Task 2: Lookalike Model

In [13]:
```
import pandas as pd
from sklearn.metrics.pairwise import cosine_similarity
from sklearn.preprocessing import StandardScaler
```

In [14]:
```
customers = pd.read_csv('Customers.csv')
# prducts = pd.read_csv('Products.csv')
transactions = pd.read_csv('Transactions.csv')
```

Filter for first 20 customers

In [16]:
```
filtered_customers = customers[customers['CustomerID'].isin([f"C{i:04d}" for i in r
```

Merge transactions with customers and aggregate features for similarity calculation

In [17]:
```
customer_transactions = transactions.merge(filtered_customers, on='CustomerID')

customer_features = customer_transactions.groupby('CustomerID').agg({
    'TotalValue': 'sum',
    'Quantity': 'sum',
    'ProductID': 'nunique'
}).reset_index()
```

Normalizing features

In [18]:
```
scaler = StandardScaler()
scaled_features = scaler.fit_transform(customer_features.iloc[:, 1:])
```

Calculating Cos similarities

In [19]:
```
similarity_matrix = cosine_similarity(scaled_features)
```

Generating top 3 lookalikes for each customer

In [20]:
```
lookalike_results = {}
for i, customer_id in enumerate(customer_features['CustomerID']):
    sim_scores = similarity_matrix[i]
    top_indices = sim_scores.argsort()[-4:-1][::-1]
    lookalike_results[customer_id] = [(customer_features['CustomerID'][j], round(si
```

Convert results to DataFrame for Lookalike.csv

In [21]:
```
lookalike_df = pd.DataFrame({
    'CustomerID': list(lookalike_results.keys()),
    'Lookalikes': [str(lst) for lst in lookalike_results.values()]
})

# Save to CSV
lookalike_df.to_csv('Lookalike.csv', index=False)

print("Lookalike.csv has been successfully generated!")
```

```
Lookalike.csv has been successfully generated!
```

In [12]: