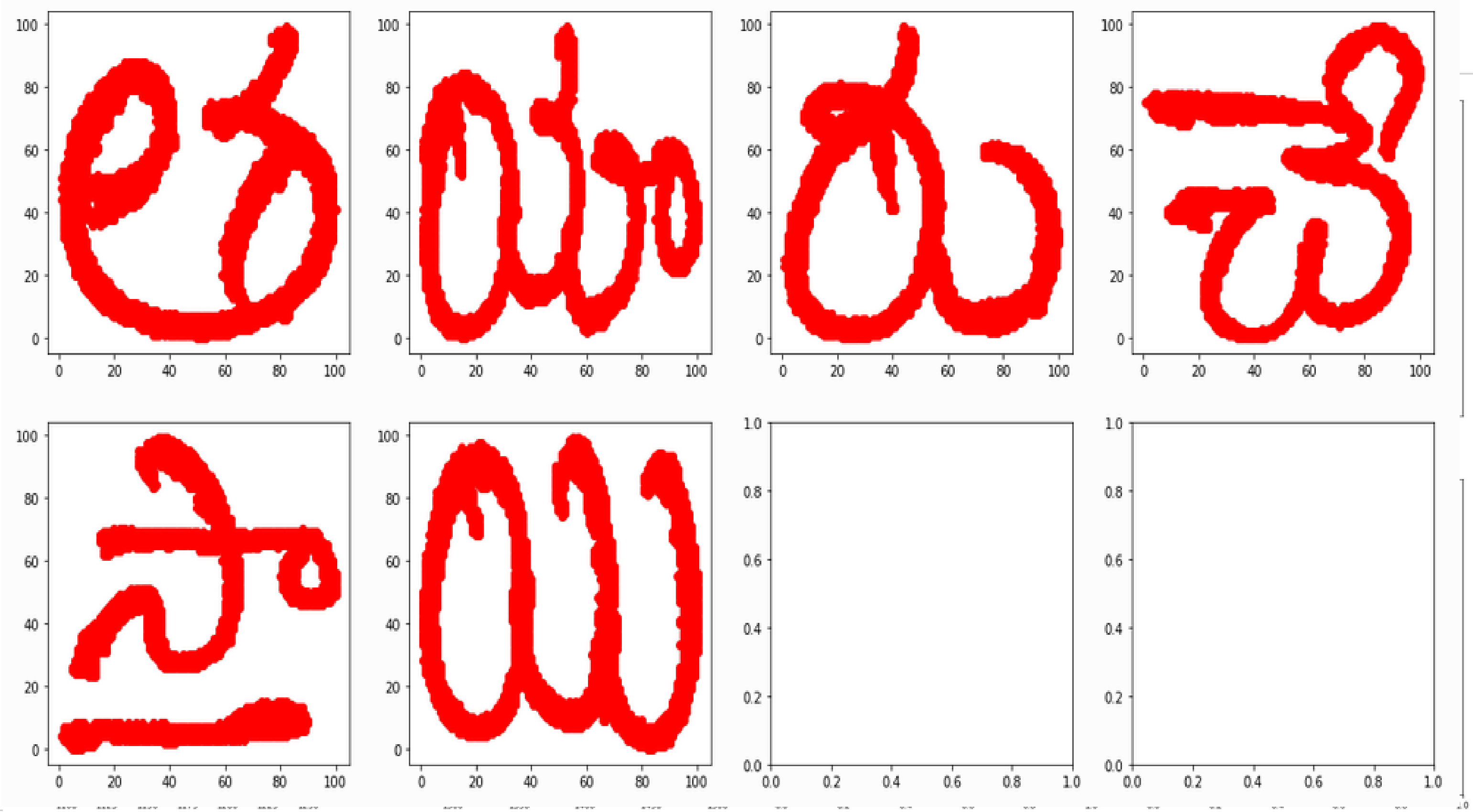




Mining Data (EDA)

- DB-Scan is used to find clusters and thus the characters of the word given
- Remove small clusters
- Combine or Split clusters as required
- Sort the characters as they are written in image
- Standardize characters

