User Profiling in Social Media

Apoorva Balasubramanian & Swetha Chinthalapati

Contents

1 Introduction:

As more and more users are creating their own content on the web, there is a growing interest to mine this data for use in personalized information access services, recommender systems, tailored advertisements, and other applications that can benefit from personalization.

Research in psychology has suggested that behavior and preferences of individuals can be explained to a great extent by age, gender and underlying psychological constructs (or so called personality traits).

1.1 Goal:

The goal of this project is to build a system for automatic recognition of the age, gender, and personality of Facebook users. When given as input the status updates, profile picture and "likes" of a Facebook user, this system should return as output the age, gender and personality trait scores of that user.

1.2 Methods and Results

We perform supervised learning on the data that is provided by training them. We have used various Machine learning techniques like Naïve Bayes (Likes information), KNN Classifier(Images) & Random Forest(Status-text) to classify the inputs and predict the labels for new/test data.

The results have been sound as we have found a probability of 60-70% for gender and age and an average RMSE on the prediction of personality traits (Openness, Conscientiousness, Extroversion, Agreeableness, Emotional Stability).

Label	Prediction	RMSE (Root Mean Square Error)
Age	59%	
Gender	68%	
Open		0.65
Neurotic		0.79
Extrovert		0.79
Agreeable		0.65
Conscientious		0.72

2 Methodology:

In this section we will look into the overall architecture and a high level glance at the methodologies used to create the system.

2.1 High Level Work Flow:

As this is a supervised learning, we are provided with existing dataset with labels based on which we will train the data to identify the relationship between features provided out labels. As part of the training data the Profile-pictures (Image files), Status updates (as text files) and page like information (Facebook PageID) along with their labels i.e. *Gender*(male/female), *Age range* (18-24, 24-35, 36-49, 50-xx) and five different personality traits – *Openness to experience, Conscientiousness, Extroversion, Agreeableness, and Emotional Stability on a scale of 1 to 5*, is provided to the classifiers which is then trained using different classifiers which will store the pattern and mapping. We then pass the test data i.e. Profile pics, status updates and Page likes info to the classifiers which will return the labels based on the earlier learnings.

