SEQUENTIAL PATTERN MINING OF ONLINE RETAIL DATABASE USING SPADE ALGORITHM

GitHub Repository Link – https://github.com/JayaAadityaaG/DM_FinalAssign

CS F415 – DATA MINING
ASSIGNMENT FOR GRADUATING STUDENTS

JAYA AADITYAA G
2016A4TS0213H

PROGRAM LAYOUT

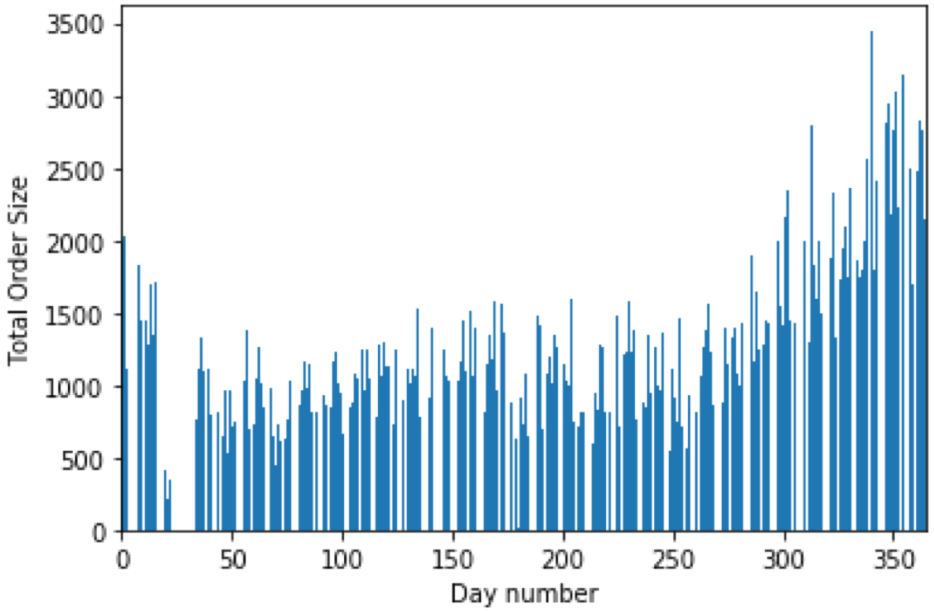
1. Initialization
2. Pre-processing
    1. Data Cleaning
    2. Data Reduction
    3. Data Transformation
3. Visualization
4. SPADE Algorithm
    1. Generation of frequent 1-subsequences
    2. Generation of frequent 2-subsequences
    3. Generation of frequent n-subsequences

PRE-PROCESSING

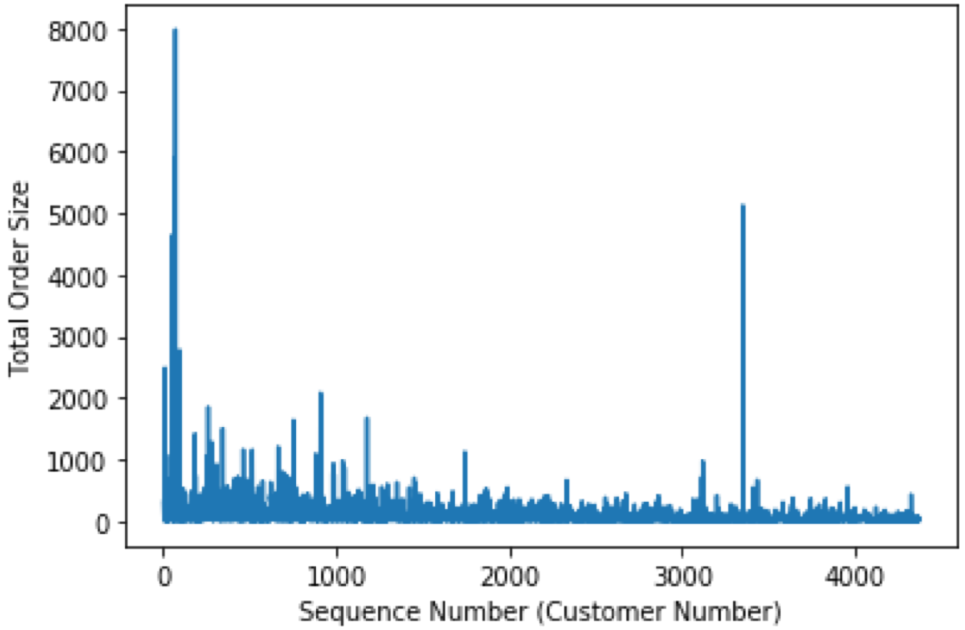
1. Data Cleaning – Removal of instances with null values in priority attributes.
2. Data Reduction – Removal of attributes unnecessary to task at hand.
3. Data Transformation
    1. Cancellation – Restructuring of cancellation by modification of 'StockCode' and 'InvoiceNo'
    2. Time – Construction of 'Time Difference' feature which time of order in days with respect to first purchase.
    3. Sequence Number – Sequence Number assigned using 'CustomerID' attibute.
    4. Event Number – 'Event Number' assigned using 'Sequence Number' and 'InvoiceNo'
    5. Sequence – Set of all items in a purchase with given sequence and event number are constructed in 'ItemSet' attribute.