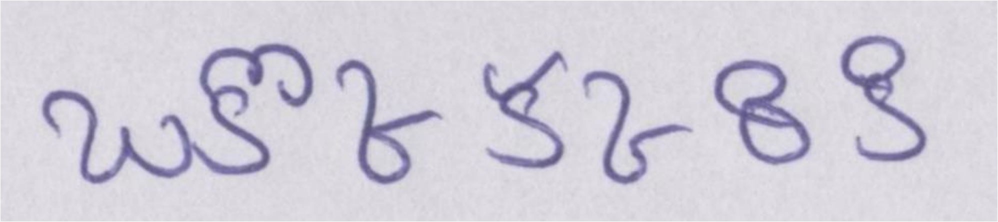




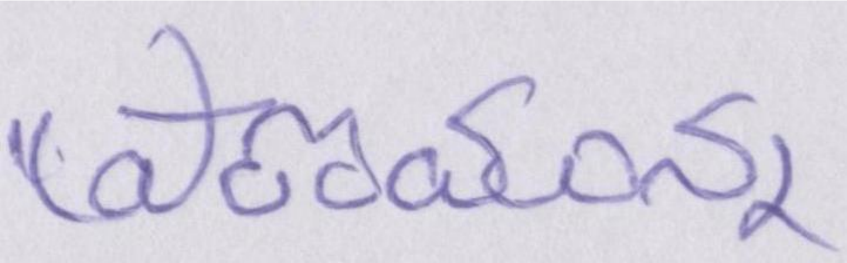
Challenges in EDA

- DB Scan (Clustering)

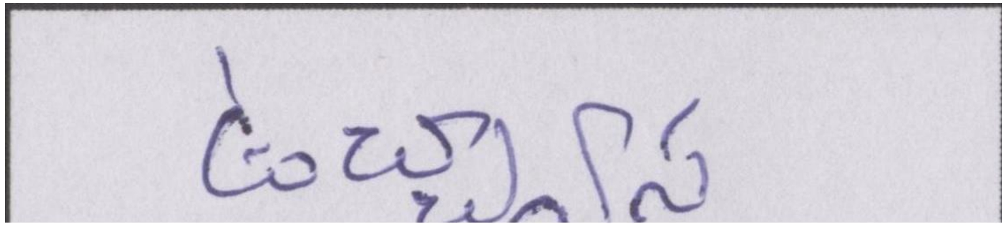
- Single character split into multiple pieces



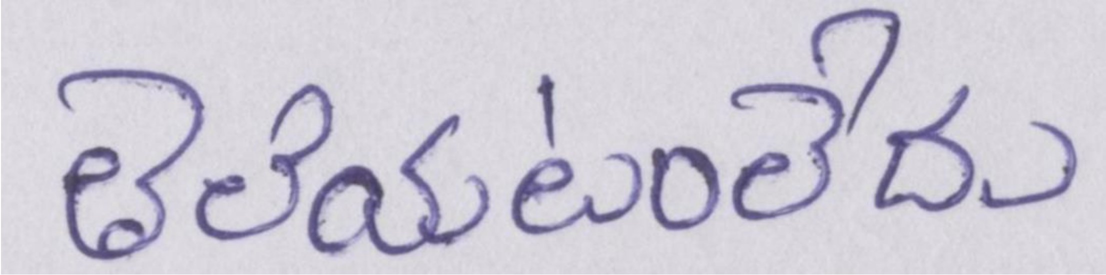
- Two or more characters are clubbed very closely



- Noise in the image



- Simple images but complex due language specificities

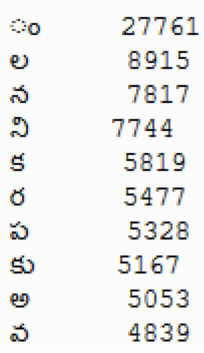


Some facts (EDA)

- Number of images processed: 80,692
- Number of images those are able to split correctly and able to use as data points: 70,803 (87.74%)
- Number of data points: 331,867 (Avg. 4.69 characters)
- Number of classes: 1326
- Size of the source images: ~3.7 GB
- Size of the files after binarization of the images: ~12.6 GB
- Estimated size of the Data Frame in memory: ~50.45 GB
- Estimated size of the Sparse Matrix in memory: ~34 GB (~66%)

