Engineering Degree Project

Bullying Detection through Graph Machine Learning

- Applying Neo4j's Unsupervised Graph Learning Techniques to the Friends Dataset

Author: Christoffer Eid
Author: Olof Enström
Supervisor: Elias Holmér
Lnu Supervisor: Arslan Musaddiq

Semester: Spring 2023
Subject: Computer Science

Abstract

In recent years, the pervasive issue of bullying, particularly in academic institutions, has witnessed a surge in attention. This report centers around the utilization of the Friends Dataset and Graph Machine Learning (GML) to detect possible instances of bullying in an educational setting. The importance of this research lies in the potential it has to enhance early detection and prevention mechanisms, thereby creating safer environments for students. Leveraging graph theory, Neo4j, Graph Data Science (GDS) Library, and similarity algorithms, among other tools and methods, we devised an approach for processing and analyzing the dataset. Our method involves data preprocessing, application of similarity and community detection algorithms, and result validation with domain experts. The findings of our research indicate that GML can be effectively utilized to identify potential bullying scenarios, with a particular focus on discerning community structures and their influence on bullying. Our results, albeit preliminary, represent a promising step towards leveraging technology for bullying detection and prevention.

Keywords: Bullying, Graph Machine Learning, Community Detection, Neo4j, Data Preprocessing, Similarity Algorithms, Friends, Neo4j, Unsupervised Learning, Anti-bullying.

Preface

The sum of this thesis project underscores a major academic trial, delving deep into the field of unsupervised Graph Machine Learning (GML) techniques. Through this exploration, it has become evident how such methodologies can be leveraged to address complex social phenomena, with a particular focus on bullying behavior. This process has not only enriched our understanding of the applications of GML but also opened ways to use these tools for societal benefit. Therefore, this project represents not only a significant accomplishment in our academic journey but also an example of how technology can contribute to advancements in the social sciences.

Our deepest appreciation goes to Elias Holmér and Henrik Leion, our project supervisors at Tietoevry, as well as Arslan Musaddiq, our project supervisor at Linnaeus University. Their expertise, knowledge, and guidance have been instrumental to the project's successful completion.

Special acknowledgment goes to Dr. Björn Johansson, Senior Lecturer at the School of Behavioural, Social, and Legal Sciences at Örebro University. His contributions to bullying prevention and intervention research, his commitment to interdisciplinary research, and his societal contributions extend the quality and depth of this study. His published works, such as *Pass or Fail?* [1] and *Dropping out of School* [2], alongside his roles in various committees and councils, exemplify his dedication to this complex field.

Lastly, profound appreciation is extended to Tietoevry for providing a conducive environment for this project, and to Friends for their generous data and expertise sharing, which significantly strengthened the project's outcome. Friends is an anti-bullying organization that investigates social relationships and bullying behavior among students and provides tools for preventing bullying [3].

Contents

1 Introduction

This bachelor's degree project aims to investigate the potential of Neo4j's unsupervised graph learning techniques in uncovering hidden patterns and clusters related to bullying behavior and victimization among students. By applying these techniques to the Friends dataset, a dataset collected from school surveys investigating social relationships and bullying behavior among students, the project seeks to contribute to the development of more targeted and effective anti-bullying programs and interventions. The introduction provides a brief overview of the project's background, motivating the reader to explore the subsequent chapters in greater detail.

1.1 Background

Bullying is a pervasive problem affecting millions of children and adolescents worldwide. It can have severe short-term and long-term consequences for both the victims and the perpetrators, impacting their mental and emotional well-being, academic performance, and overall quality of life. Developing effective strategies to prevent and intervene in bullying situations is a vital task for educators, parents, and policymakers [4, 5].

Unsupervised GML presents a compelling approach to understanding complex social structures and relationships, such as those seen in bullying dynamics. Using relevant datasets, unsupervised GML can have the potential to identify patterns or clusters that may indicate instances of bullying or victimization among students. A relevant example of such a dataset is the Friends dataset. The application of GML to this dataset might enhance the analysis and interpretation of bullying dynamics, demonstrating the utility of unsupervised GML in understanding and addressing complex social phenomena [6].

Neo4j is a graph database management system that provides a suite of graph analytical tools and algorithms within its GDS Library [7]. Utilizing these tools and algorithms, unsupervised GML analyses are conducted on the Friends dataset to reveal patterns and structures associated with bullying behavior. The insights gained, when combined with domain knowledge from behavioral scientists, contribute to the understanding of bullying dynamics.

1.2 Related work

The field of unsupervised machine learning has been successfully applied in various domains. For example, in medicine, Wardenaar et al. investigate the effects of co-morbidity in defining major depression subtypes associated with long-term course and severity. They employ a two-step approach: first, using supervised machine learning techniques such as regression tree analysis, lasso, ridge, and elastic net penalized regression to predict the persistence and severity of the major depressive disorder based on prior comorbid conditions, and second, performing unsupervised cluster analysis (k-means) on the predicted values obtained from the supervised machine learning models. This approach allows them to identify different clusters of patients and reveals patterns and correlations not seen before. The integration of information about prior comorbidity improves the classification of major depressive disorder subtypes and the prediction of illness persistence and severity [8].

Soft questions and human behavior analysis have been investigated in the context of well-being reports and their potential in validating clusters derived from unsupervised learning techniques. A study by Zhang et al. demonstrated the feasibility of applying machine learning methods to soft questions by incorporating findings from behavioral sciences. They analyzed the relationship between personality traits, depression, and subjective well-being in undergraduate students. By using domain knowledge and behavioral science research, the study was able to effectively address the challenges of working with soft questions and achieved high prediction accuracies for the subjective well-being of students. Their findings emphasize the importance of considering the complex nature of human behavior and incorporating domain knowledge when applying machine learning techniques to soft questions [9].

In the related work by Madurkar, GML methods are demonstrated to have the potential to extract information that may not be accessible through traditional machine learning techniques. By representing data as a graph, structural information can be embedded as features, which could lead to improved model performance in specific domains. Madurkar highlights that graph-based methods can capture the relational structure in the data, enabling a more comprehensive understanding of the relationships between entities. Additionally, the author showcases the applicability of graph-based methods to various tasks, such as node classification, link prediction, graph classification, clustering, graph generation, and graph evolution. The work emphasizes that networks can provide insights through the application of statistical analysis, models, and algorithms, despite their complexity [10].

These related works provide insights into the potential of unsupervised machine learning techniques in different domains and highlight the importance of incorporating domain knowledge and behavioral science research when working with soft questions in human behavior analysis.

1.3 Problem formulation

In this study, the primary objective is to investigate the application of unsupervised GML techniques in Neo4j for analyzing the Friends dataset and identifying patterns or clusters that may indicate bullying behavior or victimization among students. Furthermore, the study aims to explore how domain knowledge from behavioral scientists can enhance the analysis and interpretation of the results obtained using unsupervised GML techniques. Additionally, with the help of domain knowledge from behavioral science, the effectiveness of the machine learning approach in revealing hidden structures or relationships related to bullying behavior is evaluated.

