Team 2

UNLOCKING

THE ART OF WRITING

Analyzing Behavioral Patterns for Enhanced Writing Quality Assessment

Our Team 2

Ashish Agarwal

Project Manager/
Data Scientist

Sowmya Bathula

Data Engineer

Harshitha G.S.

Data Analyst

Shaagun Suresh

ML Engineer

Have you ever wondered if those small actions, pauses, and revisions could be the key to enhance writing quality?

Problem Statement

Does typing behavior effect the outcome of an essay?

OBJECTIVE

- To uncover and clarify the connections between different writing processes and the quality of written content.

- Focus on fostering better writing practices by clarifying how each writing process influences the overall quality of written content.

Data Source and Distribution

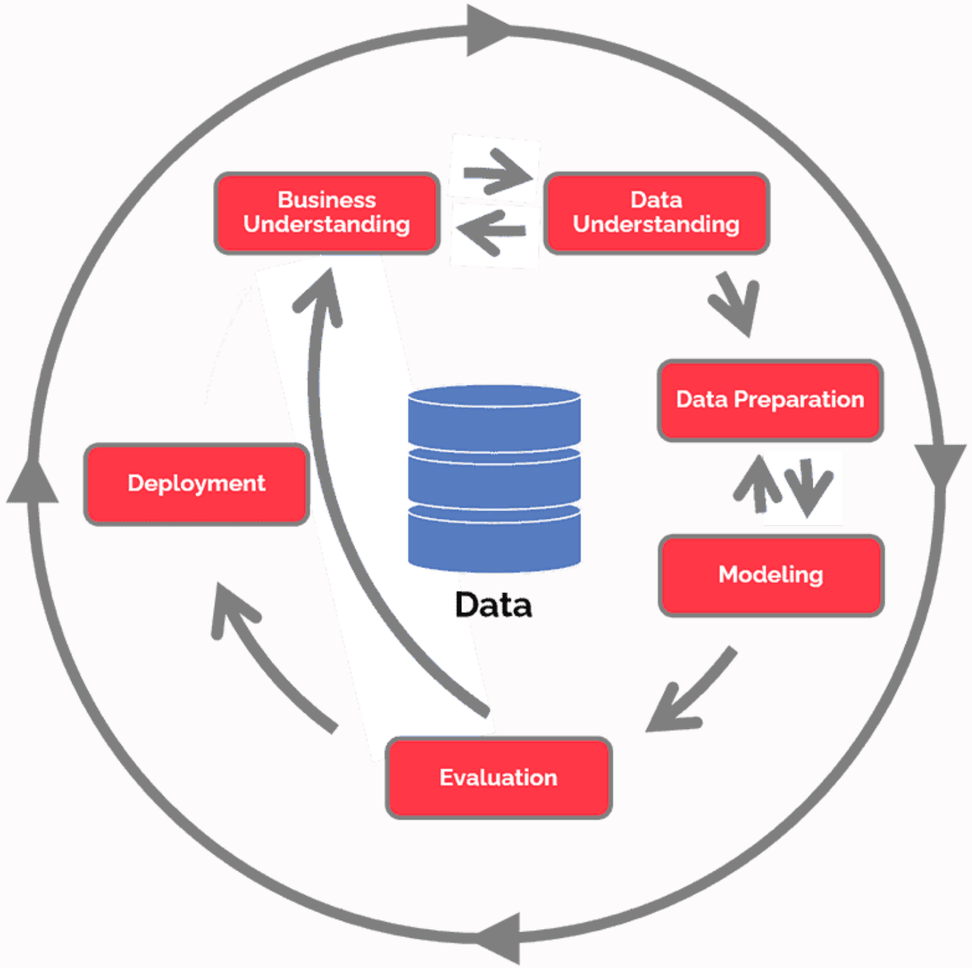
01

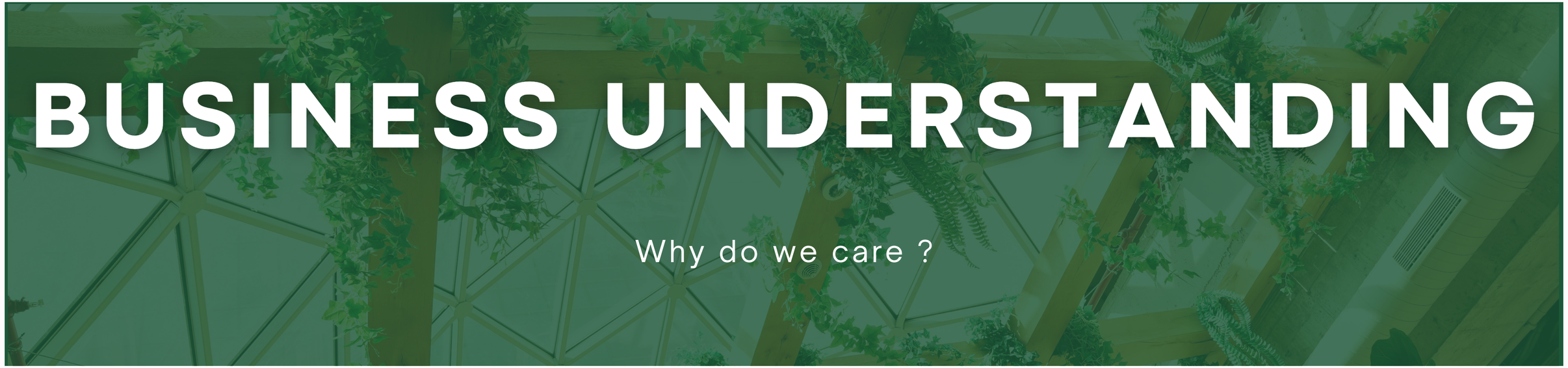
The dataset is taken from Kaggle .

Large dataset of keystroke logs capturing writing process features.

02

CRISP METHODOLOGY





BUSINESS UNDERSTANDING