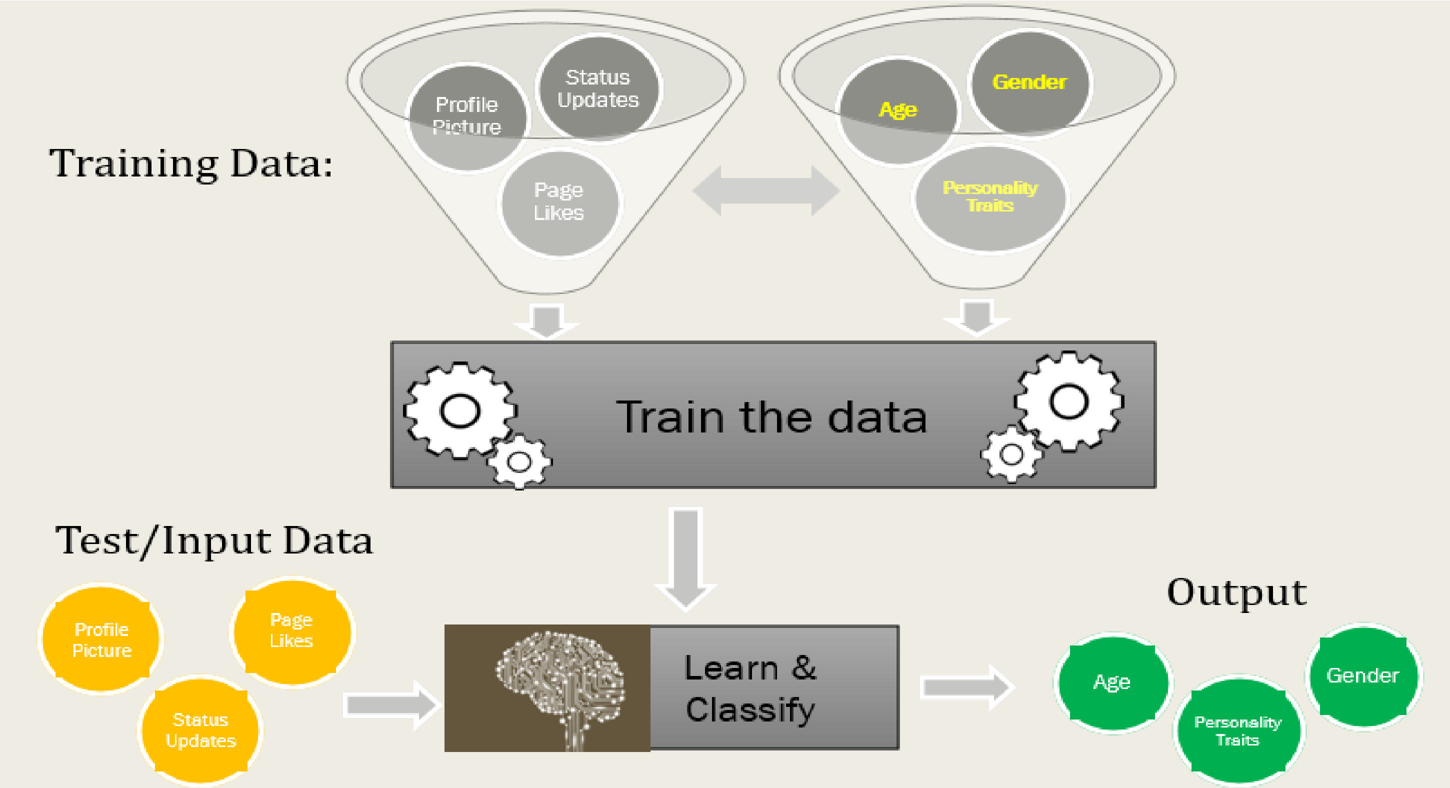
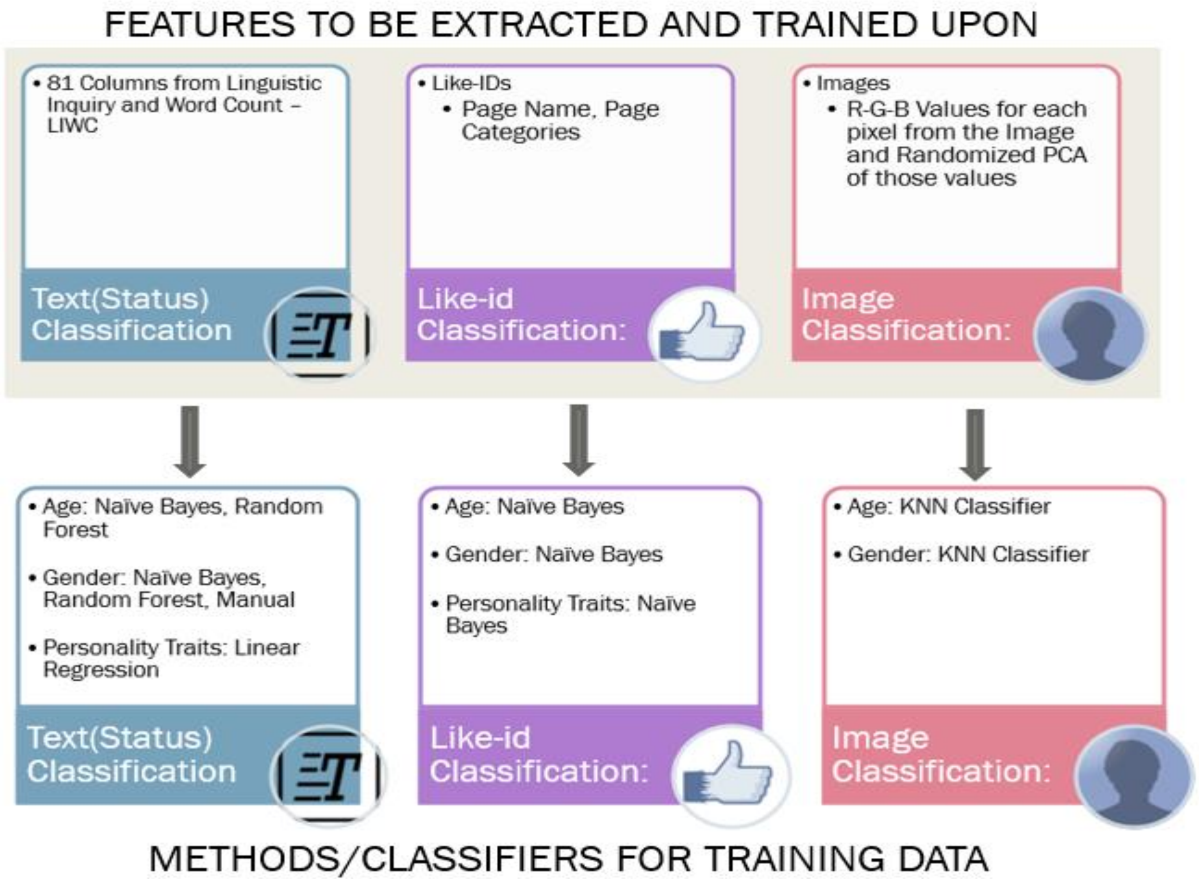


