FLEXIS: Efficient Frequent Subgraph Mining with Flexible Metrics

This paper introduces FLEXIS, a novel method for Frequent Subgraph Mining (FSM) that addresses the challenges of time-consuming and complex execution. FLEXIS utilizes an innovative candidate generation technique and a flexible metric to provide substantial performance improvements over state-of-the-art algorithms.

Overview of the Context

Frequent Subgraph Mining (FSM)

Identifying recurring subgraphs in a larger graph that exceed a predefined frequency threshold.

Two-Step Approach

The FSM problem is generally solved in two steps: the Generation Step and the Metric Step.

Generation Step

Identifying potential high-frequency subgraphs within the large graph, often using edge or vertex extension methods.

Metric Step

Determining if the generated candidate subgraphs occur more than a predefined threshold, using metrics like Maximum Independent Set (MIS) and Minimum Node Image (MNI).

The context provided describes the Frequent Subgraph Mining (FSM) problem and the two-step approach used to solve it, highlighting the key components of the Generation Step and the Metric Step.

The Generation Step

- ## Identifying Potential High-Frequency Subgraphs

  The Generation Step focuses on identifying potential high-frequency subgraphs within the large graph.

- ## Edge Extension Method

  This can be achieved through the edge extension method, which involves merging two frequent (k-1)-subgraphs to create a potential k-edge subgraph.

- ## Vertex Extension Method

  Alternatively, the vertex extension method expands frequent (k-1)-vertex subgraphs by adding a vertex to create potential k-vertex subgraphs.

Challenges in the Generation Step

Identifying Suitable Merge Points

Merge operation must maintain the meaningful connectivity of the graph. FLEXIS uses core graphs to identify suitable merge points.

Handling Labels and Attributes

FLEXIS ensures coherent merged patterns by considering the labels and attributes on vertices and edges during the merging process.

Ensuring Uniqueness and Non-Redundancy

FLEXIS checks for automorphisms and determines canonical forms to ensure that the merged patterns are unique and non-redundant.

By addressing these key challenges, FLEXIS introduces a novel approach to frequent subgraph mining that outperforms existing state-of-the-art methods.

The Metric Step

- ## Determining Pattern Frequency

  The Metric Step involves calculating the frequency of occurrence of each candidate pattern in the data graph.

- ## Verifying Frequency Threshold

  The Metric Step also checks if the frequency of each pattern exceeds a predefined threshold.

- ## Advancements in Metrics

  The Metric Step has seen advancements, particularly with the Maximum Independent Set (MIS) metric, which counts disjoint, independent patterns in the data graph.

- ## Limitations of MIS

  However, the NP-complete nature of MIS requires considerable computational time, making it challenging to use.

Alternatives to MIS

Minimum Image (MNI) Metric

A faster, approximate alternative to MIS, but can significantly overestimate pattern counts.

Fractional-Score (FS) Metric

A variation of MNI that attempts to address the overestimation, but still occasionally overestimates certain patterns.

While these alternatives to MIS offer improved computational efficiency, they come with their own limitations in accurately capturing pattern frequencies.

The Proposed Approach: FLEXIS

- ## Maximal Independent Set (mIS) Metric

  Proposes a new metric called mIS, which is an approximation of the Maximum Independent Set (MIS) metric. mIS retains the independence property of MIS and does not suffer from overestimation of pattern counts. Additionally, mIS provides a user-controlled parameter λ to tune the accuracy-speed trade-off.

- ## Efficient Candidate Pattern Generation

  Introduces a novel approach for generating candidate patterns by merging two (k-1)-vertex frequent patterns to form potential k-vertex patterns. This method is more efficient than the edge extension and vertex extension techniques used in existing approaches, effectively pruning the search space.

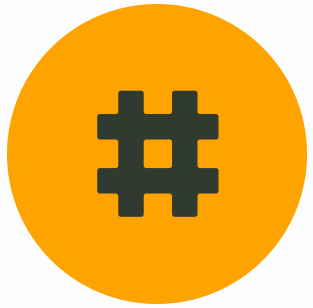
- ## Clique Pattern Generation

  Presents an extension to the candidate pattern generation process to handle the identification of clique patterns. FLEXIS strategically searches through the core groups to determine if the missing clique, which contains the missing edge of the candidate clique, is present, and then merges the two cliques to form the candidate clique pattern.

Innovative Candidate Pattern Generation

Identifying Suitable Merge Points

FLEXIS ensures that the merged patterns maintain meaningful connectivity in the graph.



Handling Labels and Attributes

FLEXIS handles labels and attributes on vertices and edges to ensure coherent merged patterns.