Why do we care ?

Empowering Intelligent Tutoring and grading Systems

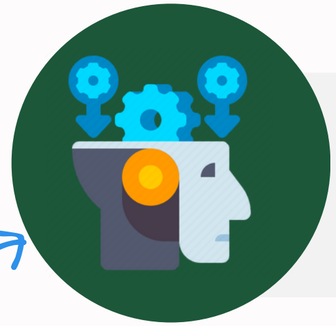
- Supporting educational institutions in optimizing teaching methods.
- Enhancing students' writing skills and learning outcomes.

DATA UNDERSTANDING

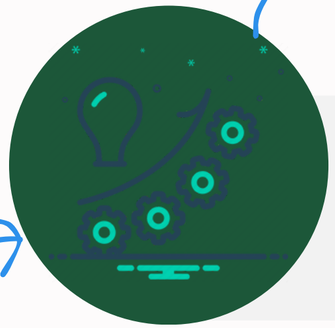
```
df.head(10)
```

	id	event_id	down_time	up_time	action_time	activity	down_event	up_event	text_change	cursor_position	word_count
0	001519c8	1	4526	4557	31	Nonproduction	Leftclick	Leftclick	NoChange	0	0
1	001519c8	2	4558	4962	404	Nonproduction	Leftclick	Leftclick	NoChange	0	0
2	001519c8	3	106571	106571	0	Nonproduction	Shift	Shift	NoChange	0	0
3	001519c8	4	106686	106777	91	Input	q	q	q	1	1
4	001519c8	5	107196	107323	127	Input	q	q	q	2	1
5	001519c8	6	107296	107400	104	Input	q	q	q	3	1
6	001519c8	7	107469	107596	127	Input	q	q	q	4	1
7	001519c8	8	107659	107766	107	Input	q	q	q	5	1
8	001519c8	9	107743	107852	109	Input	q	q	q	6	1
9	001519c8	10	107840	107978	138	Input	Space	Space		7	1