The study recognizes inherent limitations, including the scope of the Friends dataset, the capabilities of the applied unsupervised GML techniques, and the knowledge boundaries of the collaborating domain experts. Even with these limitations, it is anticipated that this project will identify patterns or clusters linked to bullying behavior and victimization and evaluate the efficacy of the machine learning approach in finding structures or relationships. The project seeks to bridge the knowledge gap between existing research on bullying behavior and the implementation of unsupervised GML techniques for the analysis and interpretation of such data.

1.3.1 Research Questions

1. How can unsupervised GML techniques in Neo4j be applied to the Friends dataset to identify patterns or clusters that may indicate bullying behavior?

2. How can these patterns, combined with domain knowledge from behavior science, be used to reveal hidden relationships that can indicate a likelihood of bullying?

1.4 Motivation

Bullying is a pervasive issue that affects individuals and communities worldwide, with negative impacts on mental health and well-being [4, 5]. Despite extensive research conducted on the topic, there remains a need for more effective prevention and intervention strategies [11]. Utilizing unsupervised GML techniques to analyze data related to bullying behavior has the potential to reveal hidden patterns and structures that traditional analysis methods may not uncover. Incorporating domain knowledge from behavioral scientists can further enhance the understanding of these patterns and structures, providing valuable insights into bullying dynamics [12].

The motivation for this study lies in the potential for these insights to inform policymakers, educators, and mental health professionals in their efforts to address bullying behavior among adolescents. Moreover, the exploration of unsupervised GML techniques in this context may serve as a foundation for future applications in other domains, such as healthcare, transportation, or finance, where the identification of hidden patterns and structures can be crucial for informing policy and decision-making.

1.5 Scope/Limitation

In the project, the focus will be on analyzing the Friends dataset using unsupervised GML techniques in Neo4j to identify patterns or clusters related to bullying behavior or victimization. Friends and the behavioral scientist involved in the project will assist in the interpretation and evaluation of the results obtained from the analysis.

This project is limited by the size and quality of the Friends dataset, as well as the capabilities of the unsupervised GML techniques used.

No interventions or implementation of anti-bullying programs, based on the insights gained from the analysis, will be conducted. Moreover, it is acknowledged that the analysis of the Friends dataset may not be representative of bullying behavior and victimization in other contexts, such as different countries or age groups.

Finally, due to time constraints, all possible unsupervised GML techniques in Neo4j, will not be explored. The scope of this project is therefore limited to the analysis of the Friends dataset using selected unsupervised machine learning techniques in Neo4j.

1.6 Target group

The target audience for this work includes a diverse set of stakeholders interested in understanding and addressing bullying behavior and victimization among students.

Educational institutions and educators can leverage the insights gained from unsupervised GML analysis of the Friends dataset to inform the development of targeted policies and programs that promote positive social interactions and a supportive learning environment.

Data scientists and researchers in data science and social network analysis can also benefit from understanding the potential of unsupervised GML techniques in Neo4j to reveal hidden structures or relationships in datasets like Friends.

Additionally, anti-bullying advocates and organizations that aim to promote healthy social dynamics and prevent bullying can use the proposed graph-based methods and tools to better understand the factors that contribute to bullying behavior and victimization, thus informing the development of more targeted and effective interventions.

Tietoevry, the company that provided the project, is also interested in the findings and recommendations arising from the application of unsupervised GML techniques in Neo4j to analyze the Friends dataset and address bullying behavior.

1.7 Outline

This report opens with an Introduction, which outlines the scope of bullying and its implications and defines the research question and objectives. Subsequent to this, the Theory section delves into fundamental concepts such as graph theory, Neo4j, Cypher, the GDS Library, and GML. It further covers unsupervised GML, similarity algorithms, and community detection algorithms, along with embeddings and the Friends Dataset.

The Method chapter follows, detailing the literature review, collaboration with domain experts, data preprocessing, validation, unsupervised learning, and ethical considerations. The Implementation section then outlines the practical steps taken, including data preprocessing, the use of similarity algorithms, community detection and clustering, the development of the Normality and Positivity indices, and visualization techniques.

In the Results and Analysis chapter, a comprehensive examination of the study's findings occurs, including the comparison of various algorithms, the exploration of unweighted and weighted graph algorithms, and the validation with a domain expert. The subsequent Discussion chapter offers a summary of key findings, a comparison of clustering algorithms, a discussion on the influence of school climate, a reflection on the study's weaknesses and challenges, and possible future directions.

The Conclusion summarizes the report's main findings and future research paths. The report concludes with the References, a comprehensive list of all the information sources used throughout the study.

2 Theory

This chapter provides an overview of unsupervised GML with Neo4j, a graph database management system. The chapter begins with an introduction to the fundamentals of graph theory in Section 2.1. Subsequently, Neo4j is introduced in Section 2.2, where Cypher and the Graph Data Science Library are presented.

In Section 2.3, GML is introduced, followed by a specific focus on unsupervised GML in Section 2.4. Section 2.5 continues the exploration of evaluation metrics for unsupervised GML models. The topic of Similarity Algorithms is addressed in Sections 2.6 and 2.7. Continuing in Section 2.8 community detection algorithms are introduced, and the Fast Random Projection (FastRP) is presented in Section 2.9. Finally, the Friends dataset, which forms the empirical basis of the study, is presented in Section 2.10.

2.1 Graph theory

Before exploring the application of machine learning in Neo4j, it is essential to understand the basic concepts of graphs and how they are represented in Neo4j. A graph consists of vertices (or nodes) and edges (or relationships) connecting these vertices [13]. This is displayed in Figure 2.1, where a graph with six vertices and seven edges is shown. Graphs can model complex relationships and structures in various domains, such as social networks, biological networks, and transportation systems [14].

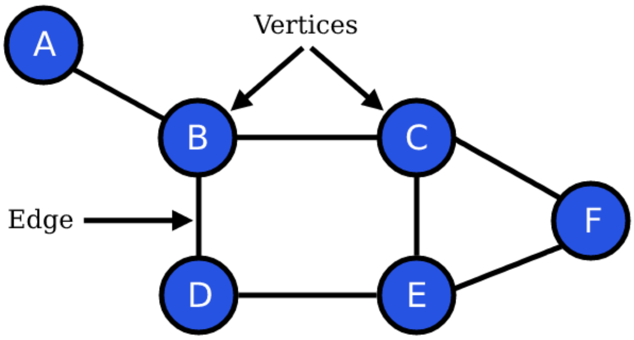

Figure 2.1: A graph with six vertices, labeled A-F, and seven edges.

One representation of a graph is through an adjacency matrix, a square matrix indicating the presence or absence of edges between vertices (see Figure 2.2). The adjacency matrix is an $n \times n$ matrix, where n is the number of vertices in the graph. Each entry A_{ij} in the matrix is either 0 or 1, depending on whether there is an edge connecting vertex i to vertex j. For undirected graphs, the adjacency matrix is symmetric, meaning that $A_{ij} = A_{ji}$ for all i, j [13]. Figure 2.2 represents the adjacency matrix for the undirected graph shown in Figure 2.1. The adjacency matrix can serve as input for certain graph algorithms and analyses [15].

Graph traversal algorithms, such as depth-first search and breadth-first search, are methods for systematically exploring the nodes and edges of a graph. These algorithms can help to understand the structure and connectivity of graphs, and they can be used to find paths, connected components, and other relevant graph features [16].

$$\begin{array}{cc} & \begin{matrix} A & B & C & D & E & F \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \\ F \end{matrix} & \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

Figure 2.2: The adjacency matrix for the graph in Figure 2.1.

