NLP - A Love Story

16:954:577:01 Statistical Software Project – Fall 2021

Team 3:

Jeffery Dean
Mohit Agarwal

Introduction

- **Motivation**:
  - What makes two people good match for one another?
  - What factors determine accurate and relevant classification on such platforms?
  - Could profile descriptions “Bio” help classify users better?

- **NLP Problem**:
  - Apply series of NLP ML models on “**biography**” essays to predict attributes of interest about a person. For ex: ‘**Age**’, ‘**Education level**’, ‘**Job_group**’ and ‘**drinking habits**’ taken as representative labels in this project.

- **Solution Methodology**:
  - Primary objective can be described as NLP Classification problem.
  - **Classification problem** : Multinomial (Age, Education and Job) and Binomial for drinking.
  - As an additional objective, **Clustering technique** also implemented to segregate similar user profiles (Un-Supervised ML)

Data Acquisition

- Data was obtained from Kaggle, CSV file (in total **59, 947 user profile data**). These user bio essays were from a popular dating site "***Ok-Cupid***".

- **Total 22 Attributes** were present in the original file. For our experiment, **4 attributes were identified**: Age, Education_group, Job_group and drinking_frequency. The CSV file was edited into an Excel file

- Batch size for the project was finalized by experimentation. 59, 947 profiles reduced to **46,000 profiles** initially to weed out profiles with incomplete / No information in "Bio".

- Further batch size of **25000, 12000, 10000 and 8000** were tried to run models based on our system / PC capability.

- In development phase, **sample of 5000 profiles** were used.

- Finally, all models were scaled up to **10, 850 profile (split ratio 0.2 b/w training and test set).**

- This data set had **equitable distribution** from all ***age groups***.

Data Cleaning

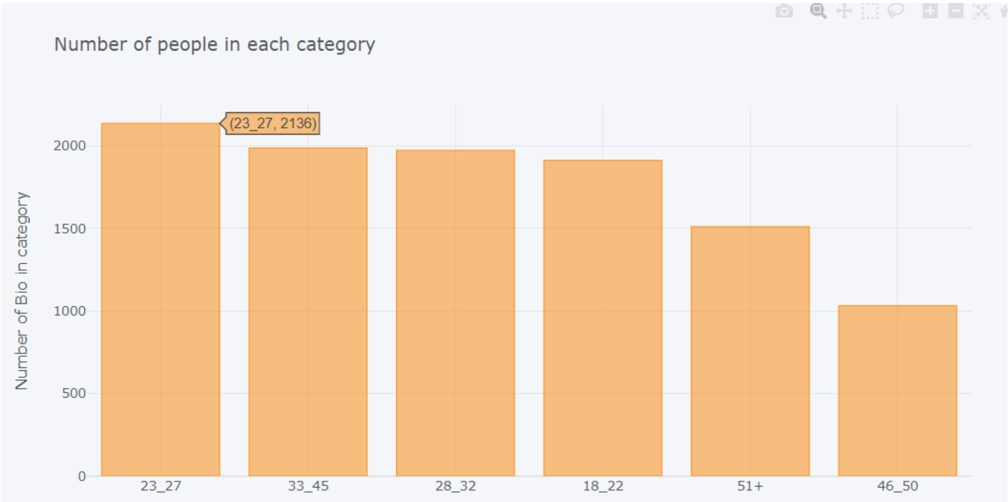
RUTGERS

- Biography **text consolidation** (merge text to reshape into essay).
- **Labels consolidation** and merger.
- Convert drinking habit to a **Boolean data** field to apply in LR and RNN, LSTM, BERT models.
- Remove entries in column of interest (Age, education, job and drinking) where **wrong** or **missing values** are found.
- **Regex cleaning**: remove symbols, mathematical operators and punctuation
- Convert all string to **lower.**
- Remove all **emoticons** from the biography passage.
- Remove **stop words**.
- **Tokenize** the data to obtain the **TF-IDF** matrix (for relevant models) and embeddings.
- Sample cleaned text:

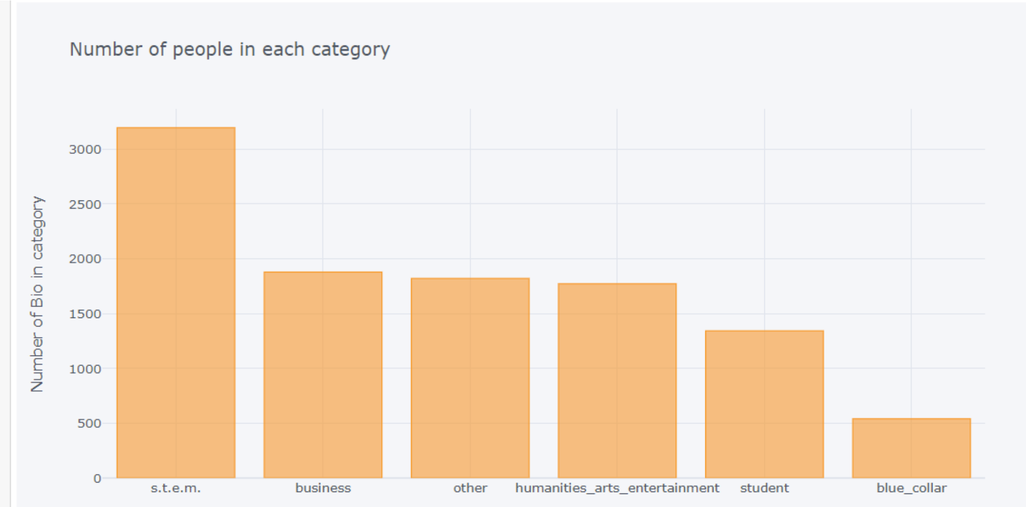

```
In [35]:
#Clean text Bio
bio_token_NB= pd.DataFrame(df_dating_NB, columns = ['Bio'])
print(bio_token_NB[0:5])
```

```
                                                 Bio
0  about me i would love to think that i was some...
1  my name is ashley and i live in san francisco ...
2  fulltime student fulltime square i change from...
3  apparently has become a new favorite word of m...
4  i grew up in iowa it gets a bad rap but let me...
```

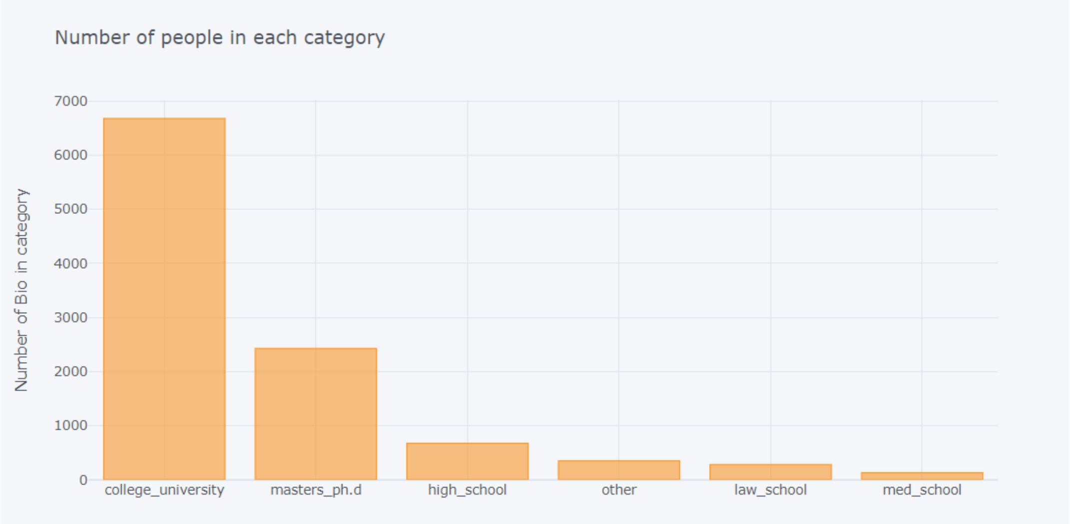
Distribution: Scaled up and refined dataset



