Bài tập Thực hành môn Khai phá Dữ liệu

Họ và tên: Huỳnh Nguyễn Thế Dân

MSSV: 21110256

Lớp: 21TTH1

Cài đặt lại thuật toán Vertical Apriori

Định nghĩa hàm VerticalApriori

```
from itertools import combinations

def VerticalApriori(df_bin, min_support):
    # Bước 1: Tạo dạng biểu diễn dọc của dữ liệu
    itemsets = {}  # Khởi tạo từ điển để lưu trữ tập hợp các mục và giao dịch tương ứng
    num_transactions = len(df_bin)  # Đếm tổng số giao dịch trong DataFrame

    # Duyệt qua mỗi cột trong DataFrame, mỗi cột đại diện cho một mục
    for item in df_bin.columns:
        # Tìm các giao dịch nơi mục xuất hiện và lưu vào một tập hợp
        transactions = set(df_bin.index[df_bin[item] == 1].tolist())
        # Lưu trữ mục như một frozenset và giao dịch tương ứng vào từ điển itemsets
        itemsets[frozenset([item])] = transactions

    # Bước 2: Lọc các mục đơn thường xuyên
    frequent_itemsets = {}  # Khởi tạo từ điển cho các tập hợp mục thường xuyên
    # Kiểm tra từng tập hợp mục trong itemsets
    for itemset, transactions in itemsets.items():
        # Nếu tỷ lệ xuất hiện của tập hợp mục >= ngưỡng hỗ trợ min_support, lưu nó
        if len(transactions) / num_transactions >= min_support:
            frequent_itemsets[itemset] = transactions

    # Bước 3: Tạo các tập hợp mục thường xuyên lớn hơn từ các mục thường xuyên hiện tại
    current_itemsets = frequent_itemsets.copy()  # Bắt đầu từ tập hợp mục thường xuyên hiện có
    k = 2  # Thiết lập kích thước ban đầu của tập hợp mục là 2

    # Tiếp tục vòng lặp cho đến khi không còn tập hợp mục mới nào được tạo ra
    while current_itemsets:
        new_itemsets = {}  # Từ điển mới cho các tập hợp mục thường
```

```
xuyên tiếp theo
        keys = list(current_itemsets.keys())  # Lấy tất cả các khóa
hiện tại (các tập hợp mục)
        # Xét tất cả các cặp khả dĩ
        for c in combinations(keys, 2):
            combined_itemset = c[0] | c[1]  # Gộp hai tập hợp mục
            # Kiểm tra nếu kích thước của tập hợp mục mới bằng k
            if len(combined_itemset) == k:
                # Giao các giao dịch của hai tập hợp mục để tìm các
giao dịch chung
                combined_transactions =
current_itemsets[c[0]].intersection(current_itemsets[c[1]])
                # Kiểm tra ngưỡng hỗ trợ cho tập hợp mục mới
                if len(combined_transactions) / num_transactions >=
min_support:
                    # Lưu vào từ điển new_itemsets nếu đủ điều kiện
                    new_itemsets[combined_itemset] =
combined_transactions

        # Cập nhật tập hợp mục thường xuyên với tập mới và tiếp tục
vòng lặp
        frequent_itemsets.update(new_itemsets)
        current_itemsets = new_itemsets
        k += 1  # Tăng kích thước tập hợp mục cần xét

    return frequent_itemsets  # Trả về tất cả các tập hợp mục thường
xuyên
```

Tóm tắt thuật toán:

Khởi tạo và Chuẩn bị Dữ liệu

- Khởi tạo từ điển itemsets: Lưu trữ mỗi mục dưới dạng khóa với các giao dịch tương ứng của nó dưới dạng giá trị. Đây là biểu diễn dọc của dữ liệu.
- Tính tổng số giao dịch: Lấy chiều dài của df_bin để xác định tổng số giao dịch trong bộ dữ liệu.
- Lặp qua các cột của DataFrame df_bin: Mỗi cột đại diện cho một mục. Đối với mỗi mục, tìm các giao dịch nơi mục xuất hiện và lưu vào itemsets dưới dạng frozenset.

Xác định các mục đơn thường xuyên

- Khởi tạo frequent_itemsets: Một từ điển mới để lưu trữ các tập hợp mục thường xuyên.
- Lọc các tập hợp mục: Kiểm tra từng tập hợp mục trong itemsets để xem liệu chúng có đáp ứng ngưỡng hỗ trợ tối thiểu hay không. Chỉ những tập hợp nào đáp ứng yêu cầu mới được thêm vào frequent_itemsets.

Tạo các tập hợp mục lớn hơn

- Khởi tạo và cập nhật kích thước của tập hợp mục: Bắt đầu từ các mục đơn và tăng kích thước của tập hợp mục qua mỗi vòng lặp.

- Lặp cho đến khi không còn tập hợp mục mới: Tạo các tập hợp mục mới bằng cách kết hợp các tập hợp mục hiện tại. Sử dụng phép giao của các tập hợp giao dịch để xác định các giao dịch chung cho mục mới.
- Kiểm tra ngưỡng hỗ trợ: Chỉ giữ lại những tập hợp mục mới nếu chúng đáp ứng ngưỡng hỗ trợ.
- Cập nhật frequent_itemsets: Thêm các tập hợp mục mới thích hợp vào từ điển frequent_itemsets.

Kết thúc và Trả về kết quả

- Trả về frequent_itemsets chứa tất cả các tập hợp mục thường xuyên tìm được qua các vòng lặp.

Hết.