In the context of machine learning in Neo4j, graph representations, such as adjacency matrices, and graph traversal algorithms play a critical role in understanding and processing graph data. By using these graph concepts, researchers and practitioners can develop and apply machine learning algorithms that take advantage of the information present in graph structures.

2.2 Neo4j

Neo4j is a graph database management system that stores and processes graph data. It employs a property graph model, wherein nodes and edges can have attributes (or properties) to store additional information [17].

2.2.1 Cypher

Neo4j provides a query language called Cypher, which allows users to perform graph operations and retrieve data from the graph. Cypher is a declarative language designed for querying graph databases and uses pattern matching to extract information from the graph. It is expressive and made to be readable, making it ideal for working with graph data [18].

An example of a graph consisting of users and their friendship relationships can be seen in Fig 2.3. The graph displays all nodes, labeled as `Person`, and all relationships, labeled as `KNOWS`. Cypher can be used to query the graph and retrieve information such as finding friends of a person. With the Cypher query declared in Fig 2.4, friends of Alice can be retrieved. The results, displayed in Fig 2.5, show each friend of Alice as nodes and their respective friendships as edges.

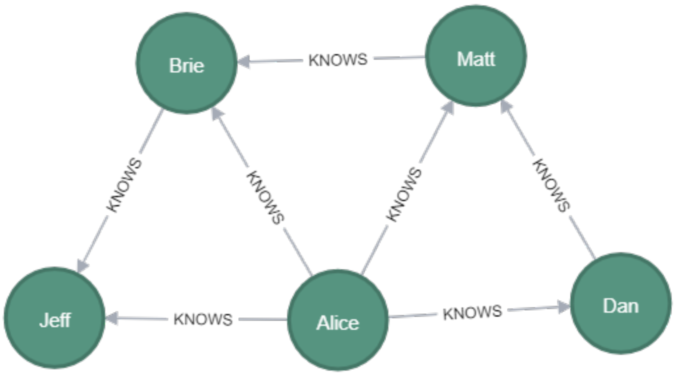

Figure 2.3: Graph representation of the example data.

```
MATCH (:Person {name: "Alice"})-[:KNOWS]-(friends)
RETURN friends;
```

Figure 2.4: Cypher query for finding friends of Alice.

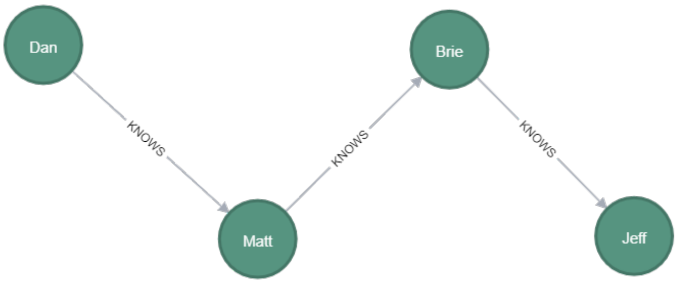

Figure 2.5: Graph representation of the Cypher query results for finding friends of Alice.

2.2.2 Graph Data Science Library

The Neo4j GDS Library is a collection of graph algorithms and analytics tools designed to work with the Neo4j graph database. It offers implementations of various graph algorithms, such as FastRP, Louvain, and LPA, which can be employed to process and analyze graph data stored in Neo4j. Additionally, the library incorporates other GML capabilities, including the possibility to classify nodes and predict relationships between nodes. These can be used in conjunction with graph algorithms to carry out machine learning tasks on graph data. By utilizing the GDS Library, researchers and practitioners can make use of graph algorithms and machine learning techniques to derive insights from graph data and build applications using Neo4j [7].

2.3 GML

GML is a subfield of machine learning that focuses on using graph structures to perform machine learning tasks. GML techniques typically involve processing and analyzing graph data to extract features, which are then used as input for machine learning algorithms. By taking advantage of the information present in graph structures, GML algorithms can achieve better performance on tasks such as node classification, link prediction, and graph classification compared to traditional machine learning approaches applied with naive graph representations or without considering the graph structure [19].

GML has gained popularity in recent years due to the increasing availability of large-scale graph data and the development of more efficient algorithms and computational resources. This subfield has been successful in a wide range of applications, including social network analysis, recommendation systems, and biological network analysis. Furthermore, GML techniques are effective in dealing with relational and structural data that is inherently present in graph representations, which traditional machine learning methods may struggle to capture [20].

2.4 Unsupervised GML

Unsupervised GML is an area of machine learning that focuses on learning from graph data without relying on labeled data. The goal is to uncover patterns, clusters, or structures in the graph data that can provide insights for various applications, such as identifying communities in social networks or detecting anomalies in systems [21].

2.5 Evaluating Unsupervised GML Models

Evaluating the performance of unsupervised GML models can be challenging due to the lack of ground truth labels. However, several approaches can be employed to assess the quality of the resulting clusters or communities:

1. Internal Evaluation Metrics: These metrics evaluate the quality of clustering or community detection without considering any external information. Examples include modularity, which quantifies the density of connections within communities compared to connections between communities, and conductance, which measures the number of edges that point outside a given community relative to the total number of edges in that community. High modularity and low conductance values are desirable for well-defined communities [22, 23].

2. External Evaluation Metrics: If ground truth labels are available, external evaluation metrics can be used to compare the clustering or community detection results with the true labels. Examples include the adjusted Rand index and normalized mutual information, which measure the similarity between two clusterings or partitions [24, 25].

3. Domain-specific Evaluation Methods: In some cases, domain-specific evaluation methods can be employed to assess the quality of the unsupervised learning results. For instance, input from behavioral scientists or expert judgment can be used to evaluate the relevance and meaningfulness of the detected communities or clusters in the context of bullying prevention and intervention strategies. As seen in the related work by Zhang et al [9].

2.6 k-Nearest Neighbors Algorithm

The k-Nearest Neighbors (kNN) algorithm is a similarity algorithm that operates on the principle of analyzing the relationship between nodes based on their proximity in the graph. The main idea is to determine the kNN of a node by evaluating the similarity between nodes using a distance metric [26, 27].

In the context of Neo4j, the kNN algorithm can be applied to find similar nodes within a graph, where the similarity score is calculated based on a distance metric. The algorithm is typically utilized for tasks such as recommendation systems, outlier detection, and data classification [27].

2.7 Node Similarity

Node Similarity is another algorithm employed in graph databases to measure the similarity between nodes. It is based on the Jaccard similarity coefficient, which quantifies the similarity between two nodes by calculating the ratio of the number of shared neighbors to the total number of distinct neighbors. In Neo4j, the Node Similarity algorithm is used to identify nodes with similar relationships to other nodes within the graph. The algorithm calculates the similarity score between each pair of nodes, where a higher score indicates a greater degree of similarity between the nodes in terms of their connected neighbors [28].

2.8 Community Detection Algorithms

Community detection algorithms are techniques in unsupervised GML that aim to identify groups of nodes in a graph that share similar properties, structures, or connections. These algorithms can be used to uncover patterns or relationships within the data [29]. This section discusses community detection algorithms: Louvain, LPA, and Weakly Connected Components.