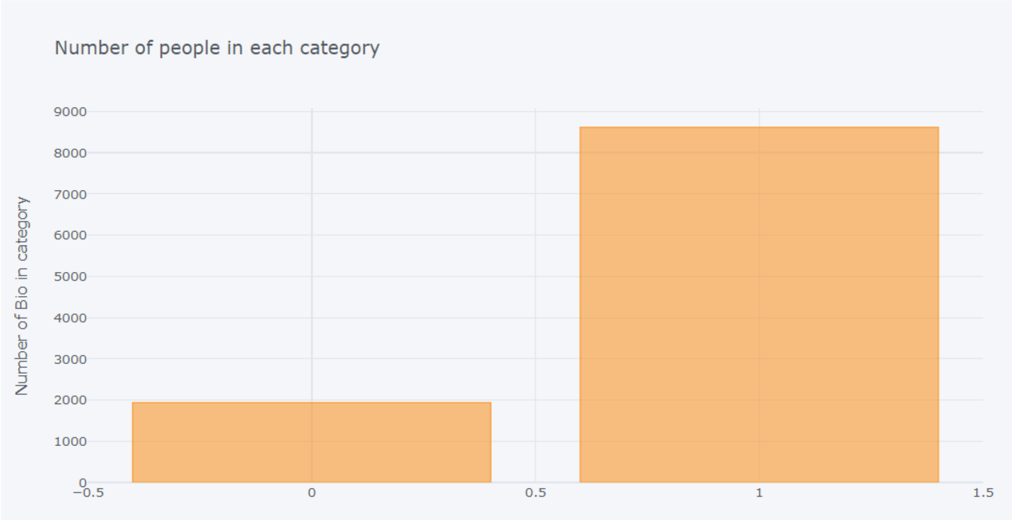
Age_group



Job_group



Education_group



Drinking_freq

Summary of Models Experimented in Project:

RUTGERS

- Baseline Models:

Model Name	Objective	Label(s) Predicted	Number of Cases (Code files)
Naïve –Bayes	Multinomial classification	Age, Education, Job, Drinking	4
Logistic Regression	Binomial Classification	Drinks_freq (0 or 1)	1

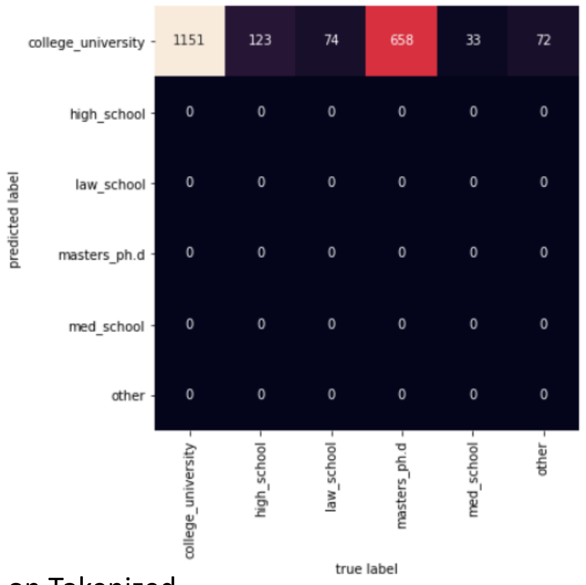
- Advanced Models:

Model Name	Objective	Label(s) Predicted	Number of Cases (Code files)
RNN with LSTM (dropout)	Multinomial classification	Age, Education, Job	3
RNN with LSTM (dropout)	Binomial Classification	Drinks_freq (0 or 1)	1
CNN (with varying architecture)	Multinomial classification	Age, Education	2
Distill-Bert	Binary classification	Drinks_freq (0 or 1)	1
LSTM (with recurrent dropout)	Binary classification	Drinks_freq (0 or 1)	1
K-means	Clustering	Unsupervised (K = 6)	1

Baseline Model, Type 1: Naïve – Bayes

Cleaned Excel file converted into **panda data frame** for running NLP models. PD made data reading corpus very easy and subsequent data formulation for importing into ML models. **Split Ratio** : 80-20 (train, test set). **Total number of dataset:** 10, 850. Train : 8440 and Test 2111. Out of 10, 850 rows approximately 300 eliminated since they had **null values**.

Education_group



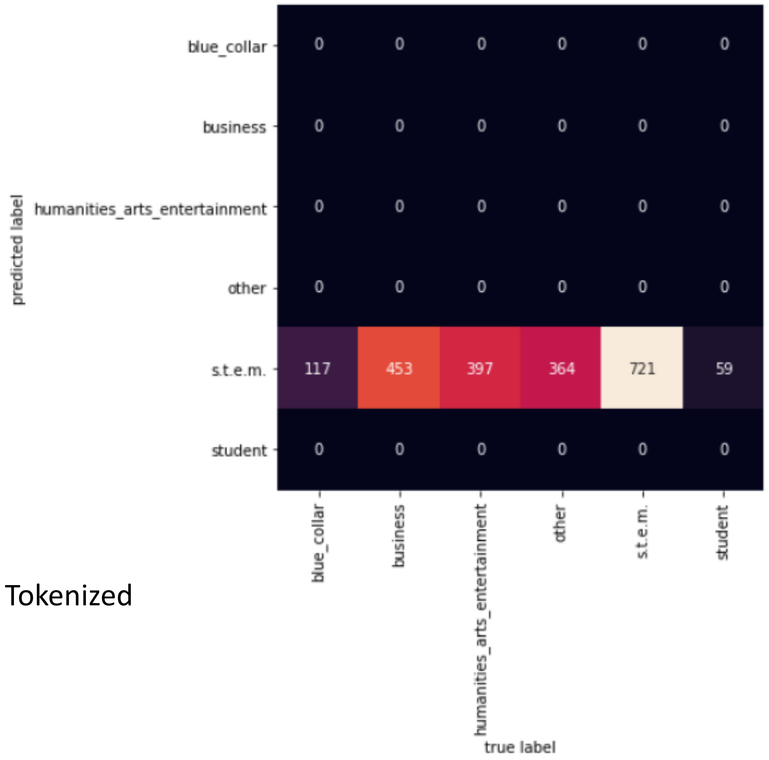
Method: TF-IDF approach was used on Tokenized text to make NB pipeline.

F-1 score (**micro**): 0.545
F-1 score (**macro**) : 0.117
F-1 score (**weighted**): 0.385

Observation:

NB as base model did not perform well it simply classified everything on the busiest group.

Job_group



Method: TF-IDF approach was used on Tokenized text to make NB pipeline.

F-1 score (**micro**): 0.3415
F-1 score (**macro**) : 0.0848
F-1 score (**weighted**): 0.1739

Observation:

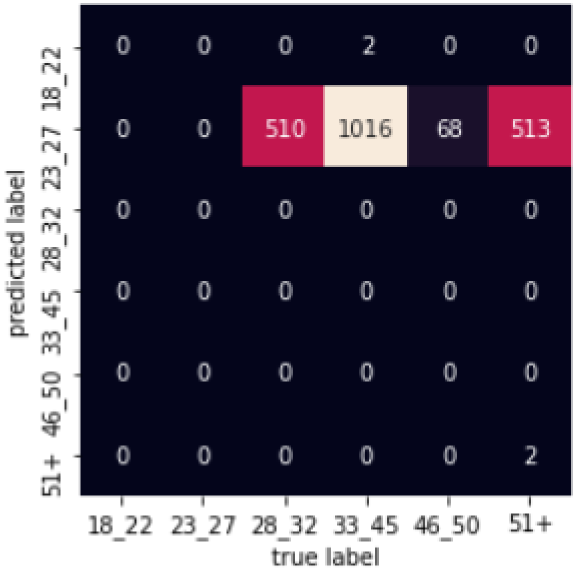
NB perform worse on classifying Job_group and surprisingly didn't catch on "student" category at all. Instead, it classified everything as "S.T.E.M."

Baseline Model, Type 1: Naïve – Bayes

RUTGERS

Cleaned Excel file converted into **panda data frame** for running NLP models. PD made data reading corpus very easy and subsequent data formulation for importing into ML models. **Split Ratio** : 80-20 (train, test set). **Total number of dataset:** 10, 850. Train : 8440 and Test 2111. Out of 10, 850 rows approximately 300 eliminated since they had **null values**.

