Pattern Mining

Lecture Notes

Nadjib Lazaar
Université Paris-Saclay
lazaar@lisn.fr

Introduction

Itemset mining is a fundamental technique in data mining aimed at discovering frequent or interesting sets of items in large transactional databases. By analyzing transactions, this approach extracts patterns—combinations of items that frequently appear together—to uncover hidden relationships and trends. This method is widely used in various fields, such as market basket analysis, fraud detection, discovering product correlations, and recommendation systems.

1 Data Mining (DM) and Knowledge Discovery in Databases (KDD)

Data Mining (DM), often referred to as *Knowledge Discovery in Databases (KDD)*, is a multidisciplinary field aimed at extracting useful patterns, relationships, and knowledge from large and complex datasets for decision-making, prediction, and exploration.

1.1 Definition and Scope

DM and KDD encompass:

- **Knowledge Exploration:** Identifying hidden structures, trends, and relationships within data.
- **Process Development:** Creating methodologies to extract useful information.
- **Algorithm Design:** Developing models to efficiently process and analyze data.
- **Knowledge Retrieval Mechanisms:** Building tools for discovering knowledge in large databases.

1.2 Steps of the KDD Process

The KDD process consists of:

1. **Data Selection:** Identifying relevant datasets.
2. **Data Preprocessing:** Cleaning and transforming data to ensure quality.
3. **Data Transformation:** Adapting data for analysis.
4. **Data Mining:** Applying algorithms to extract patterns.
5. **Evaluation and Interpretation:** Validating the discovered knowledge.
6. **Knowledge Presentation:** Communicating results in an understandable way.

1.3 Applications of DM

DM is applied in various domains:

- **Business Intelligence:** Segmentation, fraud detection, market basket analysis.
- **Healthcare:** Disease prediction, drug discovery.
- **Scientific Research:** Data analysis in genomics, astronomy, materials science.
- **Social Networks:** Sentiment analysis, trend detection.
- **Cybersecurity:** Intrusion and anomaly detection.

1.4 Challenges in DM and KDD

The main challenges include:

- **Data Complexity:** Handling noisy or incomplete data.
- **Scalability:** Analyzing large datasets.
- **Interpretability:** Making models understandable for non-experts.
- **Ethics:** Ensuring privacy and reducing biases in decision-making.
- **Human Interaction:** Incorporating user feedback into the discovery process.

DM and KDD are essential in data science, transforming raw data into actionable knowledge for informed decision-making.

2 Frequent Pattern Mining

Frequent Pattern Mining (FIM), introduced by Agrawal et al., aims to identify recurring patterns in data, revealing useful structures in market analysis, bioinformatics, and network security.

A key concept is the discovery of **frequent itemsets**, sets of elements that frequently co-occur in transactions. This field has evolved with several major extensions:

- **Closed Itemsets:** Frequent closed or maximal patterns that eliminate redundancy.
- **Association Rules:** "If-then" relationships between elements.
- **Rare Itemsets:** Infrequent yet relevant combinations (e.g., fraud detection, rare diseases).
- **Sequential Patterns:** Capturing event order (e.g., customer behavior, genomics).
- **Emerging Patterns:** Identifying evolving trends in data.

FIM remains a cornerstone of data mining, with continuous advancements in algorithmic efficiency and practical applications.

3 Notations and Definitions

In this section, we introduce fundamental concepts related to frequent itemset mining by defining the main notions and notations used.

3.1 Transactional Dataset and Itemsets

Definition 1 (Itemset and Item Set)**.** *Let $\mathcal{I} = \{p_1, \ldots, p_n\}$ be a set of n distinct objects, called* items*. An* itemset *P is defined as a non-empty subset of $\mathcal{I}$:*

$$P \subseteq \mathcal{I}.$$

Definition 2 (Transactional Dataset)**.** *A* transactional dataset *$\mathcal{D}$ is a collection of m transactions $t_1, \ldots, t_m$, where each transaction is a subset of items, i.e.,*

$$t_i \subseteq \mathcal{I}, \quad \forall i \in \{1, \ldots, m\}.$$

Example 1. *Consider a transactional dataset $\mathcal{D}_1$ containing 5 transactions and 5 items: For example, $\{A, B\}$ and $\{B, C, E\}$ are itemsets in $\mathcal{D}_1$.* $\diamondsuit$

Table 1: Example of a transactional dataset ($\mathcal{D}_1$).

Transaction	Items
t_1	A B D E
t_2	A C
t_3	A B C E
t_4	B C E
t_5	A B C E

3.2 Coverage and Frequency of an Itemset

Definition 3 (Coverage of an Itemset)**.** *The* coverage *of an itemset P in $\mathcal{D}$, denoted* $\texttt{cover}(P)$*, is the set of transactions containing P:*

$$\texttt{cover}(P) = \{t_i \in \mathcal{D} \mid P \subseteq t_i\}.$$

Definition 4 (Frequency of an Itemset)**.** *The* frequency *of an itemset P in $\mathcal{D}$, denoted* $\texttt{freq}(P)$*, is the number of transactions in which P appears:*

$$\texttt{freq}(P) = |\texttt{cover}(P)|.$$

Example 2. *In $\mathcal{D}_1$, consider the itemset $P = \{A, B\}$:*

- *It appears in transactions t_1, t_3, and t_5.*
- *Thus, its coverage is* $\texttt{cover}(P) = \{t_1, t_3, t_5\}$.
- *Its frequency is* $\texttt{freq}(P) = 3$.