2.8.1 Hierarchical Clustering

Hierarchical clustering is a method for grouping data points based on their similarity or distance in the feature space. It constructs a tree-like structure called a dendrogram, which represents the nested grouping of data points and the similarity levels at which groupings change. One of the main types of hierarchical clustering is agglomerative [30].

2.8.2 Agglomerative Clustering

Agglomerative clustering is a bottom-up approach to hierarchical clustering. It starts by considering each data point as a separate cluster and iteratively merges the closest pair of clusters until all data points belong to a single cluster or a specified number of clusters is reached. The merging process is guided by a distance metric such as Euclidean distance, and a linkage criterion, single, complete, or average. This approach results in a dendrogram that can be cut at different similarity levels to obtain the desired number of clusters [30].

2.8.3 Louvain Method

The Louvain method is a hierarchical community detection algorithm that optimizes modularity, which measures the strength of the division of a network into communities. Developed by Blondel et al., the algorithm is an agglomerative method that starts with each node as a separate community and iteratively merges communities based on a modularity gain measure until no further improvement in modularity is possible. Modularity gain measures the increase in modularity that occurs when two communities are merged. The higher the modularity gain, the more improvement is achieved by merging those communities. In the Louvain method, modularity gain is used to guide the clustering process, ensuring that the algorithm converges towards a partitioning with high modularity [31].

Levels in the Louvain method represent the hierarchy of communities at different scales. As the algorithm iterates, communities merge into larger communities, and each step corresponds to a different level in the hierarchy. Analyzing communities at different levels can provide insights into the structure of the network and the relationships between its elements. In Neo4j, the Louvain method is implemented in the GDS Library and allows for the adjustment of algorithm parameters, such as weights, the maximum number of iterations, the maximum number of levels, and tolerance [32].

2.8.4 Label Propagation Algorithm

LPA is a community detection method proposed by Raghavan et al., that works by iteratively propagating node labels throughout the graph. Each node is initially assigned a unique label, and during each iteration, nodes update their labels based on the majority label of their neighbors. The algorithm converges when no further label updates occur, and the communities are formed based on the final labels [33]. The LPA can be used in Neo4j through the GDS Library. Various algorithm parameters can be configured, such as the number of iterations, seed labels, and weights [34].

2.8.5 Weakly Connected Components

The Weakly Connected Components (WCC) algorithm is a method used to identify the connected components of a graph. In a directed graph, a weakly connected component is defined as a subgraph where a path exists between each pair of nodes when ignoring the direction of the edges. The algorithm concludes when all nodes have been visited, and the components are defined based on the groupings of the connected nodes. Through the GDS Library in Neo4j, the WCC algorithm can be used. The algorithm's parameters can be configured such as the handling of disconnected graphs, memory usage, and concurrency [35].

2.9 FastRP

Graph embeddings are techniques for representing nodes and edges in a graph as low-dimensional vectors in a continuous space. These embeddings can be used as input features for machine learning tasks involving graph data, such as node classification, link prediction, and clustering. By converting complex graph structures into low-dimensional representations, graph embeddings enable machine learning algorithms to process graph data more effectively and capture both local and global graph structures [36].

According to Chen et. al., FastRP is an efficient and scalable graph embedding technique that generates low-dimensional vector representations of nodes in a graph. FastRP is an unsupervised method based on random projection techniques, which captures both local and global graph structures. The FastRP algorithm works by projecting the graph's adjacency matrix onto a lower-dimensional space using random matrices. This process aims to maintain the graph's structure while reducing the dimensionality of the data [37]. In the context of Neo4j, FastRP can be implemented using the GDS Library, which provides an implementation of the algorithm. Once the embeddings are generated, they can be used as input features for machine learning tasks, such as classification or link prediction [38, 39].

2.10 Friends Dataset

The Friends dataset is a graph database containing survey responses from students across Sweden, covering topics related to bullying and social relations. The database was created using Neo4j, a graph database management system. The dataset is an essential resource for this study, as it provides insight into the experiences of students regarding bullying. Examples of questions in the dataset include questions on victimization: *Hur ofta har du haft svårt att somna under de senaste 2-3 månaderna?* (How often have you had trouble falling asleep the past 2-3 months?) as well as questions on their perception of the school climate: *I vår klass är det bra stämning* (In our class there is a good atmosphere) [40].

Dr. Johansson has played an active role in the development and utilization of the Friends dataset. In addition to his research work, he has also been involved in advising Friends on the formulation of questions to be included in the surveys, how to use the collected data, and how to extract meaningful insights from the results. This collaboration has contributed to Friends' understanding of the issues surrounding bullying and social relations, as well as informing the development of interventions and support strategies [12].

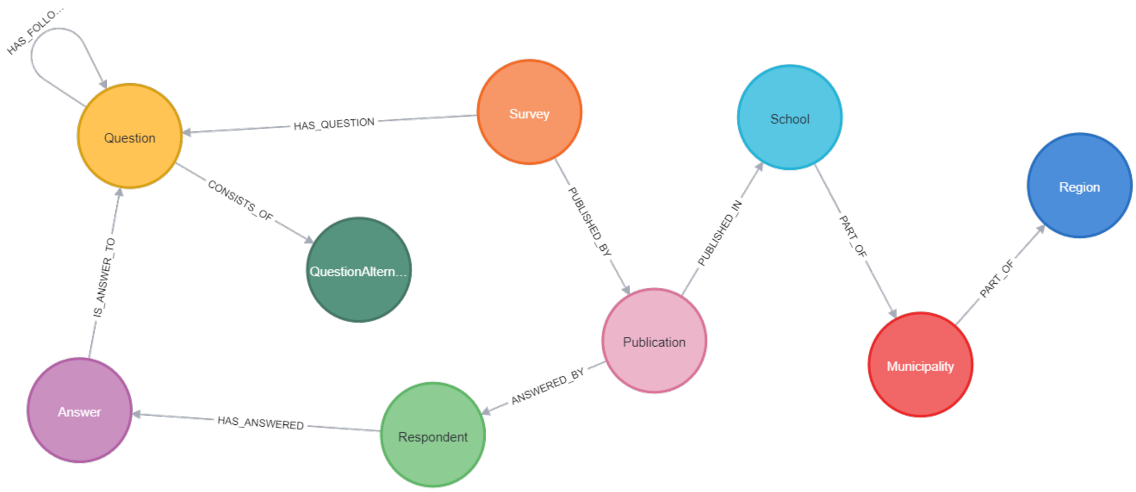

Figure 2.6: Schema of the Friends dataset in Neo4j

Figure 2.6 shows the schema of the Friends dataset. As a graph database, it consists of nodes and relationships, with each node representing an entity in the data and each relationship representing a connection between entities. The dataset includes information on schools, municipalities, regions, surveys, questions, respondents, answers, and publications.

3 Method

This study employs a two-sided methodology to address the research questions related to bullying using GML techniques. The first approach is research-oriented, featuring a literature review and collaboration with a domain expert. This approach aids in constructing a foundation for comprehending the theoretical facets of bullying, mental health, and GML algorithms. The second approach is experimental, composed of the application of assorted unsupervised machine learning algorithms for the analysis of the dataset and the identification of patterns or clusters potentially indicative of bullying behavior. Merging these two methodologies aims to discern patterns and relationships within bullying behavior, ultimately amplifying understanding of this phenomenon and offering valuable insights for future research and analysis in the field.