Age_group

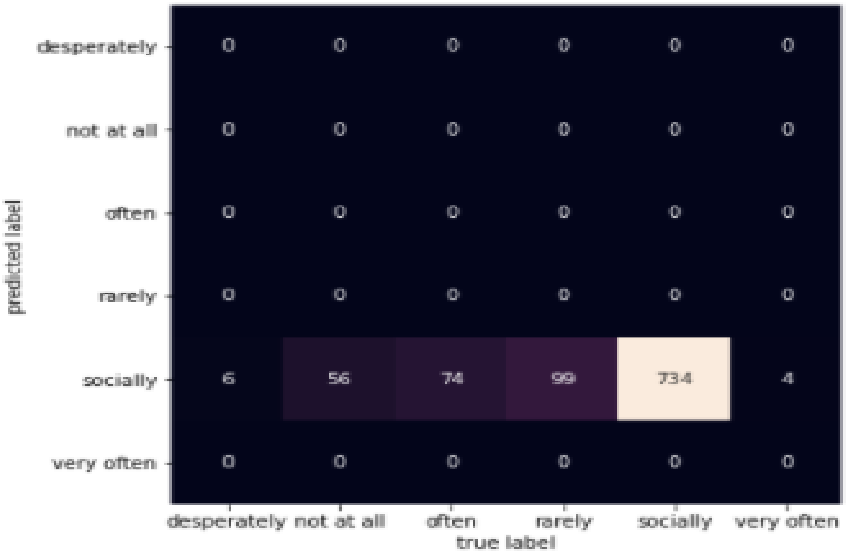


<u>**Method:**</u> TF-IDF approach was used on Tokenized text to make NB pipeline.

F-1 score (**micro**): 0.0009
F-1 score (**macro)** : 0.001
F-1 score (**weighted**): 0.00188

<u>**Observation:**</u>
NB model didn't classify age group at all. Clearly, the model didn't learn any context from people's essay and hence misclassified.

Drinking



<u>**Method:**</u> TF-IDF approach was used on Tokenized text to make NB pipeline.

F-1 score (**weighted**): 0.6487

<u>**Observation:**</u>
Heatmap skewed by uneven columns, particularly in social drinkers.

Baseline Model, Type 2: Logistic Regression

- As part of data cleaning process, Drinking habit attribute was modified from a multinomial attribute to binary attribute (0 or 1).
- All users who had indicated drinking as “never” or “rarely” classified as ‘0’ and other as ‘1’.

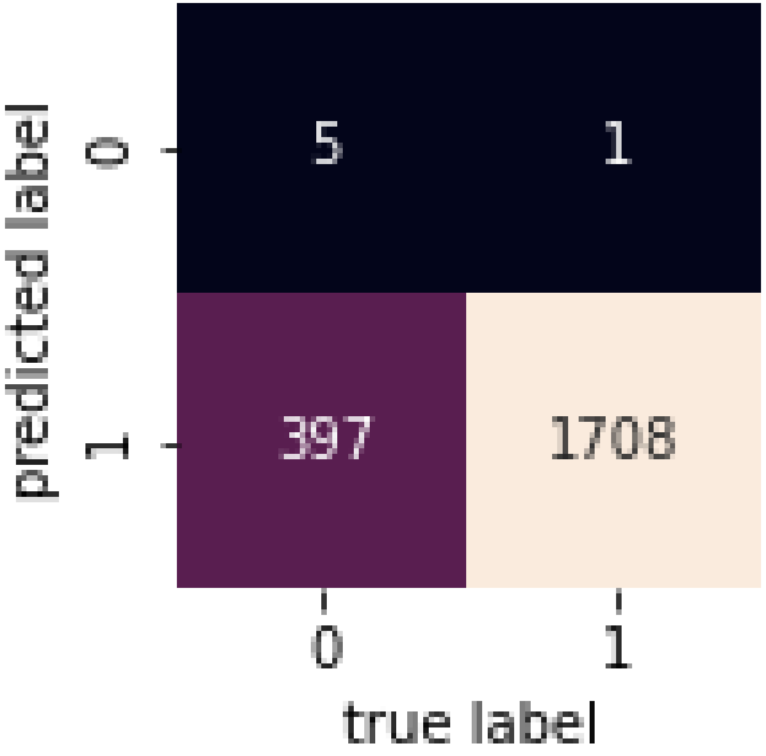
Method: TF-IDF approach was used on Tokenized text to make NB pipeline.

F-1 score (**micro**): 0.8114
F-1 score (**macro**) : 0.4600
F-1 score (**weighted**): 0.7297

Precision: 0.811
Recall: 0.999
Accuracy: 0.81

Observation:
- Logistic Regression predicted drinks_freq label reasonably well.
- This could be because “drinks_freq” due to label consolidation (binary).
- Drinking habits of most people in the dataset is ‘1’ compared to ‘0’ so that could be another reason behind high accuracy.

Drinks_freq



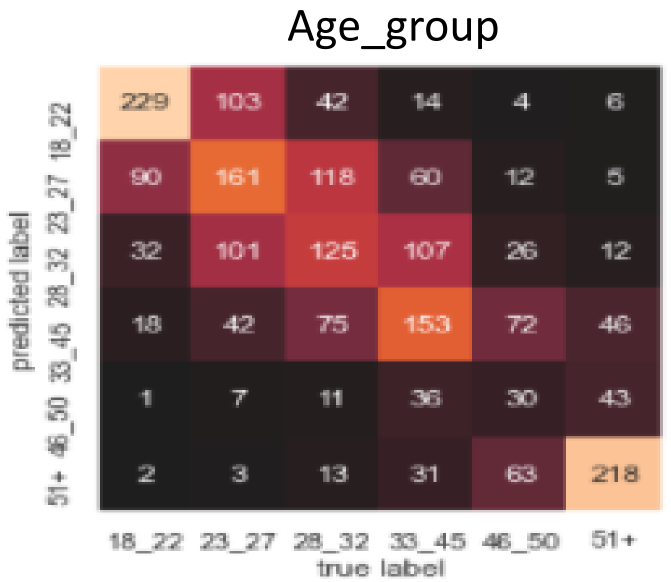
Note: Boot-strapping and K-fold cross validation can be tried as future improvement method on such a model.

Advanced Model #1 - CNN

Iteration 1 (1 layer, 100)

62 Iterations, Final Loss: 0.0069,

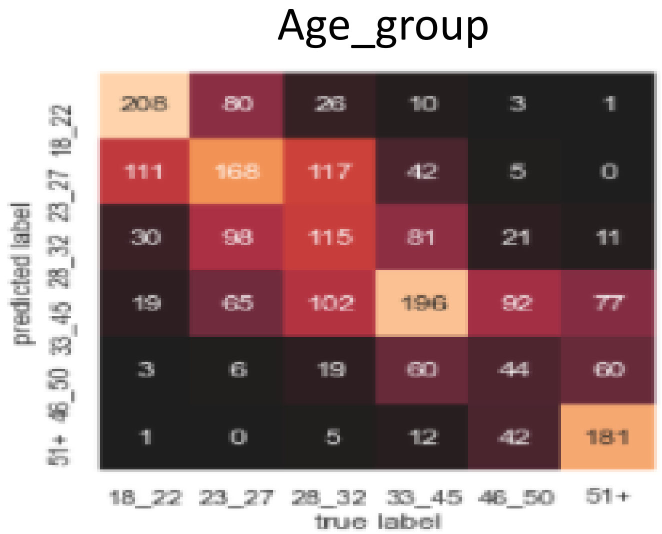
Weighted F1 Score: 0.4292



Iteration 2 (3 layers, 100)

25 Iterations, Final Loss: 0.00092,

Weighted F1 Score: 0.4374

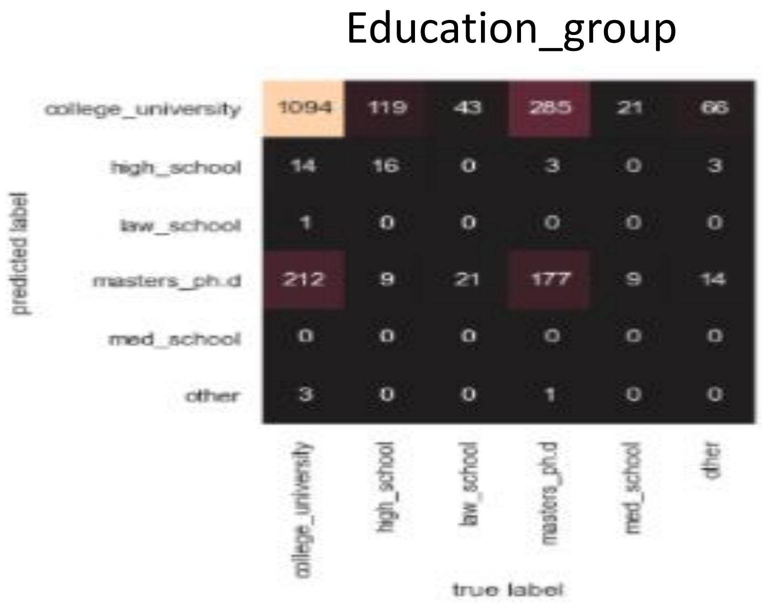


Advanced Model #1 - CNN

Iteration 1 (1 layer, 100)

