VAE Model Summary for encoder and decoder

Model: "encoder"

Layer (type)	Output Shape	Param #	Connected to
input_1 (InputLayer)	[(None, 28, 28, 1)]	0	[]
conv2d (Conv2D)	(None, 14, 14, 32)	320	['input_1[0][0]']
conv2d_1 (Conv2D)	(None, 7, 7, 64)	18496	['conv2d[0][0]']
conv2d_2 (Conv2D)	(None, 4, 4, 128)	73856	['conv2d_1[0][0]']
flatten (Flatten)	(None, 2048)	0	['conv2d_2[0][0]']
dense (Dense)	(None, 128)	262272	['flatten[0][0]']
z_mean (Dense)	(None, 2)	258	['dense[0][0]']
z_log_var (Dense)	(None, 2)	258	['dense[0][0]']
sampling (Sampling)	(None, 2)	0	['z_mean[0][0]', 'z_log_var[0][0]']

Total params: 355460 (1.36 MB)
Trainable params: 355460 (1.36 MB)
Non-trainable params: 0 (0.00 Byte)

Model: "decoder"

Layer (type)	Output Shape	Param #
input_2 (InputLayer)	[(None, 2)]	0
dense_1 (Dense)	(None, 6272)	18816
reshape (Reshape)	(None, 7, 7, 128)	0
conv2d_transpose (Conv2DTranspose)	(None, 14, 14, 128)	147584
conv2d_transpose_1 (Conv2DTranspose)	(None, 28, 28, 64)	73792
conv2d_transpose_2 (Conv2DTranspose)	(None, 28, 28, 32)	18464
conv2d_transpose_3 (Conv2DTranspose)	(None, 28, 28, 1)	289

Total params: 258945 (1011.50 KB)
Trainable params: 258945 (1011.50 KB)
Non-trainable params: 0 (0.00 Byte)

Hyperparameter	Value
Activation Function	ReLU (Rectified Linear Unit)
Weight Initializer	Glorot_uniform
Number of Hidden Layers	Encoder: 3 , Decoder: 4
Neurons in Hidden Layers	Encoder: [32, 64, 128], Decoder: [6272, 128, 64, 32, 1]
Loss function	Binary Crossentropy (BCE) for Reconstruction Loss, KL Divergence for KL Loss
Optimizer	Adam
Number of Epochs	30
Batch Size	128
Evaluation Metric	BCE and KL Loss
Encoder Layers	input_1, conv2d, conv2d_1, conv2d_2, flatten, dense, z_mean, z_log_var, sampling
Decoder Layers	input_2, dense_1, reshape, conv2d_transpose, conv2d_transpose_1, conv2d_transpose_2, conv2d_transpose_3

Problem 2:

TF-IDF + GaussianNB

```
Accuracy: 0.7989
Confusion Matrix:
[[4000  961]
 [1050 3989]]
Classification Report:
              precision    recall  f1-score   support

    negative       0.79      0.81      0.80      4961
    positive       0.81      0.79      0.80      5039

    accuracy                           0.80     10000
   macro avg       0.80      0.80      0.80     10000
weighted avg       0.80      0.80      0.80     10000
```

Word2Vec (CBoW) + GaussianNB

```
Accuracy: 0.6775
Confusion Matrix:
 [[3558 1403]
 [1822 3217]]
Classification Report:
              precision    recall  f1-score   support

    negative       0.66      0.72      0.69      4961
    positive       0.70      0.64      0.67      5039

    accuracy                           0.68     10000
   macro avg       0.68      0.68      0.68     10000
weighted avg       0.68      0.68      0.68     10000
```

Glove + GaussianNB

```
Accuracy: 0.6787
Confusion Matrix:
 [[3669 1292]
 [1921 3118]]
Classification Report:
              precision    recall  f1-score   support

    negative       0.66      0.74      0.70      4961
    positive       0.71      0.62      0.66      5039

    accuracy                           0.68     10000
   macro avg       0.68      0.68      0.68     10000
weighted avg       0.68      0.68      0.68     10000
```

Performance comparison table

Model	Accuracy	Precision (positive)	Recall (positive)	F1-Score (positive)	Precision (negative)	Recall (negative)	F1-Score (negative)
TF-IDF + GaussianNB	0.7989	0.81	0.79	0.80	0.79	0.81	0.80
Word2Vec (CBoW) + GaussianNB	0.6775	0.70	0.64	0.67	0.66	0.72	0.69
Glove + GaussianNB	0.6787	0.71	0.62	0.66	0.66	0.74	0.70

1. TF-IDF with Gaussian Naive Bayes achieved balanced performance for both positive and negative classes.
2. Word2Vec (CBoW) with Gaussian Naive Bayes exhibited lower performance for both classes compared to TF-IDF.
3. Glove with Gaussian Naive Bayes showed performance similar to Word2Vec (CBoW), with comparable metrics for both classes.
4. TF-IDF with Gaussian Naive Bayes remains the top-performing model, maintaining a balance between positive and negative class metrics.
5. Word2Vec (CBoW) and Glove with Gaussian Naive Bayes have similar performance, but TF-IDF outperforms them in most metrics.