3.1 Literature review

To gather relevant sources for the project, searches are conducted on various scholarly databases and platforms, such as Google Scholar and ScienceDirect, using keywords related to bullying, mental health, GML, and algorithms as seen in Tables 3.1, 3.2, and 3.3.

Multiple sources on bullying and mental health [5, 4, 11], GML [7, 10, 14, 36, 41], and graph algorithms [37, 26, 33, 31] are explored to identify relevant insights and techniques applicable to the project.

Table 3.1: Bullying Search Queries

Query	**Website**	**Results**
Bullying impact on health	Google Scholar	582 000
-"cyber" -"cyberbullying" bullying survey unsupervised machine learning	Google Scholar	17 700
Mental health survey unsupervised machine learning	Google Scholar	53 500
"behavioral science" unsupervised machine learning	Google Scholar	1390

Table 3.2: GML Search Queries

Query	**Website**	**Results**
Graph Neural Networks review	Google Scholar	1 550 000
Graph Embeddings techniques	ScienceDirect	154 316
Validation of unsupervised learning	Google Scholar	721 000

Table 3.3: Graph Algorithms Search Queries

Query	**Website**	**Results**
Community detection algorithms	Google Scholar	3 630 000
Community detection in graphs	ScienceDirect	56 333
Centrality algorithms	ScienceDirect	618 083

3.2 Collaboration

In addition to the literature review, there is a collaboration with Dr. Johansson, a behavioral scientist, and domain experts from Friends, to deepen the understanding of the subject matter. Dr. Johansson provides valuable insights regarding the validity of the questions used in the study, verifying their alignment with established research and practices within the field. Friends contribute by explaining the data, the usual approach of Friends toward such projects, and the gap this study can fulfill. Furthermore, Dr. Johansson aids in interpreting the relationships within the Friends dataset, using his expert knowledge of bullying behavior dynamics and potential indicators of interest. This collaboration with domain experts enhances the analysis, thereby solidifying the foundation on which the subsequent data preprocessing and unsupervised learning techniques are applied.

3.3 Data Preprocessing

The primary focus of the data preprocessing phase is to transform the Friends dataset from a relational data model (see Figure 3.7) to a graph data model (see Figure 3.8), which facilitates the application of GML techniques in Neo4j. This preprocessing step is crucial for addressing our research questions, as it enables the application of unsupervised learning methods to identify patterns or clusters in the dataset that may indicate bullying behavior.

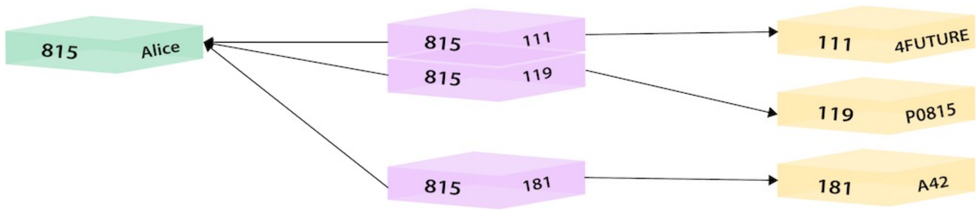

Figure 3.7: Relational representation of data. [42]

In the transformation of the dataset into a graph model, relational nodes — necessary in a relational database as tables but redundant in a graph database — are identified and eliminated [42]. These nodes are replaced by direct relationships between the relevant nodes, thereby enhancing the efficiency of the graph structure and facilitating analysis.

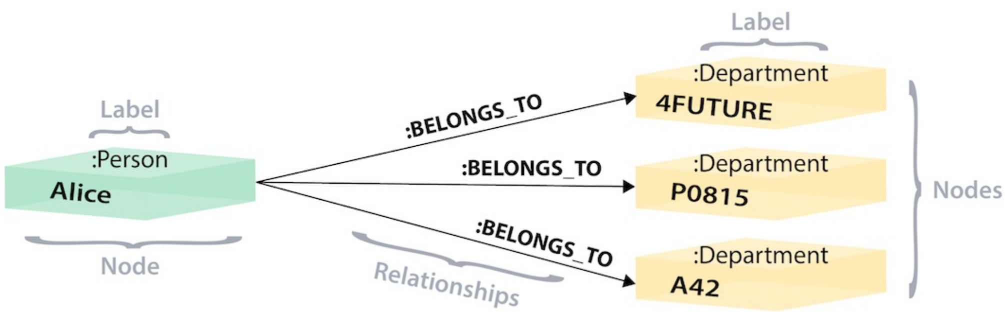

Figure 3.8: Graph representation of data. [42]

3.4 Unsupervised Learning

The hypothesis asserts that unsupervised GML algorithms are capable of revealing patterns and hidden relationships within the dataset, which could be indicative of bullying behavior. This influences the selection of GML algorithms, which allows for a systematic and structured exploration of the dataset.

3.4.1 Similarity Algorithms

The Node Similarity algorithm and the kNN similarity algorithm with FastRP are employed as similarity algorithms, designed to ascertain the degree of similarity between nodes within the graph [26, 28, 37].

Node Similarity calculates the similarity between nodes based on the structure of the graph surrounding them. By examining similarity scores, common behaviors and shared characteristics among users are detected, contributing to **RQ1** by highlighting potential bullying patterns [28].

FastRP is hypothesized to effectively project the multidimensional features of the respondents' relations to all its question alternatives. Concurrently, the kNN algorithm analyzes these projected features, to identify similarities among them and thereby find the most similar nodes within this high-dimensional space. This methodology is used to facilitate the detection of respondents that exhibit comparable answering patterns, which directly contributes to answering **RQ1** [26, 37].

3.4.2 Community Detection Algorithms

The Louvain algorithm, LPA, and WCC algorithm are community detection algorithms that are employed after the identification of similar nodes by the similarity algorithms. The Louvain algorithm is a community detection method that maximizes modularity. The discovery of clusters within the network aids in answering **RQ1**, as nodes within the same community are examined to understand their interactions and potential bullying indicators. LPA is another community detection method that spreads labels based on neighboring nodes. This approach uncovers nodes with similar patterns of interaction, providing insights for **RQ1**. Uncovering these clusters is key to identifying potential bullying communities. Lastly, WCC identifies sets of nodes in a directed graph that are interconnected but have no connections to nodes in other subgraphs. This method aids in identifying isolated groups or individuals within the network, which might be indicative of bullying behavior, providing further insight for **RQ1** [31, 34, 35].

While these algorithms are deployed to identify potential patterns (**RQ1**), interpreting the implications of these patterns (**RQ2**) necessitates a more qualitative approach. Hence, domain experts are engaged to analyze these patterns in conjunction with computational findings. This approach ensures an understanding of the data, taking into account both algorithmic results and the complex context and nuances of human behavior.

3.5 Result validation with domain expert

To ensure the validity and reliability of findings obtained through unsupervised GML techniques, results are examined and verified in collaboration with domain experts, including Dr. Johansson and Friends. Dr. Johansson's extensive knowledge in the field of behavioral science and the prior work of Friends on bullying verify the interpretation of patterns and clusters identified within the dataset. Their previous research on bullying [43] provides a reference for understanding the relationships between the variables.