57 Iterations, Final Loss: 0.0054,

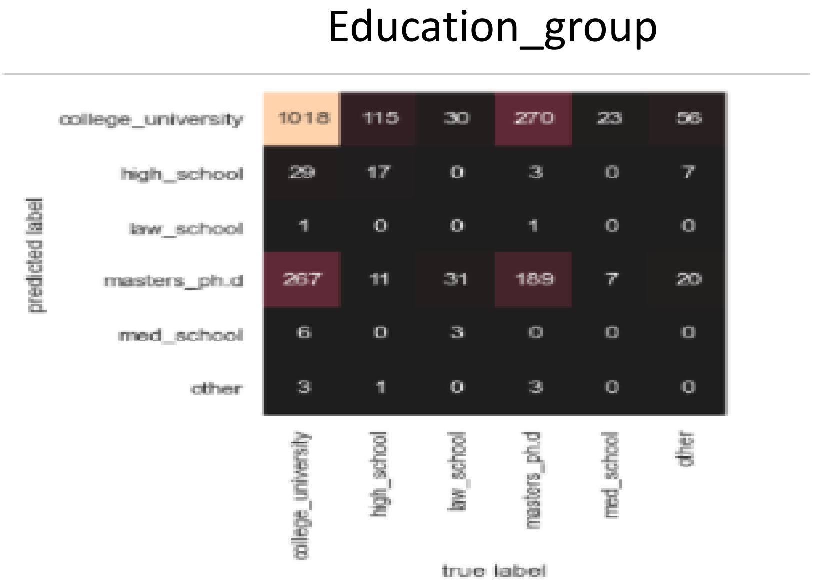
Weighted F1 Score: 0.5631



Iteration 2 (3 layers, 100)

26 Iterations, Final Loss: 0.00088,

Weighted F1 Score: 0.5461



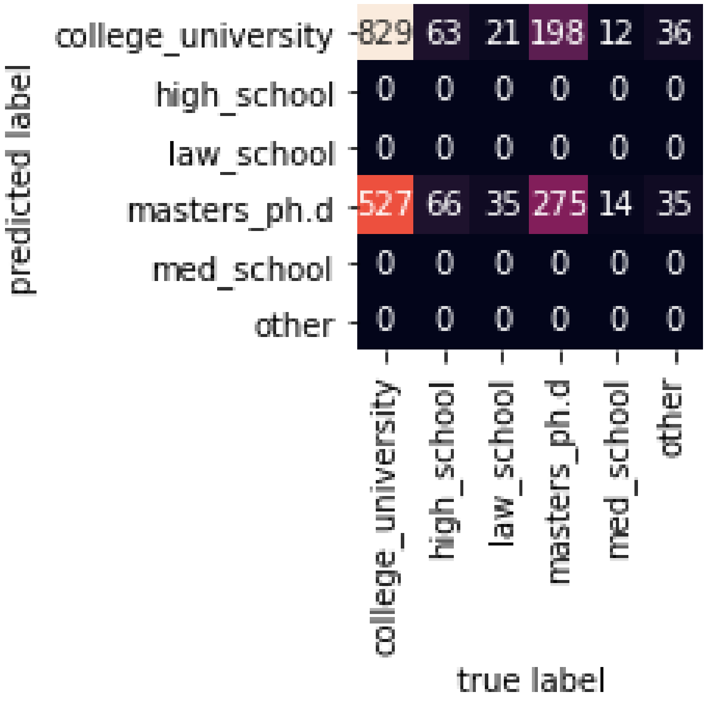
Advanced Model #2 – RNN: Case 1

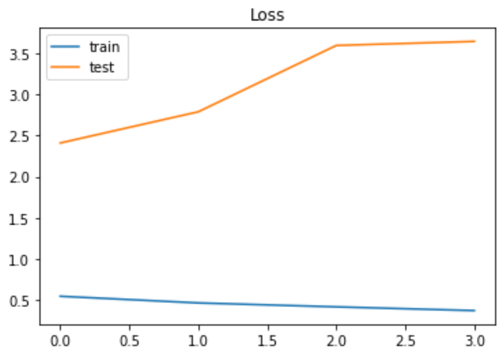
```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, None, 128)         11865728

 lstm (LSTM)                 (None, None, 25)          15400

 global_max_pooling1d (Globa  (None, 25)               0
 lMaxPooling1D)

 dropout (Dropout)           (None, 25)                0

 dense (Dense)               (None, 50)                1300

 dropout_1 (Dropout)         (None, 50)                0

 dense_1 (Dense)             (None, 50)                2550

 dropout_2 (Dropout)         (None, 50)                0

 dense_2 (Dense)             (None, 6)                 306

=================================================================
Total params: 11,885,284
Trainable params: 11,885,284
Non-trainable params: 0
_________________________________________________________________
```

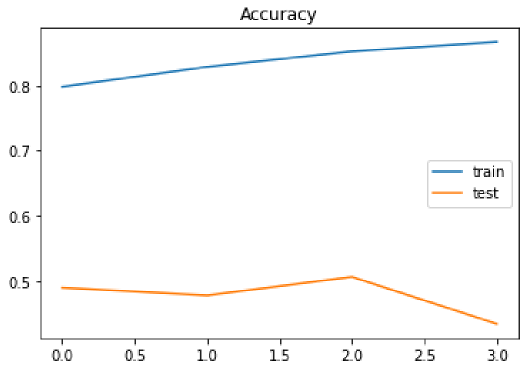
Observations:

- **Overall accuracy**: **0.5229**
- **Training error** (MSE): 0.0328
- **Validation error** (MSE): 3.8925