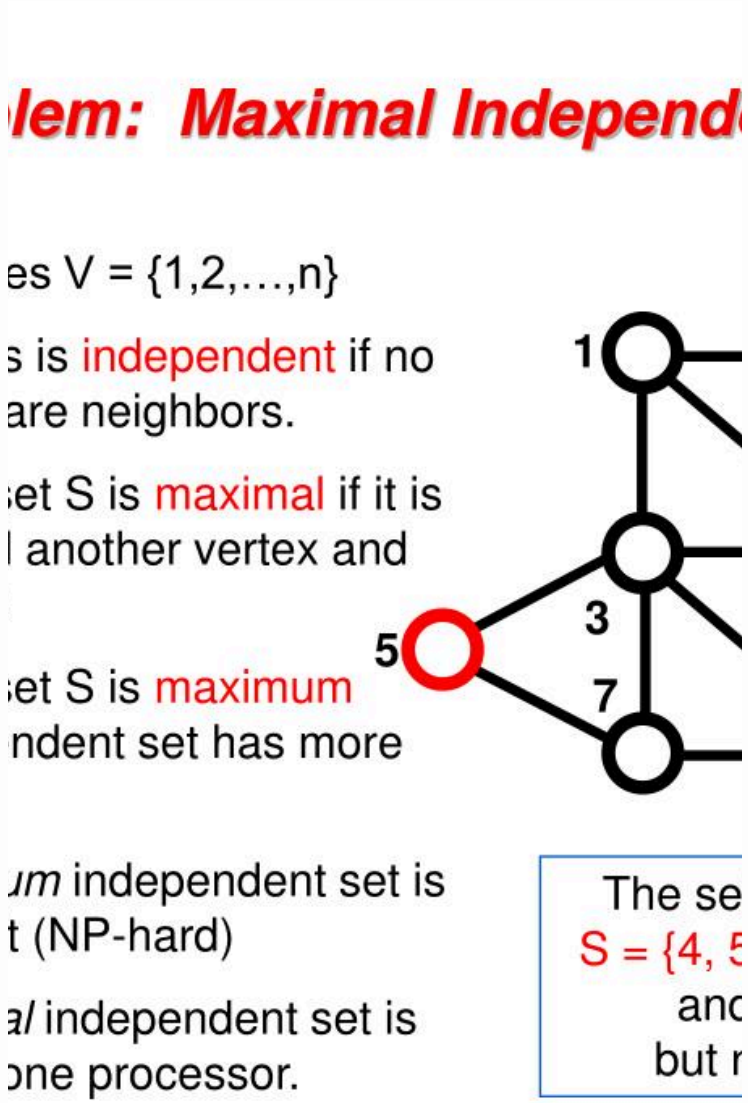
Ensuring Uniqueness and Non-Redundancy

FLEXIS checks for automorphisms and determines canonical forms to ensure the merged patterns are unique and non-redundant.

FLEXIS leverages the concepts of Core Graphs, Marked Vertices, and Core Groups to efficiently generate candidate patterns, outperforming existing approaches.