This approach follows similar methodologies to those employed in previous studies such as Brock et al. [44] and Lappas & Yannacopoulos [45], which integrate domain expert input in the context of clustering and GML. The inclusion of expert knowledge alongside machine learning techniques aids in ensuring that the discovered patterns are meaningful and relevant.

As part of this collaboration, potential patterns and hidden relationships identified through unsupervised learning are critically discussed. Seeking insights from domain experts on these identified patterns and relationships ensures the results align with the current understanding of bullying behavior, and uncovers potential unseen correlations. The critical assessment of discovered patterns in the context of established knowledge on bullying behavior and victimization not only enhances the interpretation of the results but also provides insights for understanding the implications of the findings.

In this way, the involvement of domain experts in the validation process becomes an important step to understanding whether the application of unsupervised GML techniques yields meaningful and relevant insights, thereby contributing to a more comprehensive understanding of bullying behavior and patterns.

3.5.1 Normality Index

As part of the data validation strategy, a *Normality Index* is developed. This index provides a tool for analyzing student responses in a more systematic and scalable way than individual response validation. In addition to facilitating data validation, the Normality Index also serves as an instrument for visualizing the findings from the communities created by the unsupervised GML.

The idea of developing the Normality Index emerged from a conversation with Åsa Gustafsson, a domain expert from Friends [46]. Gustafsson highlighted that students who deviate from the norm tend to be more involved in bullying, either as perpetrators or victims. Furthermore, as noted by Dr. Johansson in personal communication [12], the self-reported answers of students to questions such as *Have you been bullied?* or *Have you bullied others?* cannot be taken as absolute truths. The Normality Index provides an objective method to gain insights into a student's responses without relying solely on their direct answers about their experiences.

Methodologically, the Normality Index improves the understanding and analysis of student responses to the survey. It facilitates the observation of the formation of different communities and enables the identification of trends and patterns in student behavior. This, in turn, contributes to the validity and reliability of our research findings.

3.5.2 Positivity Index

Parallel to the *Normality Index*, a *Positivity Index* is developed. This index offers a tool for analyzing student responses to school climate and victimization questions, quantifying the positivity expressed in their answers. Similar to the Normality Index, the Positivity Index serves not only as a utility for data validation but also as an instrument for visualizing the findings derived from the clusters created by the unsupervised GML.

The idea of the Positivity Index originates from a discussion with Dr. Johansson, where he underscored the potential of examining students' responses through a lens of positivity, as this could shed light on the underlying dynamics of school climate and bullying experiences. Considering the reality that self-reported answers may not completely reflect the true experiences of students, the Positivity Index provides an objective approach to assess students' perspectives without solely depending on their direct responses [12].

In terms of methodology, the Positivity Index complements and enriches the understanding and analysis of student responses. It assists in observing the appearance of distinct groups based on positivity levels and facilitates the identification of trends and patterns within student behavior. Furthermore, it allows for a comparison with the clusters derived from community detection algorithms, thereby offering an additional layer of validation. Consequently, this approach contributes to the validity and reliability of the research findings.

3.6 Ethical Considerations

In this study, ethical considerations are addressed as the data involves sensitive topics such as bullying and mental health in a school context. To ensure the ethical integrity of the study, necessary precautions have been taken.

All personally identifiable information has been removed from the dataset to protect the privacy and confidentiality of the students who participated in the surveys. The data is securely stored and accessed only by authorized researchers, with necessary precautions taken to prevent unauthorized access or disclosure.

The findings will be reported objectively and without revealing any information that could potentially identify the students or their schools. The research aims to contribute to a better understanding of bullying behaviors, ultimately aiming to minimize potential harm or distress caused by such behaviors.

4 Implementation

This chapter presents the systematic process of implementing community detection within the Friends dataset. Data preprocessing serves as the initial stage, setting the foundation for the subsequent application of similarity algorithms and community detection techniques. Emphasis is placed on detecting patterns and relationships inherent within the data, facilitated by indices such as the Normality and Positivity Index. The result of the workflow is visualized and arranged methodically for further analysis.

4.1 Data Preprocessing

The data preprocessing phase prepares the dataset for community detection. In this section, the conceptual and concrete aspects of the preprocessing are presented, and decisions made during this phase are highlighted.

4.1.1 Adding Relationships

In the process of adding relationships between nodes, the first step involves connecting the `Answer` nodes to their corresponding `QuestionAlternative` nodes. This is done by matching their `id` or `alternatives` properties, allowing answers to be directly linked to question alternatives. Once this connection is established, the `alternative` and `alternatives` properties from the `Answer` nodes are no longer needed and are subsequently removed. Relationships between `Respondent` and `Survey` nodes are then created to enable an analysis of the connections between survey participants and the surveys they completed.

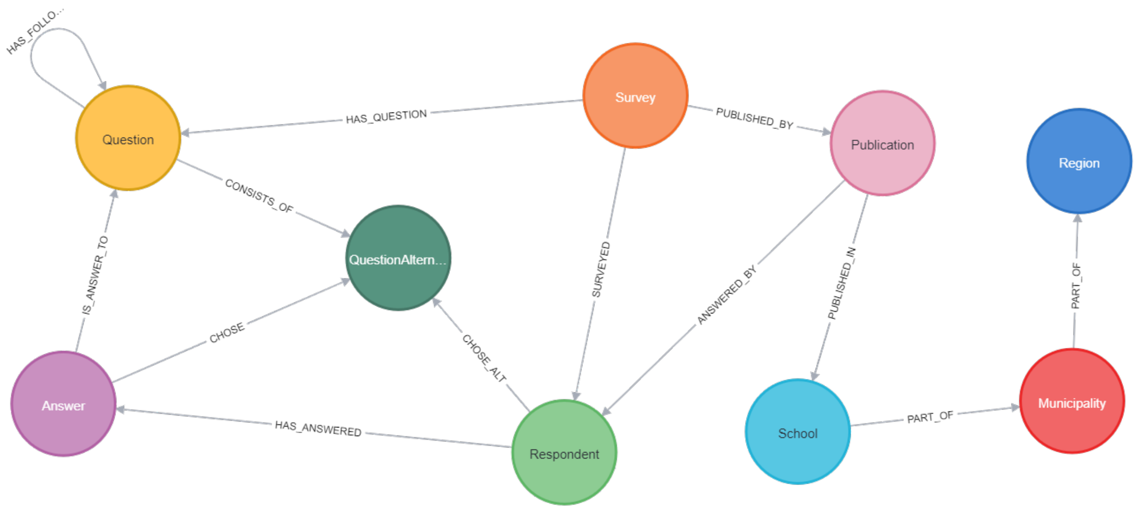

Figure 4.9: Friends dataset after preprocessing.

Additionally, a relationship is established between `Respondent` and `QuestionAlternative`. This new relationship connects respondents directly to the question alternatives they chose, further enriching the data model and facilitating the analysis of the relationships between respondents and their selected answers. By connecting respondents to question alternatives, it becomes possible to analyze the patterns of respondents' choices across different questions and surveys, providing insights into their preferences, opinions, and behaviors.