figure 1.1

2.2 Feature extraction and Methodologies:

We work on each input independently i.e. for each input type we train and classify and predict the labels. Following are features that are extracted and trained:



FEATURES TO BE EXTRACTED AND TRAINED UPON

- 81 Columns from Linguistic Inquiry and Word Count – LIWC

Text(Status) Classification

- Like-IDs
  - Page Name, Page Categories

Like-id Classification:

- Images
  - R-G-B Values for each pixel from the Image and Randomized PCA of those values

Image Classification:

- Age: Naïve Bayes, Random Forest
- Gender: Naïve Bayes, Random Forest, Manual
- Personality Traits: Linear Regression

Text(Status) Classification

- Age: Naïve Bayes
- Gender: Naïve Bayes
- Personality Traits: Naïve Bayes

Like-id Classification:

- Age: KNN Classifier
- Gender: KNN Classifier

Image Classification:

METHODS/CLASSIFIERS FOR TRAINING DATA

2.2.1 Status (Text) Classification:

The main goal of text classification is to classify the given text into a fixed number of predefines categories. With this module, we are building a system that will take status messages of each Facebook user as input and predicts age, gender and personality traits of the user and outputs an XML with all the predicted values.

- All the status updates were given as input to the LIWC (Linguistic Inquiry and word count) program for language analysis to generate a combined dictionary with basic counts and ratios with around 83 columns.
- This LIWC dictionary is used as input for age, gender and personality traits classification.
- Age and Gender Classification:
  - We are using Random Forest classifier to predict age and gender.
  - 81 columns from LIWC and the values of age and gender for each user are given as input to the Random Forest classifier from Scikit-Learn to train the classifier.
  - By using the trained classifier, age and gender for the test users are being predicted.

- Personality Traits:
    - For the prediction of personality traits, we are using Linear regression from Scikit-Learn.
    - We are giving 81 columns from LIWC and the personality traits values for each user as input to the linear regression to train the classifier.
    - By using the trained classifier, personality traits for test users are being predicted.