♢

3.3 Frequent and Rare Itemsets

Definition 5 (Frequent and Rare Itemsets)**.** *Given a minimum frequency threshold α:*

- *An itemset P is* frequent *if* $\texttt{freq}(P) \geq \alpha$.
- *An itemset P is* rare *(or infrequent) if* $\texttt{freq}(P) < \alpha$.

We denote by $\texttt{FIs}_\alpha$ and $\texttt{RIs}_\alpha$ the sets of frequent and rare itemsets, respectively, with respect to α:

$$\texttt{FIs}_\alpha = \{P \mid \texttt{freq}(P) \geq \alpha\}, \quad \texttt{RIs}_\alpha = \{P \mid \texttt{freq}(P) < \alpha\}.$$

Example 3. *If $\alpha = 3$, then in $\mathcal{D}_1$:*

- *The itemset $\{A, B\}$ is frequent because* $\texttt{freq}(\{A, B\}) = 3$.
- *The itemset $\{D\}$ is rare because* $\texttt{freq}(\{D\}) = 1$.

♢

3.4 Maximal and Minimal Itemsets

Definition 6 (Maximal and Minimal Itemsets)**.** *An itemset P is:*

- Maximal *if it is frequent and none of its supersets are frequent:*

$$P \in \texttt{MaxIs}_\alpha \iff \texttt{freq}(P) \geq \alpha \text{ and } \forall Q \supset P, \texttt{freq}(Q) < \alpha.$$

- Minimal *if it is rare and none of its subsets are rare:*

$$P \in \texttt{MinIs}_\alpha \iff \texttt{freq}(P) < \alpha \text{ and } \forall Q \subset P, \texttt{freq}(Q) \geq \alpha.$$

Example 4. *If $\alpha = 3$, in $\mathcal{D}_1$:*

- *$\{A, B, E\}$ is a maximal itemset because it is frequent, but none of its supersets are frequent.*
- *$\{A, C, E\}$ is a minimal itemset because it is rare and none of its smaller subsets are rare.*

$\diamondsuit$

3.5 Closed and Generator Itemsets

Definition 7 (Closed and Generator Itemsets)**.** *An itemset P is:*

- Closed *if none of its supersets have the same frequency:*

$$P \in \texttt{CIs}_\alpha \iff \nexists Q \supset P, \texttt{freq}(Q) = \texttt{freq}(P).$$

- Generator *if none of its subsets have the same frequency:*

$$P \in \texttt{GIs}_\alpha \iff \nexists Q \subset P, \texttt{freq}(Q) = \texttt{freq}(P).$$

Example 5. *In $\mathcal{D}_1$ with $\alpha = 3$:*

- *$\{B, E\}$ is closed because none of its supersets have the same frequency.*
- *$\{B, C\}$ is a generator because none of its subsets have the same frequency.*

$\diamondsuit$

3.6 Summary of Relationships Between Itemsets

- Every maximal itemset is a closed itemset, but the converse is not true.
- Every minimal itemset is a generator, but the converse is not true.
- Each frequent itemset has a closed superset with the same coverage:

$$\forall P \in \texttt{FIs}_\alpha, \quad \texttt{cover}(P) = \max_{Q \in \texttt{CIs}_\alpha, Q \subseteq P} \texttt{cover}(Q).$$

- Each rare itemset has a generator subset with the same coverage:

$$\forall P \in \texttt{RIs}_\alpha, \quad \texttt{cover}(P) = \min_{Q \in \texttt{GIs}_\alpha, Q \supseteq P} \texttt{cover}(Q).$$

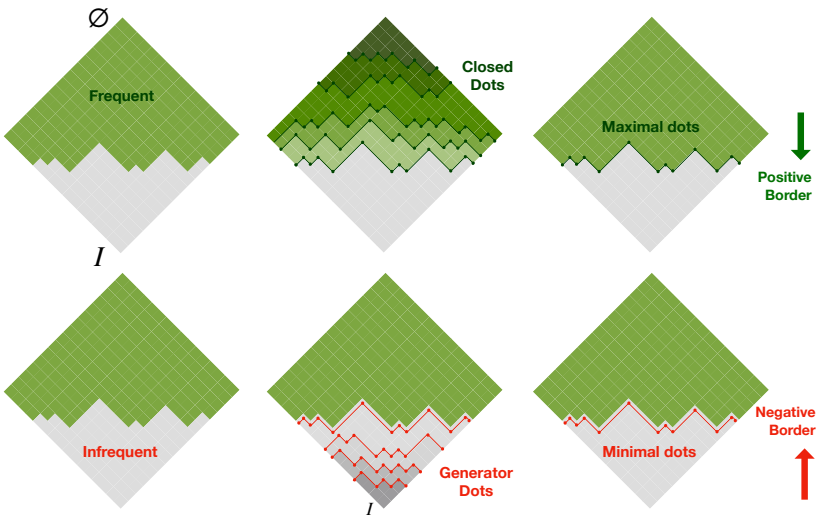

Figure 1: Frequent/Rare, Closed/Generator, and Maximal/Minimal Itemsets.