Preprocessing ends with the dataset being converted into the required vertical data format.

VISUALIZATION



In the above plot, the trend for the total order size that day is displayed. It is observed that any surge in the total order size is due to the occurrence of a popular holiday e.g. Christmas.



The above plot depicts the total order size as against each customer. We can see that other than a few outliers, most customers order between 0-1000 items overall.

SPADE ALGORITHM

1. Generation of frequent 1-subsequences
    1. Dataframe scanning to determine list of all items and their corresponding frequency.
    2. Determination of frequent 1-subsequences based on frequency and minimum support.
    3. Determination of position list (SeqNo,EveNo) of all occurrences of frequent 1-subsequences.
2. Generation of frequent 2-subsequences
    1. Generation of 2-subsequence by comparison of all combinations of different frequent 1-subsequences
    2. Generation of 2-subsequence by comparison of all combinations of same frequent 1-subsequences
    3. Determination of frequent 2-subsequence by comparison of all generated 2-subsequences with threshold support
3. Generation of frequent n-subsequences
    1. Iterative generation of (n+1)-subsequence from n-subsequence using 2-subsequnce as base case.
    2. Printing of all obtained frequent subsequences with size greater than 1 in terms of 'StockCode' and then in terms of 'Description.

RESULTS – FREQUENTLY OCCURRING SUBSEQUENCES

S.No	Subsequence
1.	[['PARTY BUNTING', 'SPOTTY BUNTING']]
2.	[['JUMBO BAG RED RETROSPOT'], ['LUNCH BAG RED RETROSPOT']]
3.	[['LUNCH BAG RED RETROSPOT', 'JUMBO BAG RED RETROSPOT']]
4.	[['LUNCH BAG RED RETROSPOT'], ['LUNCH BAG BLACK SKULL.']]
5.	[['LUNCH BAG BLACK SKULL.'], ['LUNCH BAG RED RETROSPOT']]
6.	[['LUNCH BAG RED RETROSPOT', 'LUNCH BAG BLACK SKULL.']]
7.	[['LUNCH BAG RED RETROSPOT'], ['LUNCH BAG SPACEBOY DESIGN ']]
8.	[['LUNCH BAG RED RETROSPOT', 'LUNCH BAG SPACEBOY DESIGN ']]
9.	[['LUNCH BAG RED RETROSPOT'], ['LUNCH BAG SUKI DESIGN ']]
10.	[['LUNCH BAG RED RETROSPOT', 'LUNCH BAG SUKI DESIGN ']]
11.	[['LUNCH BAG RED RETROSPOT'], ['LUNCH BAG DOILEY PATTERN ']]
12.	[['LUNCH BAG RED RETROSPOT', 'LUNCH BAG DOILEY PATTERN ']]
13.	[['JUMBO BAG RED RETROSPOT'], ['JUMBO BAG DOILEY PATTERNS']]
14.	[['JUMBO BAG DOILEY PATTERNS'], ['JUMBO BAG RED RETROSPOT']]
15.	[['JUMBO BAG RED RETROSPOT', 'JUMBO BAG DOILEY PATTERNS']]
16.	[['LUNCH BAG BLACK SKULL.', 'LUNCH BAG SPACEBOY DESIGN ']]
17.	[['LUNCH BAG BLACK SKULL.', 'LUNCH BAG SUKI DESIGN ']]
18.	[['LUNCH BAG SPACEBOY DESIGN ', 'LUNCH BAG SUKI DESIGN ']]
19.	[['JUMBO BAG DOILEY PATTERNS', 'LUNCH BAG DOILEY PATTERN ']]