2.2.2 Image Classification:

Images are complex non-structural data that have a host of features based on which we can classify the data. We use the following methodology to train and classify the data:

- Prepare the data to map Image and corresponding gender and image and corresponding age range.
- We loop through each image and for each image:
    - Standardize the image to a particular size, say 100x100, this means we will have 10000 pixels for the image.
    - For each pixel extract the RGB values and store all of them in an nx3 array (where n= no. of pixels, 3 is (R, G, B) value)
    - All these values are stored in a higher dimensional *np array* which will store the *RGB* values for all the pixels of all images.
    - In case the image is black & white then we store the intensity of the pixels across R, G, B which will be 255.
- Now that we have the np array that has all the RGB values of all pixels of images, this becomes a huge dataset to be classified. Hence we use *RandomizedPCA* to summarize the dataset.
- We bring down the dimensionality of each image (which was 10000x3=3000) to 5.
- We run *a K- Nearest Neighbor* classifier to classify the summarized data with corresponding labels. We have used the no. of components=10.
- We can now start predicting the output using this classifier.

2.2.3 Like-ID (Text) Classification:

We are provided with one-to-many mapping of each user-ID and all the facebook page IDs (of pages) that user has liked. We use the Facebook's APIs to extract the required information about the page like page name and category. We use these features and train the data further using Naïve Bayes classifier. We run the classifier against all the labels(age, gender and personality traits separately).

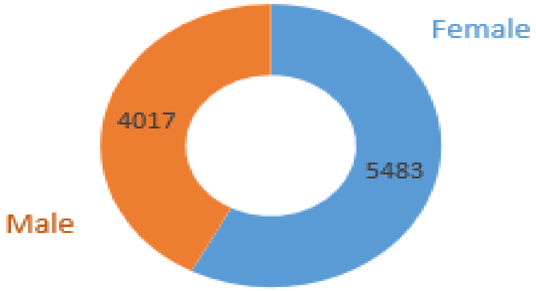
3 Analyzing the Dataset

It is very important to deeply analyze the training data provided. The training data should be diversified and have enough records in each bucket which will eventually lead to more efficient predictions. For e.g. we have majority of data skewed under a particular age-bucket, more data will be classified under that bucket.

Total records available as training: 9500

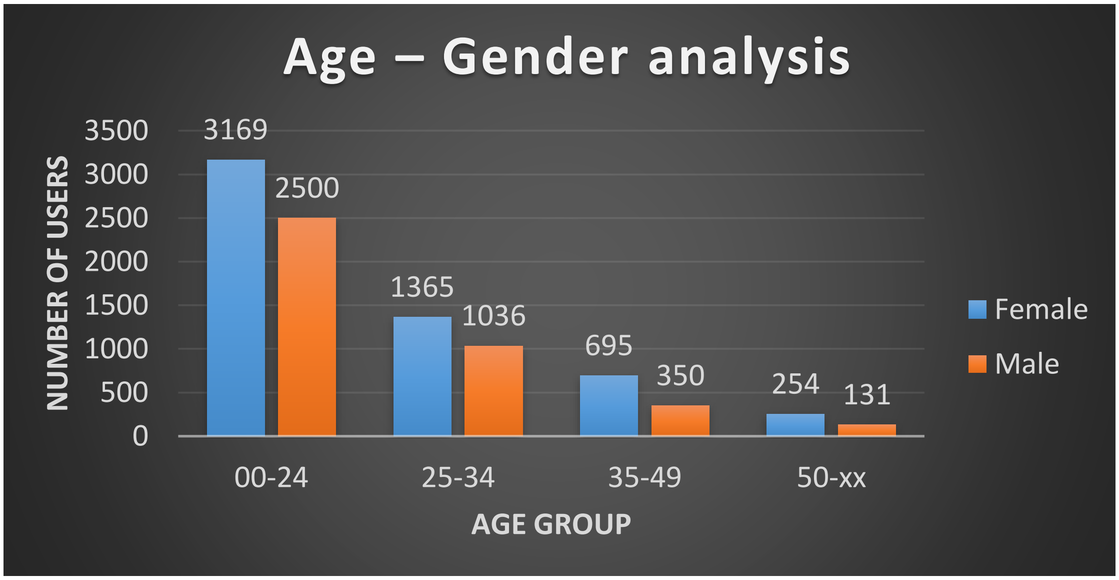
3.1 *Gender breakdown*:

Input data is equally diversified among Male and female users with a slight lenience towards female users.



