url = 'https://files.consumerfinance.gov/ccdb/complaints.csv'

df = pd.read\_csv(url)

دانلود دیتا ست

df = df[['issue', 'product']].dropna() # Select relevant columns and drop missing values

انتخاب ستونهای مرتبط و حذف ستونهای غیر مرتبط با دستور dropna()

label\_encoder = LabelEncoder()

df['label'] = label\_encoder.fit\_transform(df['product'])

تبدیل کلمات به عدد

X\_train, X\_test, y\_train, y\_test = train\_test\_split(df['issue'], df['label'], test\_size=0.2, random\_state=42)

جدا کردن 20% دیتا ست بصورت رندم برای تست

tokenizer = Tokenizer(num\_words=10000) # Limit to the top 10,000 words

tokenizer.fit\_on\_texts(X\_train)

X\_train\_seq = tokenizer.texts\_to\_sequences(X\_train)

X\_test\_seq = tokenizer.texts\_to\_sequences(X\_test)

max\_length = max(len(x) for x in X\_train\_seq) # Find the maximum length of sequences

X\_train\_pad = pad\_sequences(X\_train\_seq, maxlen=max\_length, padding='post')

X\_test\_pad = pad\_sequences(X\_test\_seq, maxlen=max\_length, padding='post')

تبدیل داده ها به دنباله ای از اعداد صحیح و padding کردن بردارهای داده تا طول آنها با یکدیگر برابر باشد

Tokenizer برای محدود کردن واژگان به 10000 کلمه برتر مقداردهی شده است.

fit\_on\_texts کلمات شاخص را از داده های آموزشی یاد می گیرد.

texts\_to\_sequences متن را به دنباله ای از اعداد صحیح تبدیل می کند.

pad\_sequences تضمین می‌کند که تمام دنباله‌ها طول یکسانی دارند، با padding

input\_layer = Input(shape=(max\_length,))

embedding\_layer = Embedding(input\_dim=100, output\_dim=64)(input\_layer)

lstm\_out = LSTM(64, return\_sequences=True)(embedding\_layer)

# Attention mechanism

attention = Attention()([lstm\_out, lstm\_out])

attention = Flatten()(attention)

# Add a dropout layer for regularization

dropout\_layer = Dropout(0.5)(attention)

# Output layer

output\_layer = Dense(len(label\_encoder.classes\_), activation='softmax')(dropout\_layer)

# Compile the model

model = Model(inputs=input\_layer, outputs=output\_layer)

model.compile(optimizer='adam', loss='sparse\_categorical\_crossentropy', metrics=['accuracy'])

# Step 5: Train the model

model.fit(X\_train\_pad, y\_train, epochs=2, batch\_size=32, validation\_data=(X\_test\_pad, y\_test))

model.save('consumer\_complaints\_model\_with\_attention.keras')

print("Model saved as 'consumer\_complaints\_model\_with\_attention.keras'")

لایه آخر سافت مکس برای کلاسیفیکیشن و بعد کامپایل کردیم و ترین روی داده آموزشی

loaded\_model = load\_model(r'E:\My Python Codes\consumer\_complaints\_model\_with\_attention.keras') # Update the path as needed

print("Model loaded successfully.")

لود مدل آموزش دیده که قبلا سیو کردیم

sample\_indices = [500, 109, 326,674,888,111,440,76,9876,53,75, 46,40,23,5,67,102,76,456,87,24,4,854,322,212,54,12,234,777,13]

sample\_texts = df['Issue'].iloc[sample\_indices].tolist()

sample\_labels = df['label'].iloc[sample\_indices].tolist()

sample\_sequences = tokenizer.texts\_to\_sequences(sample\_texts)

sample\_padded = pad\_sequences(sample\_sequences, maxlen=max\_length, padding='post')

تعدا د 30 نمونه از Issue و label انتخاب و pre process شدند تا تتیجه را ببینیم

predictions = loaded\_model.predict(sample\_padded)

predicted\_classes = np.argmax(predictions, axis=1) # Convert probabilities to class labels

از مدل برای پیش بینی کلاس این سمپل ها استفاده نمودیم و نتیجه به صورت زیر شد با دقت 96.67%

Sample 20:

Text: Other transaction problem

Actual Label: Money transfer, virtual currency, or money service

Predicted Label: Money transfer, virtual currency, or money service

--------------------------------------------------

Sample 21:

Text: Communication tactics

Actual Label: Debt collection

Predicted Label: Debt collection

--------------------------------------------------

Sample 22:

Text: Problem with a company's investigation into an existing problem

Actual Label: Credit reporting or other personal consumer reports

Predicted Label: Credit reporting or other personal consumer reports

--------------------------------------------------

Sample 23:

Text: Incorrect information on your report

Actual Label: Credit reporting or other personal consumer reports

Predicted Label: Credit reporting or other personal consumer reports

--------------------------------------------------

Sample 24:

Text: Managing an account

Actual Label: Checking or savings account

Predicted Label: Checking or savings account

--------------------------------------------------

Sample 25:

Text: Other transaction problem

Actual Label: Money transfer, virtual currency, or money service

Predicted Label: Money transfer, virtual currency, or money service

--------------------------------------------------

Sample 26:

Text: Dealing with your lender or servicer

Actual Label: Student loan

Predicted Label: Student loan

--------------------------------------------------

Sample 27:

Text: Closing an account

Actual Label: Checking or savings account

Predicted Label: Checking or savings account

--------------------------------------------------

Sample 28:

Text: False statements or representation

Actual Label: Debt collection

Predicted Label: Debt collection

--------------------------------------------------

Sample 29:

Text: Incorrect information on your report

Actual Label: Credit reporting or other personal consumer reports

Predicted Label: Credit reporting or other personal consumer reports

--------------------------------------------------

Sample 30:

Text: Other transaction problem

Actual Label: Money transfer, virtual currency, or money service

Predicted Label: Money transfer, virtual currency, or money service

--------------------------------------------------

Accuracy of selected samples: 96.67%