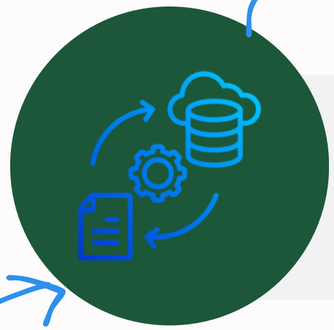
DATA PREPARATION



Feature Engineering



Data Augmentation



Data Transformation

Data Cleaning

Modeling

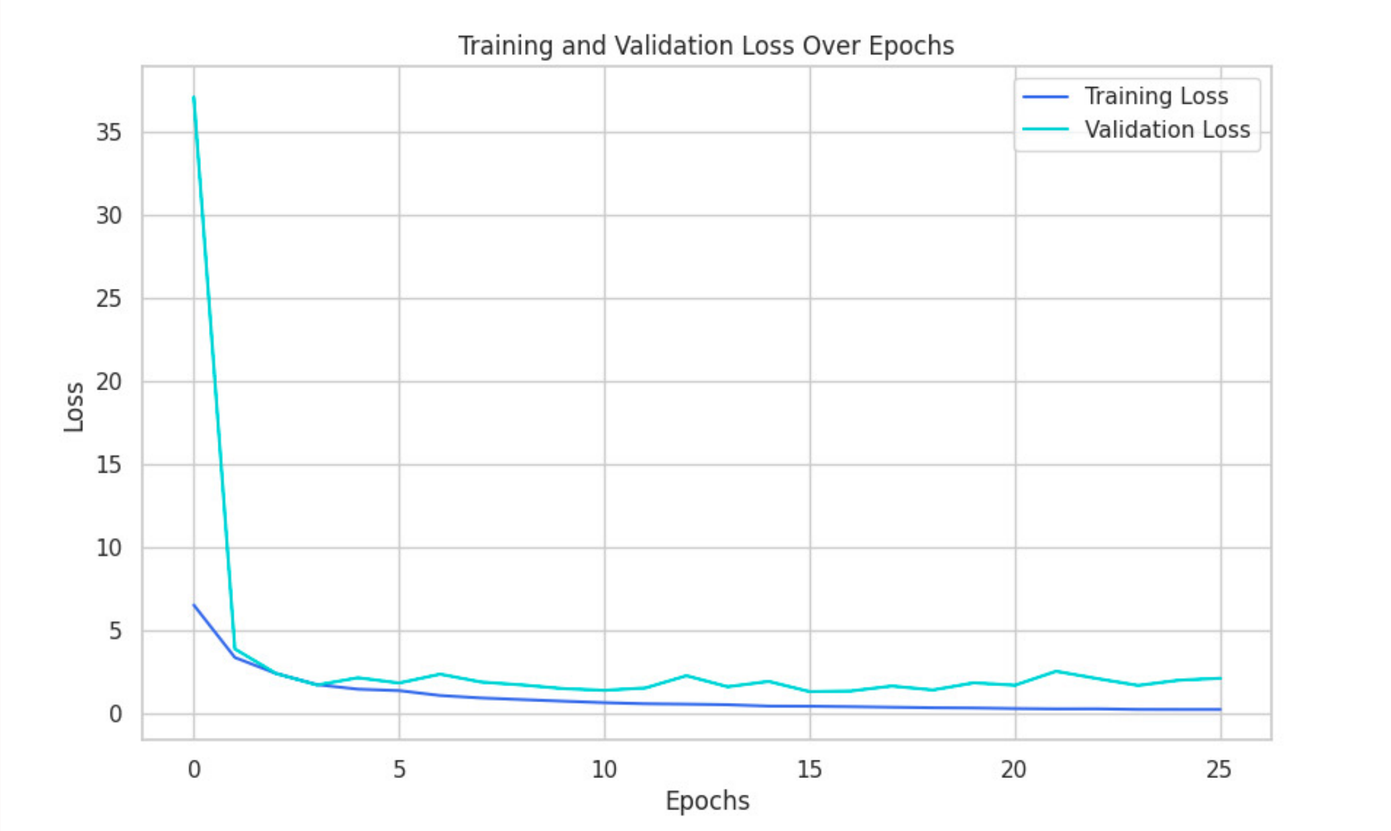
- Neural network, more precisely, sequential model is being used for Modeling.
- The Sequential model organizes layers in a linear stack, facilitating a step-by-step flow of data through the network.

```
# Define the model with dropout layers
model = Sequential()
model.add(Dense(1024, input_dim=X_train.shape[1], activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(512, input_dim=X_train.shape[1], activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(256, input_dim=X_train.shape[1], activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(128, activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(64, activation='relu'))
model.add(Dropout(0.2))
model.add(Dense(32, activation='relu'))
model.add(Dense(1, activation='linear'))

model.compile(optimizer='rmsprop', loss='mean_squared_error')

# Define callbacks for early stopping and model checkpointing
early_stopping = EarlyStopping(monitor='val_loss', patience=10, restore_best_weights=True)
model_checkpoint = ModelCheckpoint('best_model.h5', monitor='val_loss', save_best_only=True)

# Train the model with callbacks
history = model.fit(
    X_train, y_train,
    epochs=100,
    batch_size=64,
    validation_data=(X_test, y_test),
    callbacks=[early_stopping, model_checkpoint]
)

# Load the best model from model checkpoint
best_model = load_model('best_model.h5')
# best_model = load_model('best_model.h5', custom_objects={'loss': weighted_loss})

# Make predictions using the best model
predictions = best_model.predict(X_test)

predictions = np.clip(predictions, 0.3, 6.2)


predictions_rounded_values = [round_to_nearest_half(value) for value in predictions]

# Print some predictions
for true_value, pred_value in zip(y_test[:5], predictions_rounded_values[:5]):
    print(f'True Value: {true_value:.2f}, Predicted Value: {pred_value[0]}')