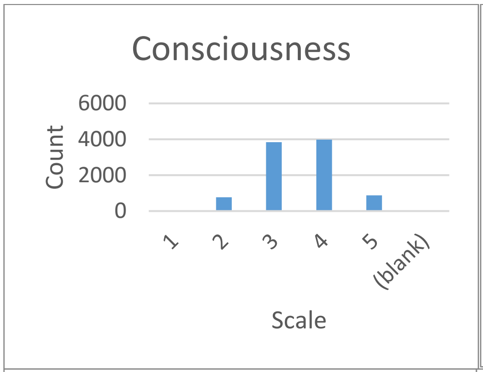
3.2 *Age-Gender analysis:*

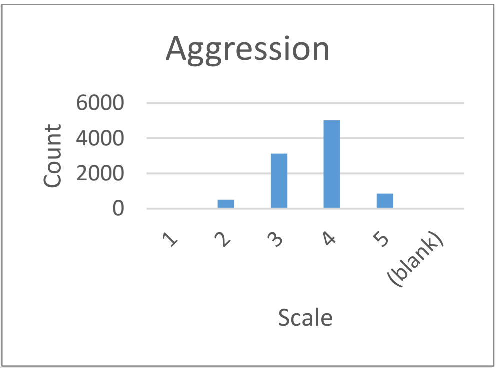
We can observe that the training data has majority of users under the age of 30 and each age-group has data equally distributed between Male/Female users.

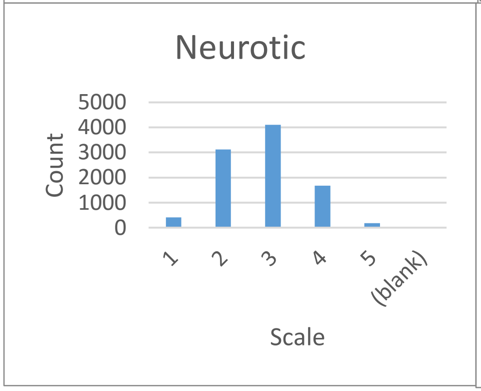


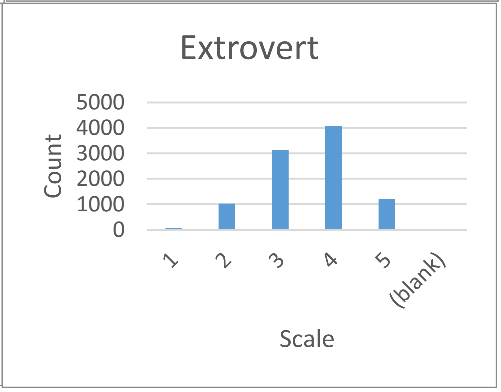
3.3 *Personality Traits drilldown and grouping analysis by gender:*

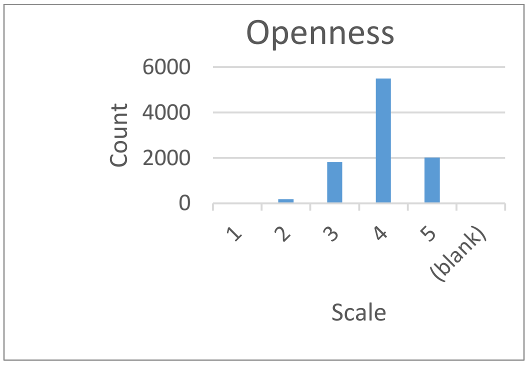
We perform a linear drill down of all of users' data for each of the personality traits and notice that the data is skewed for each of the personality trait and an overall mean for the traits is around 3 and 4.