4.1.2 Updating Labels

There are two types of surveys in the Friends dataset, surveys for students and surveys for teachers. To further refine the dataset and distinguish between different groups of respondents, a `Student` label is added to the `Respondent` nodes associated with surveys made for students. This step allows for focusing the analysis on a particular group of respondents and a better understanding of their community structure.

4.2 Similarity Algorithms

As part of the larger implementation process, two similarity algorithms are employed, Node Similarity and kNN in combination with FastRP, to analyze the relationships between students and identify patterns in their responses. The implementation is carried out using the Neo4j GDS library in conjunction with Python programming language.

4.2.1 Node Similarity Algorithm

The first step is to project a graph using GDS, incorporating the `Student` label, the `QuestionAlternative` label, with the `position` property (default value of 0), and the relationship `CHOSE_ALT`. The Node Similarity algorithm is then run on the graph, creating the `NODESIMILARITY_POS_PROP` relationship, containing the `similarity` property, between `Student` nodes.

4.2.2 FastRP and kNN Similarity Algorithms

Next, the FastRP algorithm is run, which mutates the graph to include the `embedding` property containing the node embedding. See Table 4.4 for parameters the FastRP algorithm is configured with. The choice of parameters is based on recommendations from Neo4j, where a value in the range of 128 to 1024 is typical for the `embeddingDimension` and at least a few iterations with a non-zero value in an odd and even position is good for the `iterationWeights` [39]. Finally, the `randomSeed` is set to 42 for reproducibility.

Table 4.4: FastRP parameters.

Parameter	Value
embeddingDimension	128
iterationWeights	[0.8, 1, 1, 1, 1]
randomSeed	42

Once the embeddings are generated, the kNN similarity algorithm is applied to the newly mutated graph. Table 4.5 displays the parameters applied to the kNN algorithm, where the multiple values listed for `topK` imply the different values tested for the parameter. The other parameters are chosen to maximize accuracy. The valid range for the `sampleRate` is between 0.0 and 1.0, where a higher sample rate will increase the accuracy while requiring more memory and computing time. The `deltaThreshold` is set to 0.0 to allow the algorithm to run for as long as possible without stopping early [27]. As for previous algorithms, the `randomSeed` is set to 42 for reproducibility.

Table 4.5: kNN parameters.

Parameter	Value
topK	2, 5, 10
sampleRate	1.0
deltaThreshold	0.0
randomSeed	42

Similar to the Node Similarity Algorithm, the kNN algorithm writes the `KNN_SIMILARITY_POS_PROP` relationship between students, which contains the `score` property, indicating the similarity between students.

4.3 Community Detection

In this section, community detection algorithms are applied to both unweighted and weighted graph projections. These algorithms aim to explore potential patterns and relationships between students and question alternatives.

For the unweighted graph projection, which is based on the same information as the graph projection in section 4.2, the Louvain, Label Propagation, and Weakly Connected Components algorithms are employed. The Louvain algorithm is used to partition the graph into communities of nodes with high modularity, while the LPA assigns unique labels to nodes based on neighboring nodes' labels. The Weakly Connected Components algorithm is used to identify connected components within the graph, where nodes in a component are connected through a path of relationships, regardless of their direction. All three methods contribute to the identification and analysis of the structure of communities within the dataset.

In the weighted graph projection, the Louvain, Label Propagation, and Weakly Connected Components algorithms are utilized again. This time the graph projection includes the `Student` label and the `NODESIMILARITY_POS_PROP` relationship with the `similarity` property. These algorithms consider the similarity metric when detecting communities, and the results are returned to the graph with appropriate property names.

4.4 Normality Index

The Normality Index for a student s is calculated using the formula presented in Equation 1, where $\mathrm{a_i}$ is the number of responses given by students for option connected to question i, and $\mathrm{q_i}$ is the total number of responses given by students for question i. n represents the number of questions answered by the student.

$$\mathrm{NormalityIndex_s} = \frac{\sum_{i=1}^{n} \frac{\mathrm{a_i}}{\mathrm{q_i}}}{\mathrm{n}} \tag{1}$$

The Normality Index was developed to provide an indication or overview of how students responded to the survey without having to manually examine each answer option. It has also been used in graphs to indicate how the various communities have been formed based on different machine learning models. The Normality Index has not been fed into the models or used as any sort of ground truth; it has been exclusively employed for our understanding of the data and results.

The index reflects how students have answered in relation to each other. If a student's answers are, on average, similar to those of other students, they will score high on the Normality Index. Conversely, if a student's answers significantly deviate from the majority of other students, they will score low on the index.

The main Normality Index has been further divided into two subindices: the School Climate Normality Index (SCNI) and the Victimization Normality Index (VNI). These specialized indices allow for the evaluation of specific questions related to aspects of the school environment and student experiences, providing an understanding of the factors that contribute to bullying and victimization.

4.5 Positivity Index

The first step in the implementation of the Positivity Index involves determining whether a question is phrased positively or negatively. This is achieved by setting the property `is_positive` to 1 for positively phrased questions and to 0 for negatively phrased questions. For instance, the question *In our school, students are friendly to each other* is phrased positively and is assigned `is_positive` equal to 1. Conversely, the question *I feel afraid of other students at school* is phrased negatively and is assigned `is_positive` equal to 0.

The next step addresses inconsistencies in the `position` property of the question alternative nodes. The `position` property should be set such that a value of 0 for a positive question corresponds to the most positive answer, and a value of 0 for a negative question corresponds to the most negative answer. Some question alternatives did not initially have the correct ordering, and this step ensures the proper assignment of the `position` values.

Finally, with the `is_positive` and `position` properties correctly set, the Positivity Index can be computed for each student using the formula provided in Equation 2. The equation calculates the average score of positivity for a student s, where n is the total number of questions answered by the student, $position_i$ is the position of the chosen answer for question i, and $total\_positions_i$ is the total number of answer positions for question i. The $is\_positive_i$ variable indicates whether the question i is positively framed.

$$\text{PositivityIndex}_\text{s} = \frac{1}{n} \sum_{i=1}^{n} \begin{cases} 1 - \frac{position_i}{total\_positions_i}, & \text{if } is\_positive_i = 1 \\ \frac{position_i}{total\_positions_i}, & \text{otherwise} \end{cases} \tag{2}$$

4.6 Visualization

Four joint plots are generated to visualize the relationships between the SCNI and the VNI, coloring the data points according to their community labels assigned by different community detection algorithms. For each of these algorithms, the effect of a similarity index is accounted for. The plots provide a visual representation of the distribution of students in the space of the two Normality Indices and reveal the potential structure of different communities.

Table 4.6: Example data of a community in the spreadsheet

id	VNI	SCNI	I am	I attend	...
0x00	0,225514671	0,340739916	a boy	8th grade	...
0x01	0,340739916	0,225514671	non-binary	7th grade	...
...	...	...	...	...	...

The results are exported to a spreadsheet for further examination. The spreadsheet is organized with separate sheets for each Community, and each sheet contains columns for `respondent.id`, `respondent.schoolClimateNormalityIndex`, `respondent.victimizationNormalityIndex`, and `question.name`, with corresponding `questionAlternative.name` values chosen by `respondent` as seen in Table 4.6.