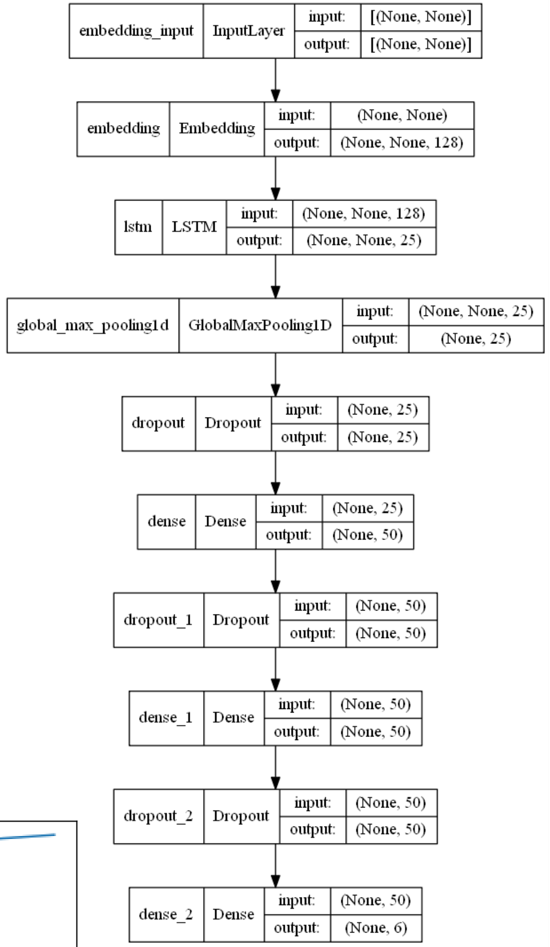
Confusion Matrix







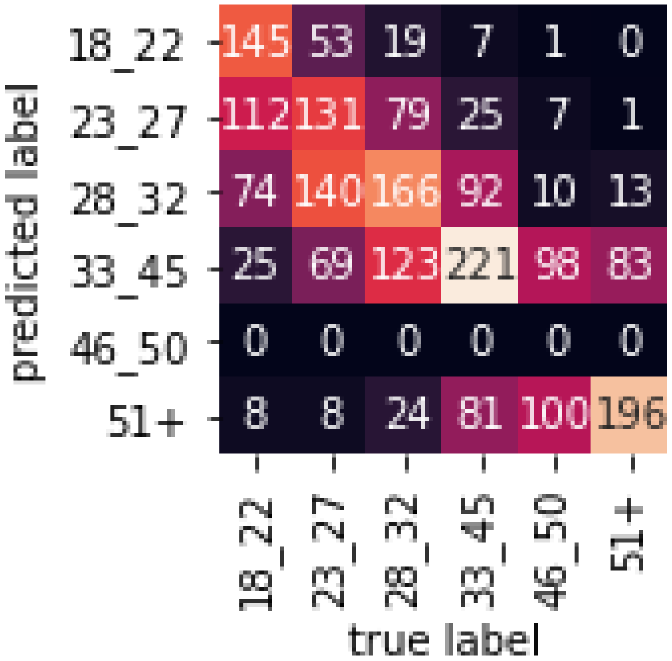
Education_group



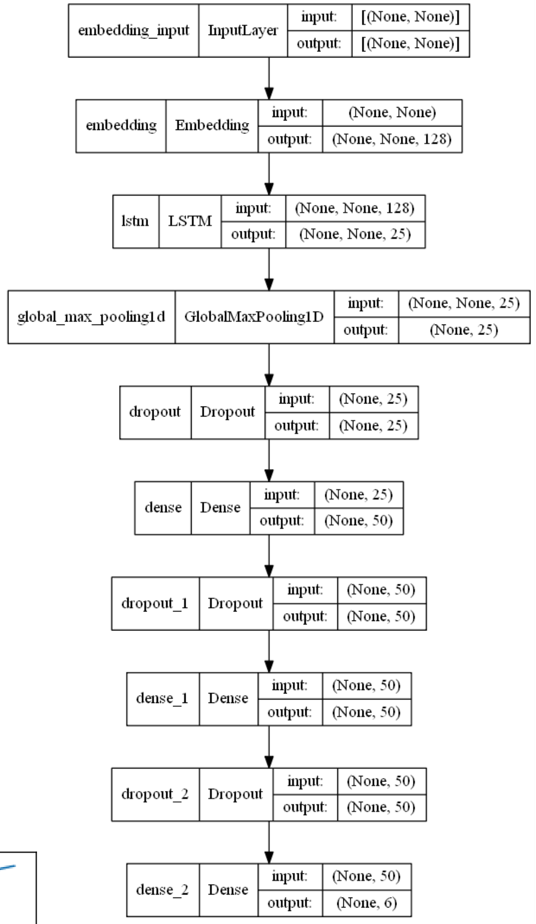
Advanced Model #2 – RNN: Case 2

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, None, 128)         11865728

 lstm (LSTM)                 (None, None, 25)          15400

 global_max_pooling1d (Globa  (None, 25)               0
 lMaxPooling1D)

 dropout (Dropout)           (None, 25)                0

 dense (Dense)               (None, 50)                1300

 dropout_1 (Dropout)         (None, 50)                0

 dense_1 (Dense)             (None, 50)                2550

 dropout_2 (Dropout)         (None, 50)                0

 dense_2 (Dense)             (None, 6)                 306

=================================================================
Total params: 11,885,284
Trainable params: 11,885,284
Non-trainable params: 0
_________________________________________________________________
```

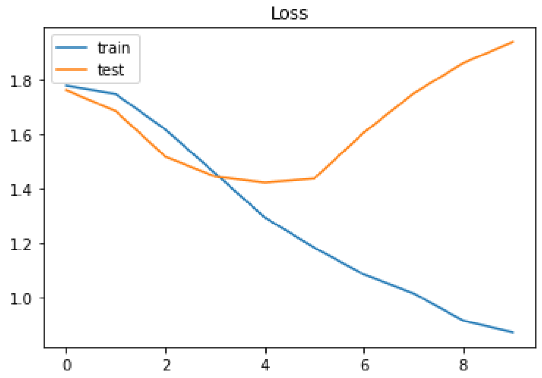
Confusion Matrix

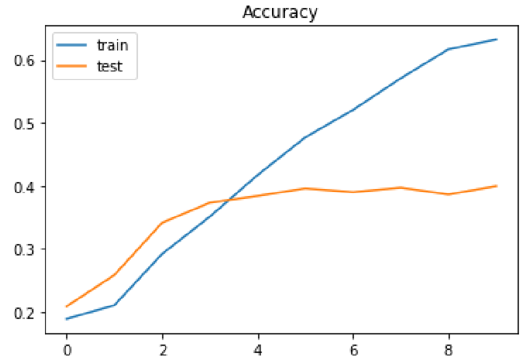


Age_group



Observations:
- **Overall accuracy**: **0.4069**
- **Training error** (MSE): 0.0678
- **Validation error** (MSE): 1.5561

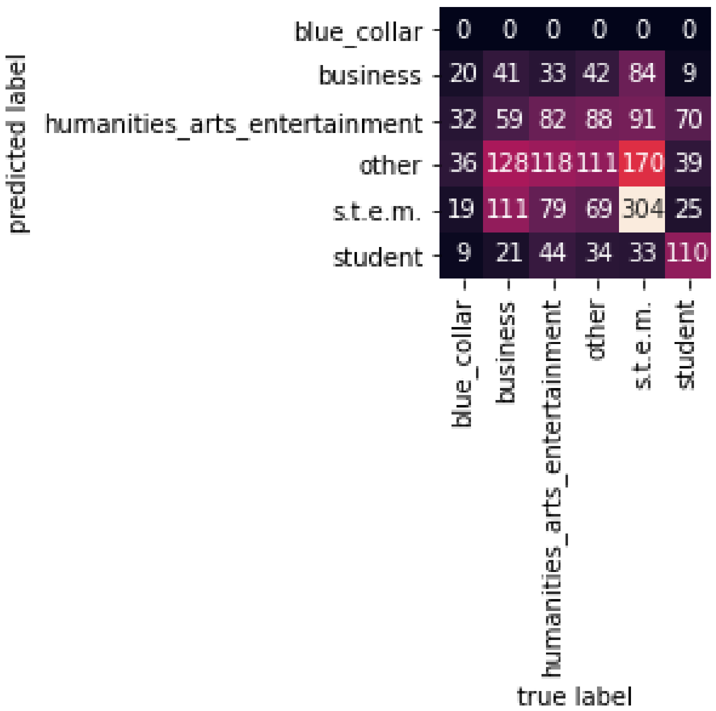




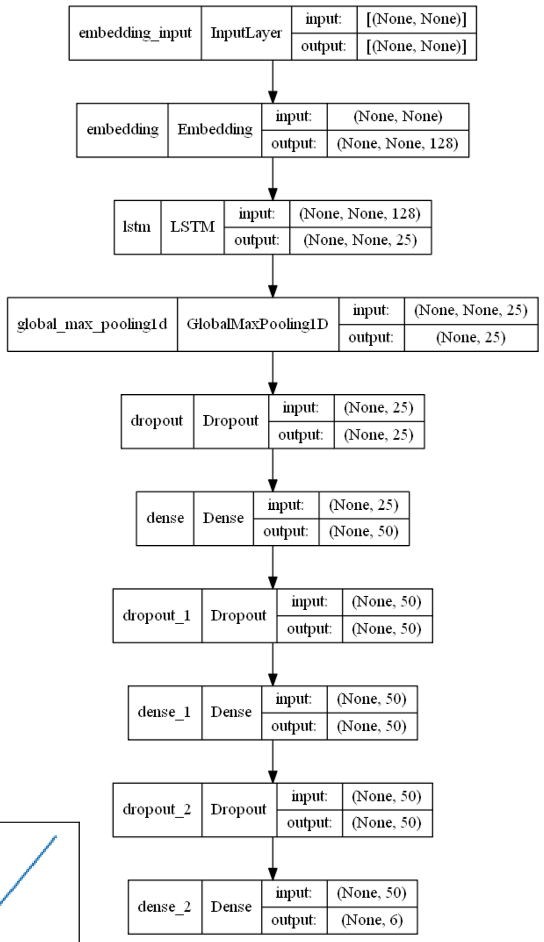
Advanced Model #2 – RNN: Case 3

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, None, 128)         11865728

 lstm (LSTM)                 (None, None, 25)          15400

 global_max_pooling1d (Globa  (None, 25)               0
 lMaxPooling1D)

 dropout (Dropout)           (None, 25)                0

 dense (Dense)               (None, 50)                1300

 dropout_1 (Dropout)         (None, 50)                0

 dense_1 (Dense)             (None, 50)                2550

 dropout_2 (Dropout)         (None, 50)                0

 dense_2 (Dense)             (None, 6)                 306

=================================================================
Total params: 11,885,284
Trainable params: 11,885,284
Non-trainable params: 0
_________________________________________________________________
```

