Computer Science and Networking, School of Computing and Data Science, Wentworth Institute of Technology

Data Mining

Dr. Salem Othman

Summer 2023

Assignment Topic: Mining Frequent, Maximal, and Closed Itemsets with Emojis using Python

Problem Statement 1:

We have a dataset that consists of sequences of Emojis, where each Emoji represents a different item. For example, "😃 🍎 🏀 🐶" represents four items: happiness, apple, basketball, and dog, respectively. Each sequence is an individual transaction. Your task is to mine this dataset for maximal frequent itemsets and closed frequent itemsets.

Consider the following transactions as your dataset:

1: 😃 🍎 🏀 🐶

2: 😃 🍎 🐶

3: 😃 🏀 🐶

4: 😃 🏀 🍎

5: 😃 🏀 🐶

6: 😃 🍎

7: 😃 🏀

8: 😃 🐶

Consider the minimum support count as 3.

Tasks 1:

1. Manually calculate the frequent itemsets from the given dataset.

2. Identify the closed frequent itemsets and maximal frequent itemsets from the frequent itemsets you found in task 1.

3. Describe the difference between maximal frequent itemsets and closed frequent itemsets, providing examples from the dataset.

4. Discuss the advantages and disadvantages of using either maximal frequent itemsets or closed frequent itemsets in the context of data mining.

Please document your process and the findings in a report format, providing appropriate explanations and insights.

Submission: Submit your report in PDF format.

Problem Statement 2:

We have a dataset that consists of sequences of Emojis, where each Emoji represents a different item. For example, "😃 🍎 🏀 🐶" represents four items: happiness, apple, basketball, and dog, respectively. Each sequence is an individual transaction.

Consider the following transactions as your dataset:

1: 😃 🍎 🏀 🐶

2: 😃 🍎 🐶

3: 😃 🏀 🐶

4: 😃 🏀 🍎

5: 😃 🏀 🐶

6: 😃 🍎

7: 😃 🏀

8: 😃 🐶

The minimum support count is 3.

Your task 2 is to write a Python program that:

1. Reads the provided data.

2. Encodes the data appropriately for frequent itemset mining.

3. Uses the Apriori algorithm to find all frequent itemsets in the data.

4. Implements a function to identify the closed frequent itemsets and maximal frequent itemsets from the frequent itemsets found in task 3.

5. Prints the frequent, maximal, and closed itemsets.

Make sure your code is well-commented, so it's clear what each part of the code does. Submit your Python code file (.py) as your assignment.

Hints:

You can use the `mlxtend` library, which provides functions to preprocess the data (`TransactionEncoder`) and perform the Apriori algorithm (`apriori`). You'll need to write your own function to find the maximal and closed itemsets.

Submission: Submit your Python code file (.py) for grading.

Learning Outcomes:

Upon completion of this assignment, you should be more comfortable with:

1. The concept of frequent itemset mining and the use of the Apriori algorithm.

2. The definitions and differences between frequent, maximal, and closed itemsets.

3. Writing Python code to preprocess data and perform itemset mining.

4. Implementing algorithms for identifying maximal and closed frequent itemsets.

5. The use of the `mlxtend` library for data mining tasks.