4 Anti-monotonicity and Apriori Property

Frequent itemset mining relies on two key principles: anti-monotonicity and the Apriori property. These properties help efficiently reduce the search space by quickly eliminating non-relevant itemsets.

4.1 Anti-monotonicity

Anti-monotonicity is an essential property that facilitates the exploration of frequent itemsets.

Definition 8 (Anti-monotonicity)**.** *An itemset P is* anti-monotone *if, for any itemset $Q \supseteq P$, we have:*

$$\texttt{freq}(Q) \leq \texttt{freq}(P).$$

This means that an itemset can never have a higher frequency than its subsets. An important consequence is that if an itemset is rare, then all its supersets will also be rare. This property is used to quickly eliminate infrequent candidates and avoid unnecessary computations.

Example 6. *In a transactional dataset, suppose that* $\texttt{freq}(\{A, B\}) = 2$*. Then, any itemset containing $\{A, B\}$ (such as $\{A, B, C\}$ or $\{A, B, D\}$) will have a frequency less than or equal to 2.* ♢

4.2 Apriori Property

The Apriori property directly follows from anti-monotonicity: a frequent itemset cannot contain rare subsets.

Definition 9 (Apriori Property)**.** *If an itemset P is frequent, then all its subsets must also be frequent:*

$$\forall S \subseteq P, \quad \texttt{freq}(P) \geq \alpha \Rightarrow \texttt{freq}(S) \geq \alpha.$$

This property allows for a progressive exploration of itemsets: we start by identifying small frequent itemsets and then build larger candidates only from already validated itemsets.

Example 7. *If $\{A, B, C\}$ is frequent, then $\{A, B\}$ and $\{B, C\}$ must also be frequent. Conversely, if $\{A, B\}$ is rare, then $\{A, B, C\}$ cannot be frequent.* $\diamondsuit$

4.3 The Apriori Algorithm

The Apriori algorithm is one of the most well-known algorithms used for frequent itemset mining. It leverages the Apriori property to efficiently generate candidate itemsets and prune those that are infrequent. The core idea of the Apriori algorithm is a breadth-first search where candidate itemsets are generated by joining frequent itemsets from the previous level, and infrequent itemsets are pruned using the anti-monotonicity property.

The algorithm works as follows:

1. **Initialization:** Start with the set of all frequent itemsets of size 1, which are the items appearing in at least the minimum threshold of transactions.
2. **Iterative Process:**
   - Generate candidate itemsets of size k by joining frequent itemsets of size $k-1$.
   - For each candidate itemset, scan the dataset and compute its frequency.
   - Prune infrequent candidates (i.e., those whose frequency is below the minimum threshold α).
3. **Repeat the process:** Repeat until no more frequent itemsets can be found.
4. **Generate rules:** Once all frequent itemsets are identified, generate association rules from them.

Algorithm 1: Apriori Algorithm

Input: Transactional database $\mathcal{D}$, minimum support threshold α
Output: Frequent itemsets
$k \leftarrow 1$;
$L_k \leftarrow \{p_i \mid p_i \in \mathcal{I} \wedge \texttt{freq}(p_i) \geq \alpha\}$;
while $L_k \neq \emptyset$ **do**
    $C \leftarrow \texttt{aprioriGen}(L_k)$;
    $k \leftarrow k + 1$;
    $L_k \leftarrow \{c \mid c \in C \wedge \texttt{freq}(c) \geq \alpha\}$;
return $\bigcup_i L_i$;

Function `aprioriGen`(L_k)**:**
    $E \leftarrow \emptyset$;
    for *each pair of itemsets* $P', P'' \in L_k$ *such that* $P' = \{p_{i_1}, \ldots, p_{i_{k-1}}, p_{i_k}\}$ *and* $P'' = \{p_{i_1}, \ldots, p_{i_{k-1}}, p_{i'_k}\}$ **do**
        if $p_{i_k} \neq p_{i'_k}$ **then**
            $P \leftarrow P' \cup P''$;
            if $\forall p_i \in P,\ P \setminus \{p_i\} \in L_k$ **then**
                $E \leftarrow E \cup \{P\}$;
    return E ;

The key steps of the Apriori algorithm involve iteratively generating candidate itemsets, scanning the dataset to determine their frequencies, and pruning infrequent itemsets based on the Apriori property. This process is repeated until no more frequent itemsets can be found, after which association rules can be generated from the frequent itemsets.