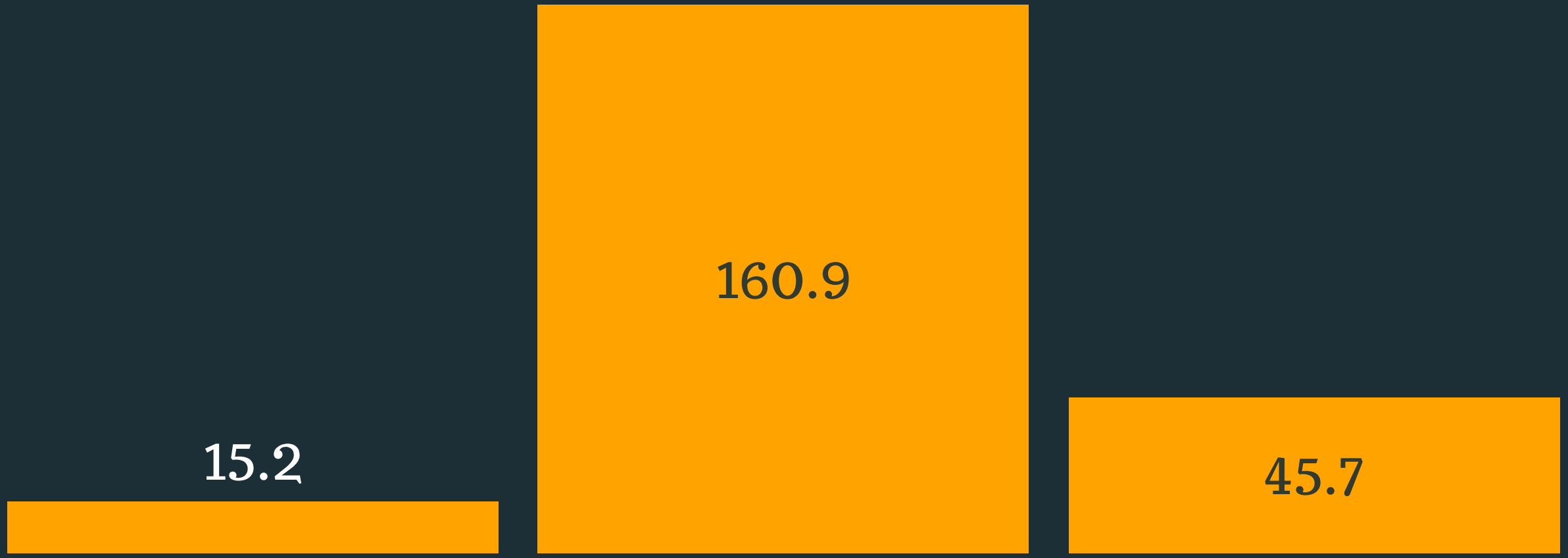
Maximal Independent Set (mIS) Metric

FLEXIS introduces a new metric called the Maximal Independent Set (mIS) metric, which aims to address the limitations of existing approaches like Maximum Independent Set (MIS) and Minimum Node Image (MNI). The mIS metric provides accurate pattern frequency counts while offering faster computation times compared to the computationally expensive MIS metric.



Comprehensive Experiments and Comparisons

Execution time (in seconds) for frequent subgraph mining on various datasets



The Generation Step: Candidate Pattern Generation

- ## Identifying High-Frequency Subgraphs

  The core focus of the Generation Step is to identify potential high-frequency subgraphs within the larger graph.

- ## Innovative Pattern Merging Approach

  FLEXIS introduces an innovative approach to generate k-vertex candidate patterns by merging two (k-1)-vertex patterns that have been previously identified as frequent.

- ## Generating Size-2 Candidate Patterns

  FLEXIS starts by generating all size-2 candidate patterns (edges) from the data graph.

- ## Identifying Frequent Size-2 Patterns

  FLEXIS uses a matcher (vf3matcher) to determine which size-2 patterns are frequent.

- ## Merging Frequent Patterns

  The frequent size-2 patterns are then merged to create candidate patterns of size 3, and the process continues for larger pattern sizes.

Core Graphs and Core Groups

Core Graphs

The FLEXIS approach introduces the concept of core graphs, which are obtained from a pattern graph by disconnecting one of its vertices along with its incident edges. The disconnected vertex is referred to as the marked vertex of the core graph.

Core Graph Isomorphism

Two core graphs are considered isomorphic if the core graphs with marked vertices excluded are isomorphic.

Core Groups

Core groups are defined as the set of all core graphs generated from patterns of the same size, grouped by isomorphic core graphs. A single core group is represented as a pair, where the key is a core graph without the marked vertex and the value is a list of all core graphs that are isomorphic to the key and to each other.

Extended Core Graphs

The paper also introduces the concept of extended core graphs, which handle the case where edges are also labeled. In this case, an edge (u,v) with label L(u,v) is replaced with two edges (u,w) and (w,v), where the newly introduced vertex w is assigned label L(u,v).

Merging (k-1)-Vertex Patterns to Form k-Vertex Patterns

Compute Core Groups

FLEXIS first computes core groups from the input set of (k-1)-vertex frequent patterns. A core group is a set of core graphs that are isomorphic to each other, except for their marked vertices.

Find Automorphisms

For each core group, FLEXIS considers pairs of core graphs and finds their automorphisms. Automorphisms are permutations of vertices that preserve the graph structure.

Merge Core Graphs

FLEXIS then merges the core graphs, potentially using the automorphisms found in the previous step, to generate k-vertex candidate patterns. The merge operation adds the marked vertices from the two core graphs to the common (k-2)-vertex component.

Conclusion

Frequent Subgraph Mining (FSM) Problem

The presentation covers the FSM problem, which involves identifying recurring subgraphs in larger graphs that exceed a predefined frequency threshold.

FLEXIS Approach

The FLEXIS approach introduces innovative techniques to address the limitations of existing FSM methods, including a new candidate pattern generation process and a flexible Maximal Independent Set (mIS) metric.

Improved Efficiency and Flexibility

FLEXIS aims to improve the efficiency and flexibility of the FSM problem by providing significant improvements in computational time and memory usage, while offering users control over the accuracy-speed trade-off.

Key Contributions

The key contributions of FLEXIS address the limitations of existing FSM methods, expanding the utility of graph mining techniques across a wide range of applications.