Confusion Matrix

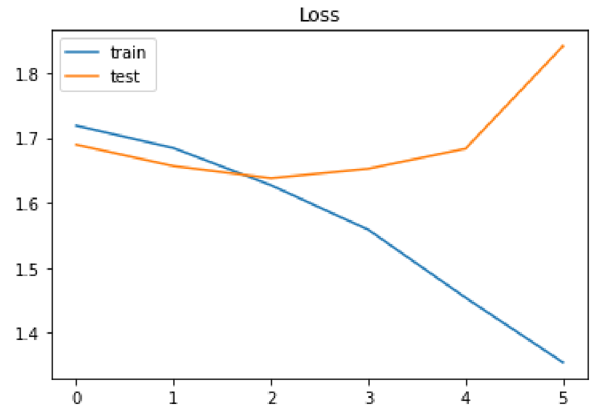


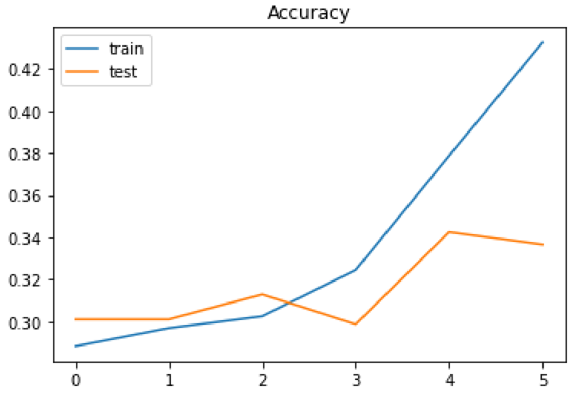
Job_group



Observations:

- **Overall accuracy**: **0.3069**
- **Training error** (MSE): 0.10023
- **Validation error** (MSE): 3.0833

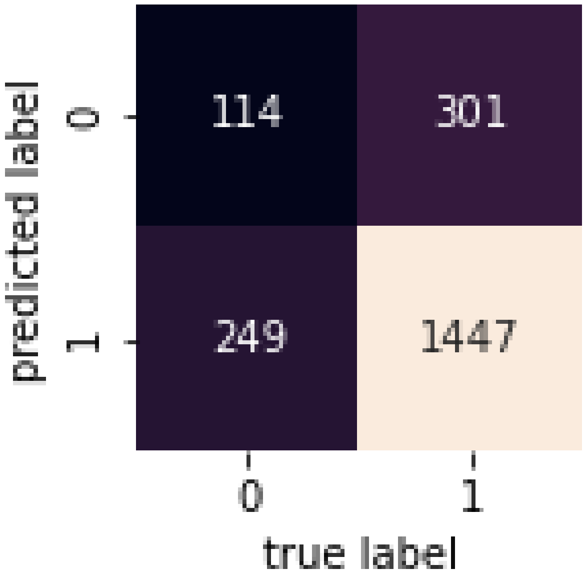




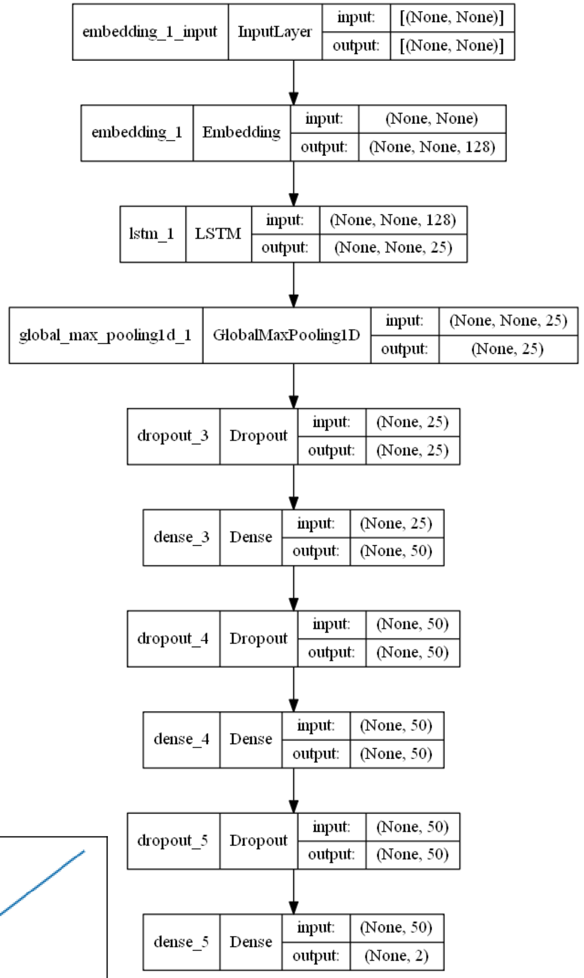
Advanced Model #2 – RNN: Case 4

```
Model: "sequential_1"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding_1 (Embedding)     (None, None, 128)         11865728

 lstm_1 (LSTM)               (None, None, 25)          15400

 global_max_pooling1d_1 (Glo  (None, 25)               0
 balMaxPooling1D)

 dropout_3 (Dropout)         (None, 25)                0

 dense_3 (Dense)             (None, 50)                1300

 dropout_4 (Dropout)         (None, 50)                0

 dense_4 (Dense)             (None, 50)                2550

 dropout_5 (Dropout)         (None, 50)                0

 dense_5 (Dense)             (None, 2)                 102

=================================================================
Total params: 11,885,080
Trainable params: 11,885,080
Non-trainable params: 0
_________________________________________________________________
```

Confusion Matrix

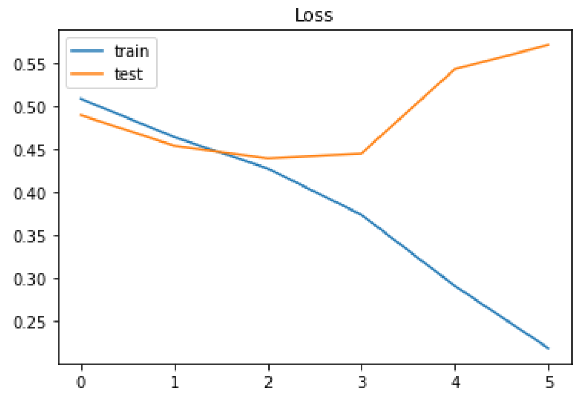


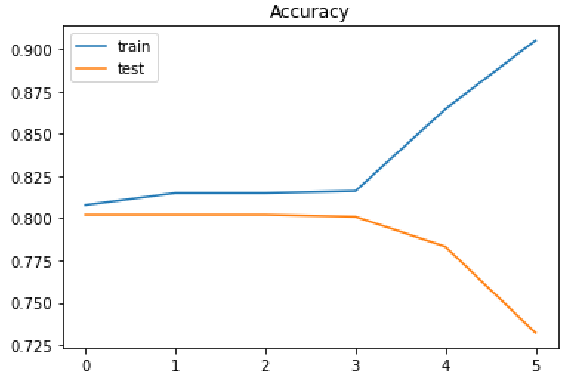
Drinks_freq



Observations:

- **Overall accuracy**: **0.73945997**
- **Training error** (MSE): 0.05238813
- **Validation error** (MSE): 0.260540