```

```
Epoch 1/100
70/70 [==============================] - 3s 21ms/step - loss: 4.1818 - val_loss: 12.7029
Epoch 2/100
 6/70 [=>............................] - ETA: 1s - loss: 3.1076
/usr/local/lib/python3.10/dist-packages/keras/src/engine/training.py:3079: UserWarning: You are saving your model as an HDF5 file via `model.save()`. This fil
red legacy. We recommend using instead the native Keras format, e.g. `model.save('my_model.keras')`.
  saving_api.save_model(
70/70 [==============================] - 2s 22ms/step - loss: 2.5480 - val_loss: 6.7932
Epoch 3/100
70/70 [==============================] - 2s 26ms/step - loss: 2.0351 - val_loss: 2.6139
Epoch 4/100
70/70 [==============================] - 2s 25ms/step - loss: 1.6336 - val_loss: 1.7025
Epoch 5/100
70/70 [==============================] - 2s 30ms/step - loss: 1.3215 - val_loss: 2.1414
Epoch 6/100
70/70 [==============================] - 1s 17ms/step - loss: 1.2010 - val_loss: 2.2354
Epoch 7/100
70/70 [==============================] - 1s 18ms/step - loss: 1.0775 - val_loss: 2.3753
Epoch 8/100
70/70 [==============================] - 1s 18ms/step - loss: 0.9195 - val_loss: 2.2649
Epoch 9/100
70/70 [==============================] - 1s 17ms/step - loss: 0.8622 - val_loss: 3.0921
Epoch 10/100
70/70 [==============================] - 1s 17ms/step - loss: 0.7645 - val_loss: 7.2220
Epoch 11/100
70/70 [==============================] - 1s 18ms/step - loss: 0.7045 - val_loss: 1.9068
Epoch 12/100
70/70 [==============================] - 2s 22ms/step - loss: 0.6311 - val_loss: 1.9907
Epoch 13/100
70/70 [==============================] - 2s 25ms/step - loss: 0.5947 - val_loss: 1.8560
Epoch 14/100
70/70 [==============================] - 2s 25ms/step - loss: 0.5804 - val_loss: 3.1803
16/16 [==============================] - 0s 3ms/step
True Value: 3.50, Predicted Value: 4.0
True Value: 3.50, Predicted Value: 3.0
True Value: 4.00, Predicted Value: 3.0
True Value: 4.00, Predicted Value: 3.0
True Value: 2.50, Predicted Value: 4.0
```