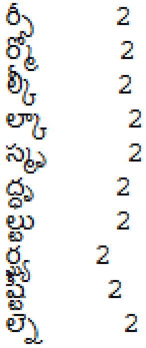
Somemore facts (EDA)

- Classes with highest number of observations:



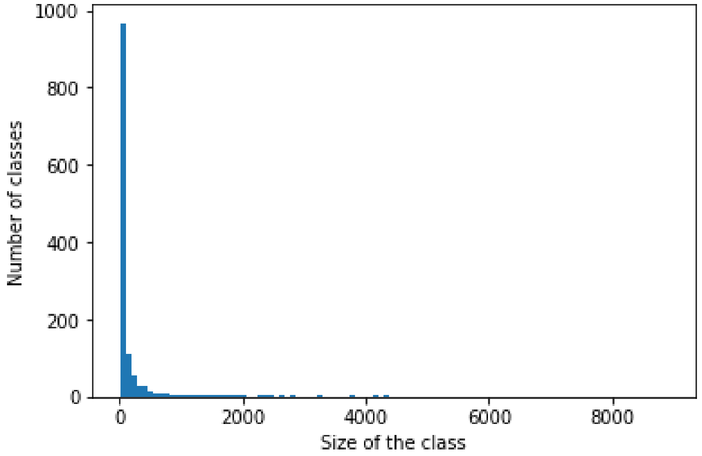
```
ం        27761
ల         8915
న         7817
ని       7744
క         5819
ర         5477
ప         5328
కు       5167
అ         5053
వ         4839
```

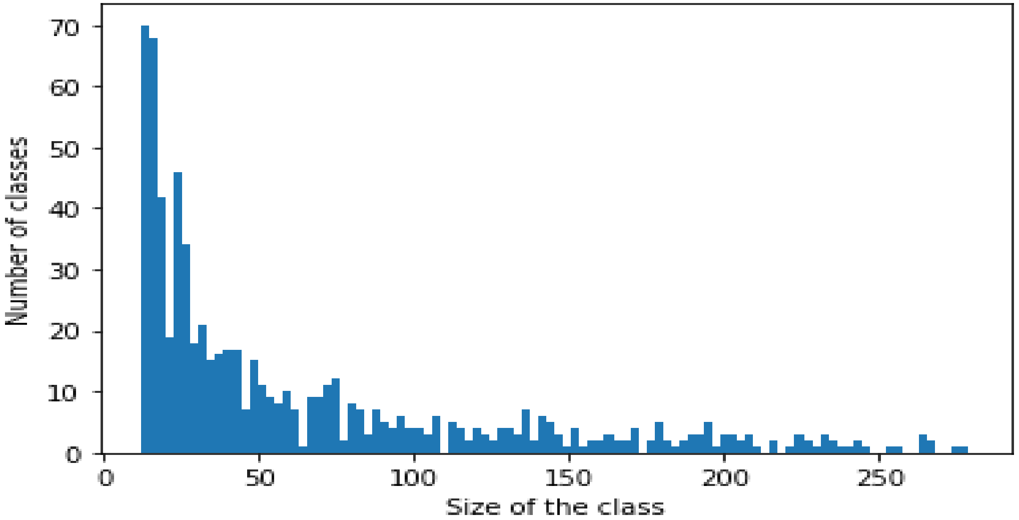
- Classes with least number of observations:



```
ర్వై       2
ర్శ్లే       2
త్రు       2
ల్టు        2
స్తృ        2
థ్ఖ        2
లఋ       2
ర్ఘ      2
ఓఋ      2
ల్జ        2
```