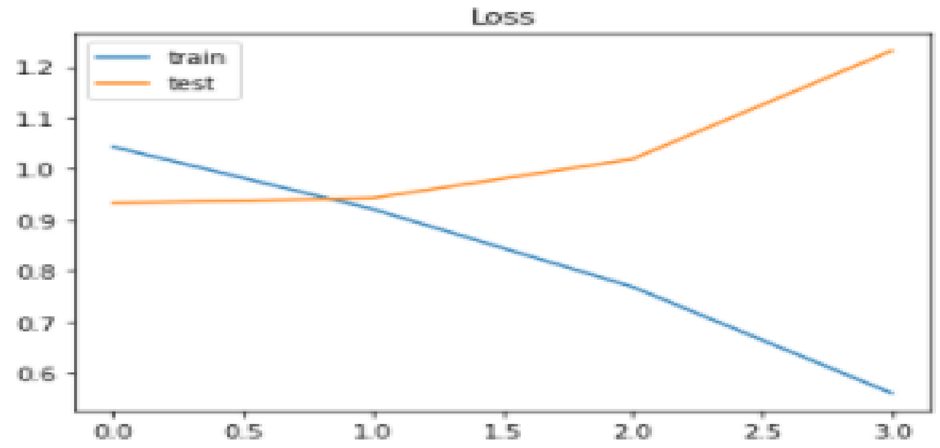


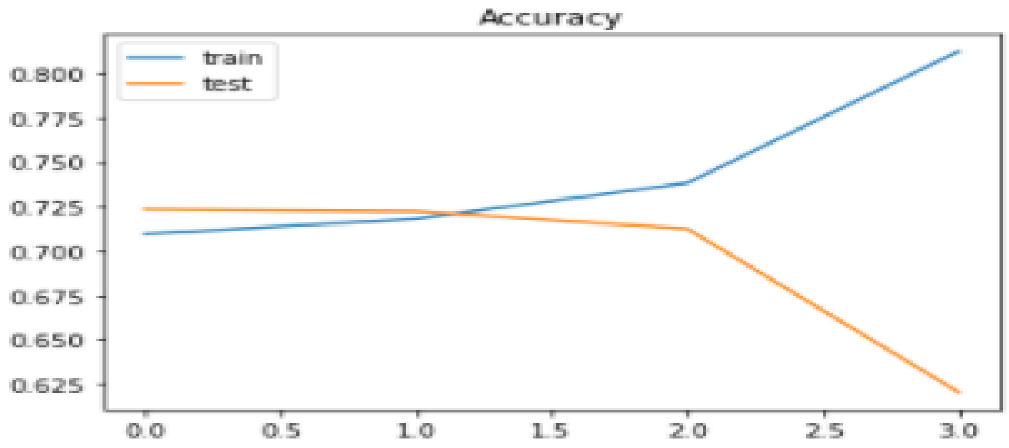
Advanced Model #3 - LSTM Model to classify: ‘Drinking’

- This LSTM model had ***recurrent dropout***, ***different architecture*** **& dropout values.** This makes it distinct from the RNN architecture described before.

The model had trouble with testing accuracy, often puts many types of sentences in the same category regardless of context given.

```
Epoch 1/5
109/109 [==============================] - 83s 739ms/step - loss: 1.0432 - accuracy: 0.7095 - val_loss: 0.9329 - val_accuracy:
0.7235
Epoch 2/5
109/109 [==============================] - 76s 701ms/step - loss: 0.9213 - accuracy: 0.7182 - val_loss: 0.9428 - val_accuracy:
0.7224
Epoch 3/5
109/109 [==============================] - 74s 678ms/step - loss: 0.7694 - accuracy: 0.7383 - val_loss: 1.0191 - val_accuracy:
0.7126
Epoch 4/5
109/109 [==============================] - 74s 682ms/step - loss: 0.5610 - accuracy: 0.8126 - val_loss: 1.2314 - val_accuracy:
0.6204
```

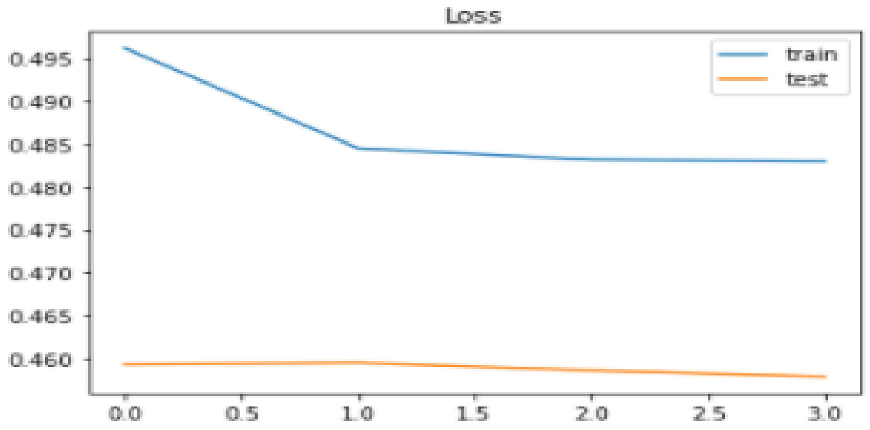


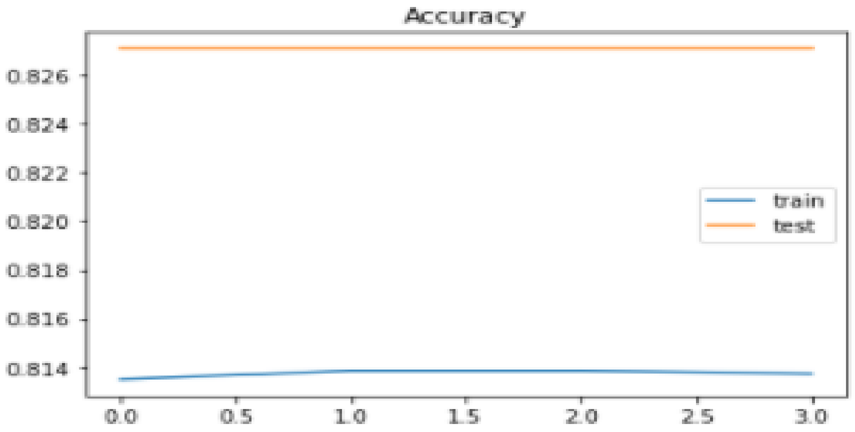


Advanced Model #4:Distill-BERT for 'Drinking_Frequency' Classification

Iteration 1 - Simple:

```
Epoch 1/4
132/132 [==============================] - 1779s 13s/step - loss: 0.4962 - accuracy: 0.8135 - val_loss: 0.4593 - val_accuracy:
0.8271
Epoch 2/4
132/132 [==============================] - 1603s 12s/step - loss: 0.4845 - accuracy: 0.8139 - val_loss: 0.4595 - val_accuracy:
0.8271
Epoch 3/4
132/132 [==============================] - 1693s 13s/step - loss: 0.4832 - accuracy: 0.8139 - val_loss: 0.4586 - val_accuracy:
0.8271
Epoch 4/4
132/132 [==============================] - 1867s 14s/step - loss: 0.4830 - accuracy: 0.8137 - val_loss: 0.4579 - val_accuracy:
0.8271
```

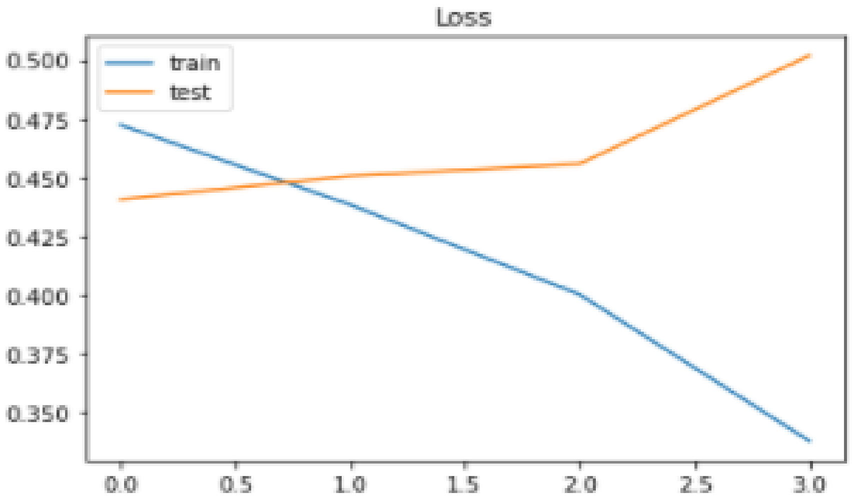


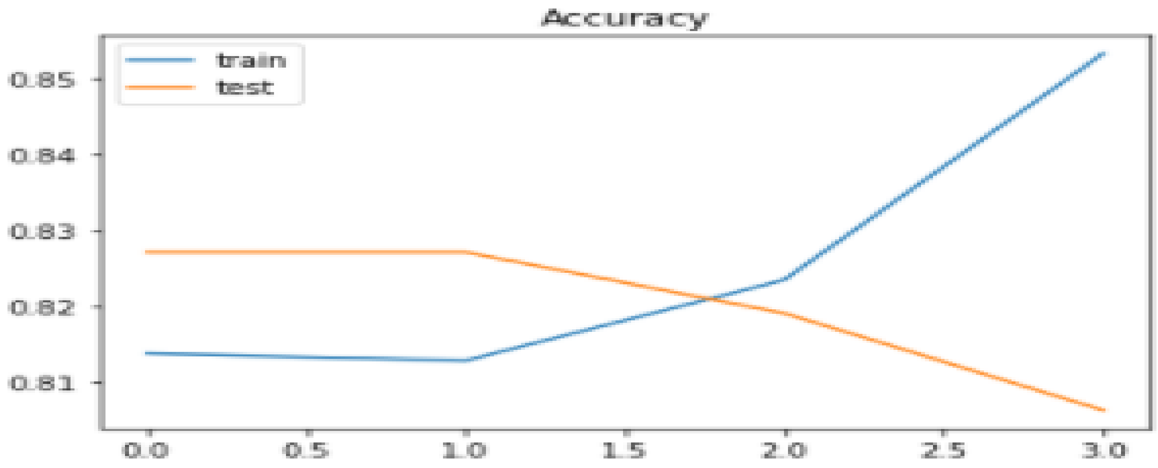


Advanced Model #4: Distill-BERT for ‘Drinking_Frequency’ Classification

Iteration 2 - Complex:

```
Epoch 1/4
132/132 [==============================] - 4112s 31s/step - loss: 0.4724 - accuracy: 0.8137 - val_loss: 0.4407 - val_accuracy:
0.8271
Epoch 2/4
132/132 [==============================] - 7416s 56s/step - loss: 0.4385 - accuracy: 0.8128 - val_loss: 0.4507 - val_accuracy:
0.8271
Epoch 3/4
132/132 [==============================] - 3762s 28s/step - loss: 0.4002 - accuracy: 0.8235 - val_loss: 0.4559 - val_accuracy:
0.8190
Epoch 4/4
132/132 [==============================] - 3760s 28s/step - loss: 0.3379 - accuracy: 0.8533 - val_loss: 0.5020 - val_accuracy:
0.8063
```