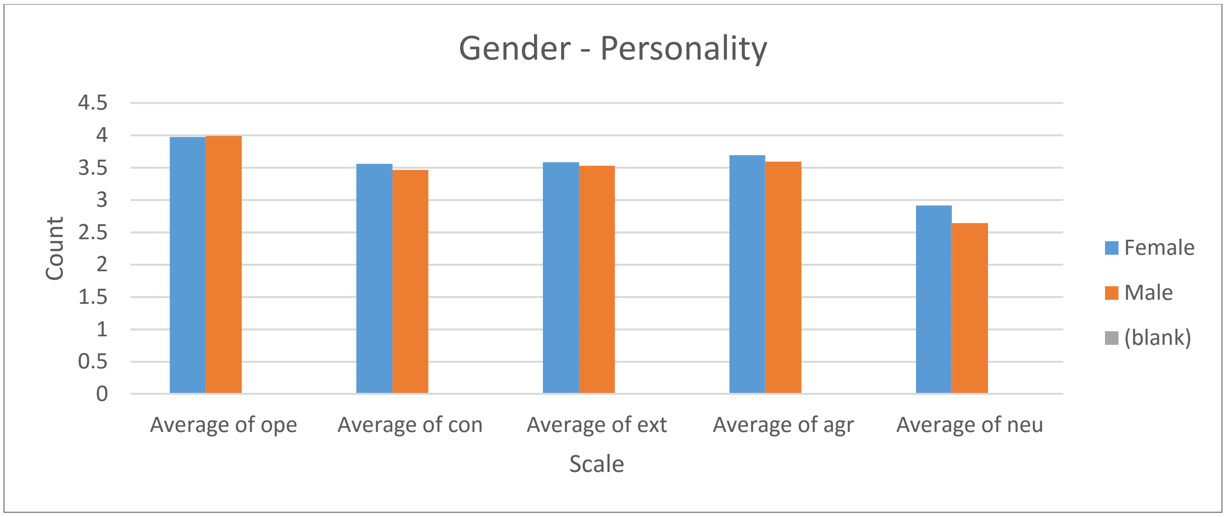








We also observe perform a quick grouping of mean of each personality traits of male and female users and analyze numbers. There are no major surprises here and the data is evenly distributed between male and female users.



Note: We have done a ROUND() of the personality traits data. E.g. 3.6 is considered as 4.

4 Results

Following are the results that were procured after testing the classified data for each of the parameters.
We have tested against below test data

- Data split from the original training data of 9500 records. (70%Training, 30 % test)
- Public test data of around 334 records.
- Hidden test data with 1000 records.

4.1 Overview of results

The overall results have been sound despite the challenges faced (memory, lack of diversification of data and wrong/bad data).

Below are the overall results with hidden test data on VM.

	Age	Gender	Open	Neurotic	Extrovert	Agreeable	Conscientious
Baseline	59%	59%	0.65	0.80	0.79	0.66	0.73
Text Classification Results	59%	68%	0.65	0.79	0.79	0.65	0.72
Like-id Classification	51%	59%	0.70	0.90	0.90	0.75	0.90
Image Classification	55%	56%	-	-	-	-	-

The base results show that we are getting a 60% to 70% accuracy for each of the predictions are as following:

4.2 Text Classification

We predicted the Age and Gender for the users by using Random forest classifier and Naïve Bayes classifier. Input for the Random Forest classifier are 3 to 83 columns from LIWC dictionary. Input for the Naïve Bayes classifier are the status updates from each user.

Below are the results from Age and Gender classification using status updates.