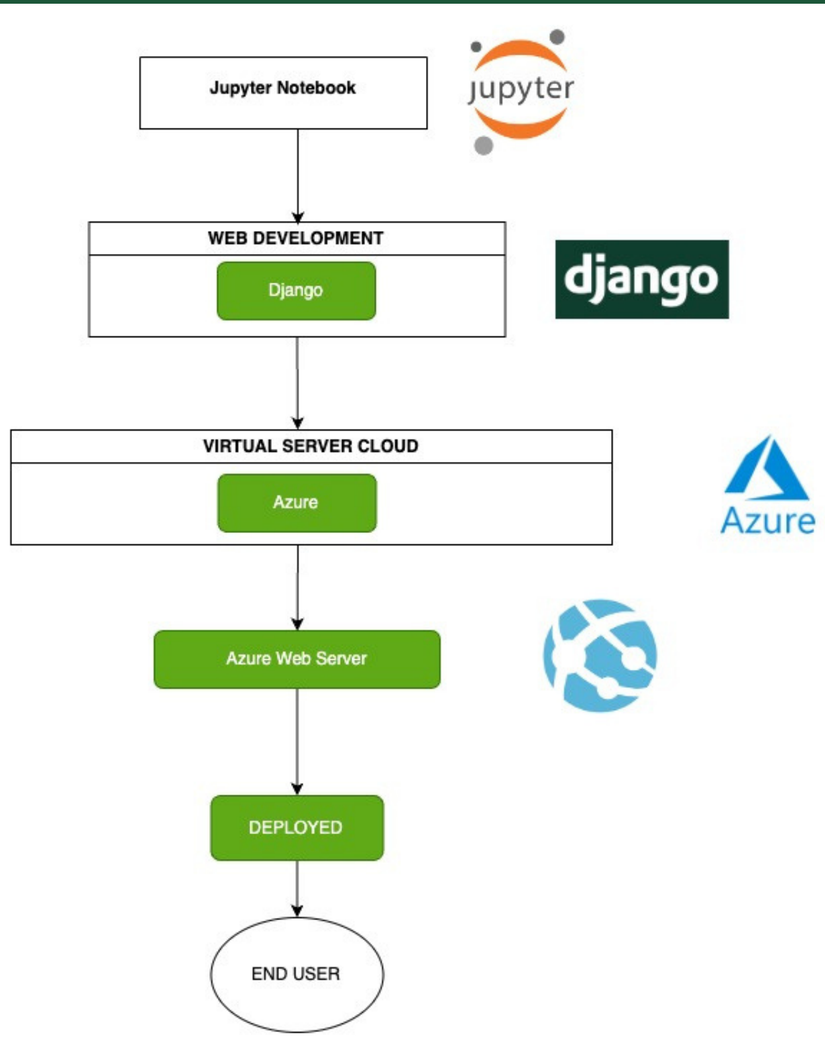
EVALUATION

- Mean Squared Error is being used for evaluation .
- It quantifies how close a regression line fits the data points.
- Lower MSE indicates better accuracy, as it reflects smaller prediction errors.



DEPLOYMENT

https://team-2-dsci-6002-f23-fp-unlocking-the-art-of-writing.azurewebsites.net/dashboard/



University of New Haven

THANK YOU