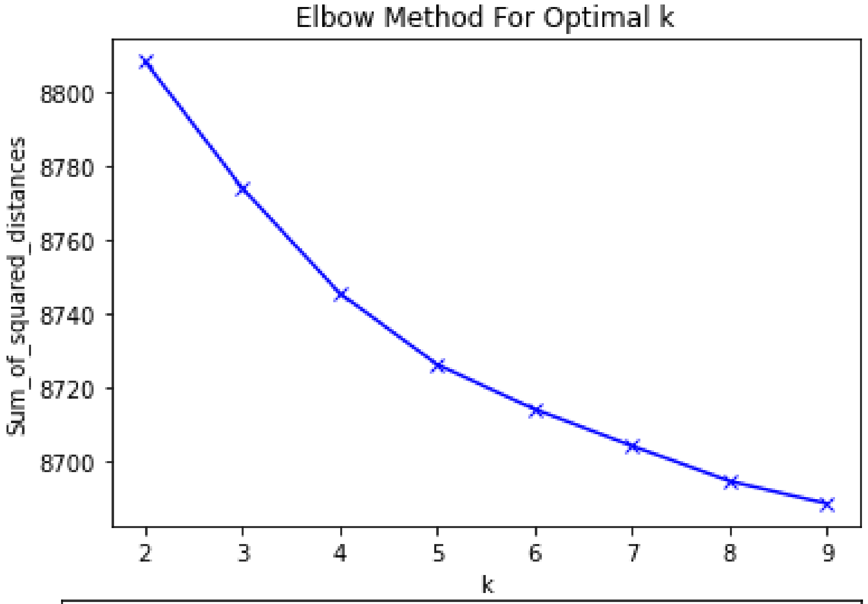




K-means Clustering: Additional project objective

- As additional objective, **K-means clustering** was implemented on the **10,850 bio essays** text to group them based on text similarity.
- TF-IDF is used to vectorize the bio values.
- Top **10 words** in each cluster were also obtained. No relevant inferences drawn.
- Experimented with **K = 3 , 4 and 5 clusters** to obtain different scatter plots. For **K =3 and K = 4** silhouette score improved but still close to 0, so it doesn't indicate proper clustering.

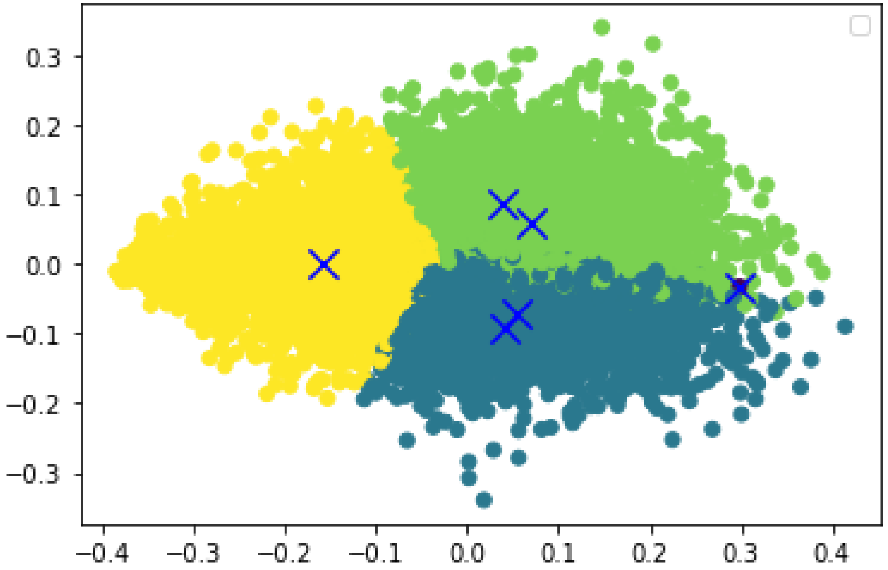
Elbow method was used to determine ***optimal number of clusters***



Performance Metrics:

- Estimating Model performance on unlabeled dataset:
  - ✓ **silhouette_score**: -0.0605493 (negative value indicates that some bio placed in wrong cluster.
  - ✓ Silhouette_score lies between -1 to +1. 0 indicating (***border-line***)

- Estimating Model performance against true labels:
  - ✓ **homogeneity_score** (education_group): 0.0065
  - ✓ **homogeneity_score** (age_group): 0.0111
  - ✓ **homogeneity_score** (job_group): 0.0040
  - ✓ **homogeneity_score** (drinks_freq): 0.0020



Cluster Map (**K =6**) chosen but **only 3** discernable groups realized:

Comparison of Accuracies across models

RUTGERS

Label/ Category being investigated	Baseline Model & Accuracy	Advanced Model & Accuracy	Observations
Education_group	NB, 0.385	RNN, 0.5229	36% improvement in RNN
		CNN, 0.5461	3-layer, 100 units CNN model had 41.8% higher accuracy
Age_group	NB, 0.0009	RNN, 0.4069	NB model couldn't classify at all, so RNN is significant improvement
		CNN, 0.4374	3-layer, 100 units CNN architecture perform better than RNN model with dropout
Job_group	NB, 0.1739	RNN, 0.3069	76% improvement with RNN
Drinks_freq	LR, 0.81	RNN, 0.7394	RNN <LR, but it is more generalized than base LR model.
		BERT, 0.82	BERT model edged LR for accuracy

Experimentation to overcome Bias/ Variance

- **Bigger dataset**: For Age RNN case, model was experimented with 12000 datasets and accuracy improved from 30% to 34%. Thus, larger datasets can help reduce High Variance.

- Due to computational limitations, bigger datasets beyond 12000 couldn't be handled by our PCs but as a future exploration, entire dataset volume (46000 entries) can be imported to run the model.

- **To mitigate Higher Variance:** Dropout and Recurrent dropout were incorporated. As shown, advanced models such as RNN, LSTM performed comparatively better due to introduction of dropout. Regularization could be tried in future as another strategy to reduce variance.

- **Changing Epoch length, batch size** : Batch sizes were changed from 32 to 50, 500 even for higher volume dataset (12000 entries). For data size of (10,000 to 15,000 entries), high batch size helps reduce computation time, but the accuracy didn't improve.

- **Bigger Architecture tried with CNN** : 3 layers (100 units each).

Conclusion

- **Size of experimented dataset** directly influences classification results.
- **Nature of data used (essays)** could be directly responsible for low accuracies. For example: "**Age_group**" cannot be discernably inferred from essays description unless someone explicitly mentions it out. So demarcating **b/w 23-27 and 28-32 age group** was particularly challenging even with sophisticated models.
- **Deeper Data investigation** and **pre-processing** could be necessary to understand if the sample dataset had out of English language words (foreign language bio).
- **Mathematical symbols** eliminated during text cleaning and emoticons could also have played a role. For instance: a doctor could have described their profession by an emoji, but the data cleaning removed it hence classification accuracy got impacted
- **Binary labels** classified better in both baseline and advanced ML models.
- **Poor performance from base models** such as NB, clearly indicated the importance of RNN -LSTM architecture to **capture context in long text (essays).**
- **Advanced models helped classify multinomial labels** to some extent.
- In general, most of the discussed models suffered from **over-fitting.**
- **Higher ML** architectures yielded better results.

Thank You!