	Age (Accuracy)	Gender (Accuracy)
Baseline	59%	59%
NaïveBayes – Test Data	52%	55%
Random Forest – Test Data	61%	68%
Hidden Test Data	59%	68%

For the prediction of personality traits, Linear Regression classifier is being used with input as 3-81 columns from LIWC dictionary. Below are the results for the Personality Traits prediction.

	Open	**Neurotic**	**Extrovert**	**Agreeable**	**Conscientious**
Baseline	0.65	0.80	0.79	0.66	0.73
Test data (With Gender)	0.63	0.72	0.80	0.65	0.78
Test data (Without Gender)	0.67	0.71	0.80	0.67	0.82
Hidden Data (Without Gender)	0.65	0.79	0.79	0.65	0.72

Improvements/ other considerations:

- Multi label classification for age as four categories using Naïve Bayes classifier gave us very less accuracy. Both the Naïve Bayes method and manual construction of Naïve Bayes classifier didn't give us satisfying results. So we had to move to Random Forest classifier.
- Random Forest classifier is very robust to overfitting and gave us better results than Naïve Bayes classifier.
- Age classification's accuracy is still near baseline so if multiple classifiers and ensemble are used, they might give us better results.
- As future work in text classification, usage of multiple classifiers to classify the data and then by using ensemble methods, the overall accuracy of text classification to predict age and gender can be improved.

4.3 Like-ID Classification

We have derived the Page information by running text classification using Naïve Bayes on the Page Information that user has liked and page category. Due to memory constraints we could train the data for 2000 records.

The data was concatenated for each user-ID and the corresponding prior to running the classifier

	Age	Gender	Open	Neurotic	Extrovert	Agreeable	Conscientious
Baseline	56%	54%	0.65	0.80	0.79	0.66	0.73
Iteration 1 (Random training data)	10%	58%	0.75	1.45	0.79	0.68	0.85
Iteration 2 K-Fold (75%)	59%	59%	0.65	0.80	0.79	0.66	0.73
Iteration 3 K-Fold (50%)	59%	59%	0.65	0.80	0.79	0.66	0.73

There are other methodologies that were considered and will be looked upon in future work to improve the numbers:

Improvements/future enhancements/other considerations:

- Not to concatenate the data and treat each record separately.
- Using a Random Forest classifier will also be looked upon to classify the derived text data.

- Use Bag of words classifier
- Use two or more classifiers and using of ensemble method to enhance the results.

4.4 Image Classification

Image Classification has been done by extracting RGB features of each pixels of images, taking a Randomized PCA of the vast data and running a KNN classifier.

	Age	**Gender**
Baseline	45%	56%
Iteration 1 (Random training data)	38%	48%
Iteration 2 K-Fold (75%) and RandomizedPCA, n=6; KNN (components=10)	45%	56%
Iteration 3 K-Fold (75%) and RandomizedPCA, n=10; KNN (components=10)	45%	56%

Challenges: Following were some of the challenges that had to be considered:

- ✓ Handling Black/white images.
- ✓ Rich features and memory allocation issues.
- ✓ Bad data:
  Ex: there are datapoints in traing dataset that are labelled with age=109 but that doesn't seem to be the case when we check the picture manually:

UserID : 8508ded7468dcfee0eaf560530792f8f *Labeled as age=109*

- ✓ Presence of multiple people in the same picture.

Improvements/Other methods that were considered:

- ✓ Using visual bag of words is another efficient way to improve the performance as we can extract features using SIFT methodology and store the training data in secondary memory.

- ✓ Usage of Local Binary Pattern Histogram: this will help in improving the numbers and also avoid the challenge of handling B/W images as we need to convert the images to greyscale first. However, the internal memory used is very high and could not train more than 100 images at one go.

5 Conclusion and Future work:

We have received a satisfactory results from Text classification, Like-Id classification and Image classification modules of our project. The same process of text classification and image classification can be used in many other real world scenarios. This entire project was implemented using Python language. Other technologies like R, AzureML Studio can also be used in achieving the same results.