20.	[['WHITE HANGING HEART T-LIGHT HOLDER'],['WHITE HANGING HEART T-LIGHT HOLDER']]
21.	[['ASSORTED COLOUR BIRD ORNAMENT'], ['ASSORTED COLOUR BIRD ORNAMENT']]
22.	[['SET OF 3 CAKE TINS PANTRY DESIGN'],['SET OF 3 CAKE TINS PANTRY DESIGN ']]
23.	[['PARTY BUNTING'], ['PARTY BUNTING']]
24.	[['REGENCY CAKESTAND 3 TIER'], ['REGENCY CAKESTAND 3 TIER']]
25.	[['SMALL POPCORN HOLDER'], ['SMALL POPCORN HOLDER']]
26.	[['SPOTTY BUNTING'], ['SPOTTY BUNTING']]
27.	[['LUNCH BAG RED RETROSPOT'], ['LUNCH BAG RED RETROSPOT']]
28.	[['JUMBO BAG RED RETROSPOT'], ['JUMBO BAG RED RETROSPOT']]
29.	[['LUNCH BAG BLACK SKULL.'], ['LUNCH BAG BLACK SKULL.']]
30.	[['LUNCH BAG SUKI DESIGN '], ['LUNCH BAG SUKI DESIGN ']]
31.	[['JUMBO BAG DOILEY PATTERNS'], ['JUMBO BAG DOILEY PATTERNS']]
32.	[['LUNCH BAG DOILEY PATTERN '], ['LUNCH BAG DOILEY PATTERN ']]
33.	[['LUNCH BAG RED RETROSPOT', 'LUNCH BAG BLACK SKULL.']]
34.	[['LUNCH BAG RED RETROSPOT', 'LUNCH BAG SPACEBOY DESIGN ']]
35.	[['LUNCH BAG BLACK SKULL.', 'LUNCH BAG SPACEBOY DESIGN ']]
36.	[['LUNCH BAG RED RETROSPOT','LUNCH BAG BLACK SKULL'],['LUNCH BAG BLACK SKULL']]
37.	[['LUNCH BAG RED RETROSPOT', 'LUNCH BAG SUKI DESIGN '], ['LUNCH BAG SUKI DESIGN ']]
38.	[['JUMBO BAG RED RETROSPOT'], ['JUMBO BAG DOILEY PATTERNS'], ['JUMBO BAG RED RETROSPOT']]
39.	[['JUMBO BAG RED RETROSPOT'], ['JUMBO BAG DOILEY PATTERNS'], ['JUMBO BAG DOILEY PATTERNS']]
40.	[['LUNCH BAG BLACK SKULL.', 'LUNCH BAG SUKI DESIGN '], ['LUNCH BAG SUKI DESIGN ']]
41.	[['LUNCH BAG SPACEBOY DESIGN ', 'LUNCH BAG SUKI DESIGN '], ['LUNCH BAG SUKI DESIGN ']]
42.	[['JUMBO BAG DOILEY PATTERNS', 'LUNCH BAG DOILEY PATTERN '], ['LUNCH BAG DOILEY PATTERN ']]

MAIN ACHIEVEMENTS

- Conversion of online retail database into vertical format.
- Construction of pseudo-standardized time feature in days which can be used for future processing with timing constraints.
- Visualization after preprocessing which aided in the reduction of bias while observing trends.
- Implementation of COBJ support counting method in order to ensure personal biases and large occurrence of a particular subsequence in a customers history will not skew obtained results.
- Implementation of separate support thresholds for frequent 1-subsequence, 2-subsequend and n-subsequence which can be tweaked based on desired objectives.

DRAWBACKS

- Significant time consumption particularly if support thresholds are lowered.
- Usage of 4-nested loop in order to generate subsequences.
- Removal of a significant portion of instances due to missing values for ‘CustomerID’.
- Usage of lower thresholds for higher subsequences in order as the support of the generated patterns are significantly lower .