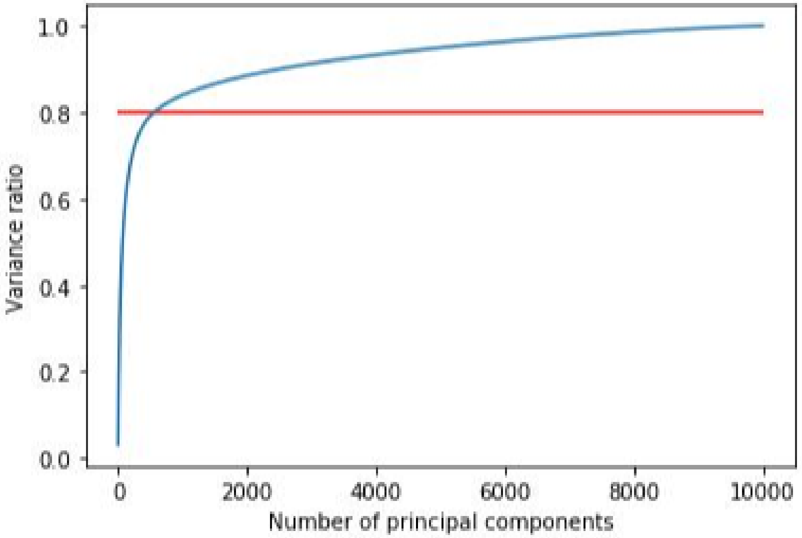
- Distribution of class size:

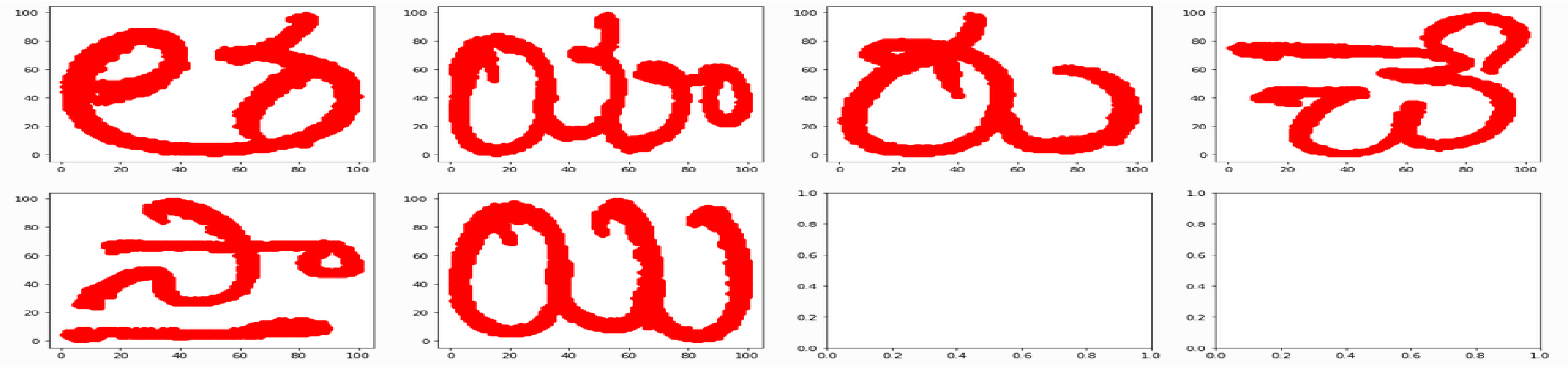




Pre-processing of data before modelling

- Dimensionality reduction
  - Method used: Incremental PCA in association with sparse matrix.
- 500 components are used for modelling capturing 80% of variance
- Standardization, shifted all the resultant character into fixed 100X100 pixels





Modelling

- Model used: Logistic regression with regularisation
- Baseline score : 0.08365098066394068
- Best score (Accuracy): 0.574984180552626 (Mean CV)
- Failed models to beat the score: Decision trees, Bagging with Decision trees, Random forests
- Failed to complete: Bagging with Logistic regression, Tensorflow

Key findings during project

- Handwritten character recognition is one of the complex issue and there is lot of scope to improve as the current models available are not generalized enough.
- Providing more servers for larger training data is not a default solution.
- Assessment of the memory, disk space and CPU requirements are essential in working with larger volumes.
- Coding standards also play major role as creating an additional object will take up double the memory.
- Do not involve target variable to improve the quality of the predictors

Future work

- Prediction inconsistencies can be analysed to find exactly where the model is failing which may help to identify additional features required to improve the score
- Additional layer of modelling can be done at word level to predict correct word even when some of the characters are predicted incorrectly
- Thickness of the letters can be minimized to bring down the size of the training data
- Generalize the cleansing process to be able to process even more patterns of hand writing