In addition to the joint plots, several other graphs are produced. These include a comparison of Node Similarity and kNN similarity, the number of students in different communities, communities based on the Positivity Index, the communities' average Positivity Index for all questions, and the average Positivity Index for each question. This allows for a comparison and examination of both algorithms and communities.

5 Results and Analysis

This section presents an overview of the results obtained from the machine learning algorithms implemented in the study, as well as the corresponding analysis to interpret these results. The graph algorithms are employed in the investigation, and they are compared in terms of the insights they generate regarding potential bullying patterns in the student population.

Through a combination of computational analysis and expert-driven insights, the aim is to examine the data, addressing research questions and revealing patterns indicative of bullying behavior. The results from each algorithm are explored in the subsequent subsections, providing a view of the findings and their implications.

5.1 Similarity Algorithms

In this section, the outcomes of the two similarity algorithms, Node Similarity, and kNN, are presented. Tables 5.7 and 5.8, provided below display a subset of the data that the algorithms generate, comprising three columns: Student 1, Student 2, and Similarity. Additionally, the mean similarity values for both algorithms are presented to enable further comparison and analysis.

The Node Similarity algorithm generates a similarity score between 0.0 and 1.0 for the students, indicating the extent of similarity between their choices of question alternatives. Like the Node Similarity algorithm, the kNN algorithm also generates a similarity score between students, ranging from 0.0 to 1.0. An unwanted side effect of the kNN algorithm is that it also creates relationships between students to question alternatives and between question alternatives themselves.

Examining the results, it is observed that some pairs of students exhibit high similarity scores, suggesting a strong correlation in their choices. Conversely, other pairs display low similarity scores, indicating a weak association between their selected question alternatives.

Table 5.7: Sample of relationships generated by Node Similarity.

Student 1	Student 2	Similarity
7829	7894	0.266667
7957	8020	0.089744
7957	8297	0.161905
8360	8423	0.310345
7894	8423	0.448718
...	...	...
52683	232167	0.594937
37448	232167	0.482353
16612	232167	0.524390
219349	232484	0.357143
210289	232484	0.337349
Mean		0.486277

Table 5.8: Sample of relationships generated by kNN.

Student 1	Student 2	Similarity
8171	8234	0.786535
8549	8675	0.816928
8486	8675	0.768934
8675	8738	0.802278
7829	8801	0.771719
...	...	...
122715	232103	0.764755
104596	232167	0.866336
165956	232167	0.875138
128179	232484	0.819436
127857	232484	0.82352
Mean		0.83421

5.1.1 Comparison of Node Similarity and kNN

When comparing Node Similarity and kNN algorithms, a difference in the distribution of similarity scores emerges, as depicted in Fig 5.10. The Node Similarity algorithm displays a wider distribution of similarity values, ranging from approximately 2% to 97%. In contrast, the kNN algorithm exhibits a distribution skewed towards the upper end of the range, with similarity values between approximately 57% and 99%.

This skewed distribution in the kNN algorithm is an inherent characteristic of the algorithm, not a result of the parameter choice. Unlike Node Similarity, which solely examines student-student relationships, the kNN algorithm also includes question alternative-question alternative and student-question alternative relationships. This wider scope of comparison results in the observed high similarity scores.

In terms of relationships generated, the Node Similarity algorithm creates a total of 15,586, while the kNN algorithm yields 3,156, a number marginally higher than the total number of students. This difference is directly influenced by the `topK` parameter. The Node Similarity algorithm is configured with a higher `topK` value than kNN, thus generating more relationships. This is favorable for the coming steps. Conversely, a lower `topK` value is selected for the kNN algorithm. Higher values of `topK` in the kNN algorithm lead to a thinner and higher distribution of similarity scores between students, which is considered less optimal for the purposes of this analysis. The values tested for the `topK` parameter can be observed in Table 4.5.

Therefore, caution is recommended when interpreting the results and comparing the two algorithms, considering the inherent differences in their respective similarity score distributions and the number of relationships generated.

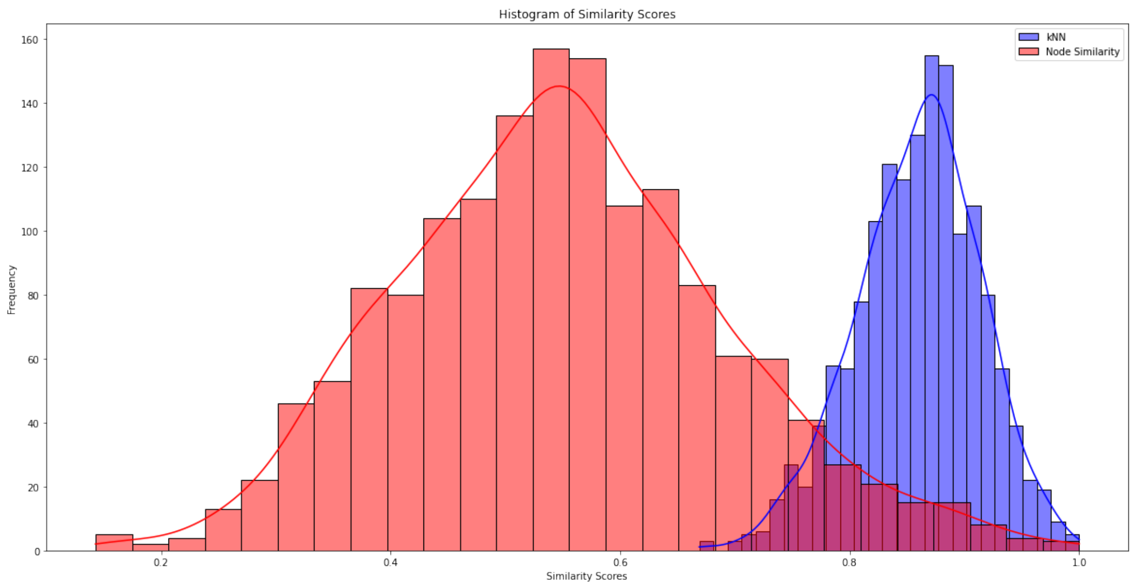

Figure 5.10: Comparison of similarity scores produced by kNN and Node Similarity.

5.2 Unweighted Graph Algorithms

The unweighted application of Louvain, LPA, and WCC to the graph data reveals a lack of distinct clusters, suggesting indistinct separation among student groups as per the school climate and victimization normality indices. Despite visual ambiguity, data structures are still identifiable, aligning with the research objective. Preliminary analysis suggests that unweighted graphs may not fully capture data intricacies, pointing to potential limitations of unweighted algorithms. This assertion is further supported by student distribution across clusters, which contradicts behavioral science research findings [40], hence advocating for the superiority of weighted algorithms in identifying victimized students.

5.2.1 Louvain Algorithm

Figure 5.11 demonstrates the application of the Louvain algorithm to the unweighted graph. The clusters are less defined, indicating that the separation among the student groups, as per the school climate and victimization normality indices, is less apparent. These indices serve as the axes for the visualization of clusters and can potentially highlight instances of bullying behavior. While the school climate index reflects the environmental conditions which are conducive to bullying [47], the VNI can offer insights into a student's potential to be a victim or perpetrator of bullying [48].

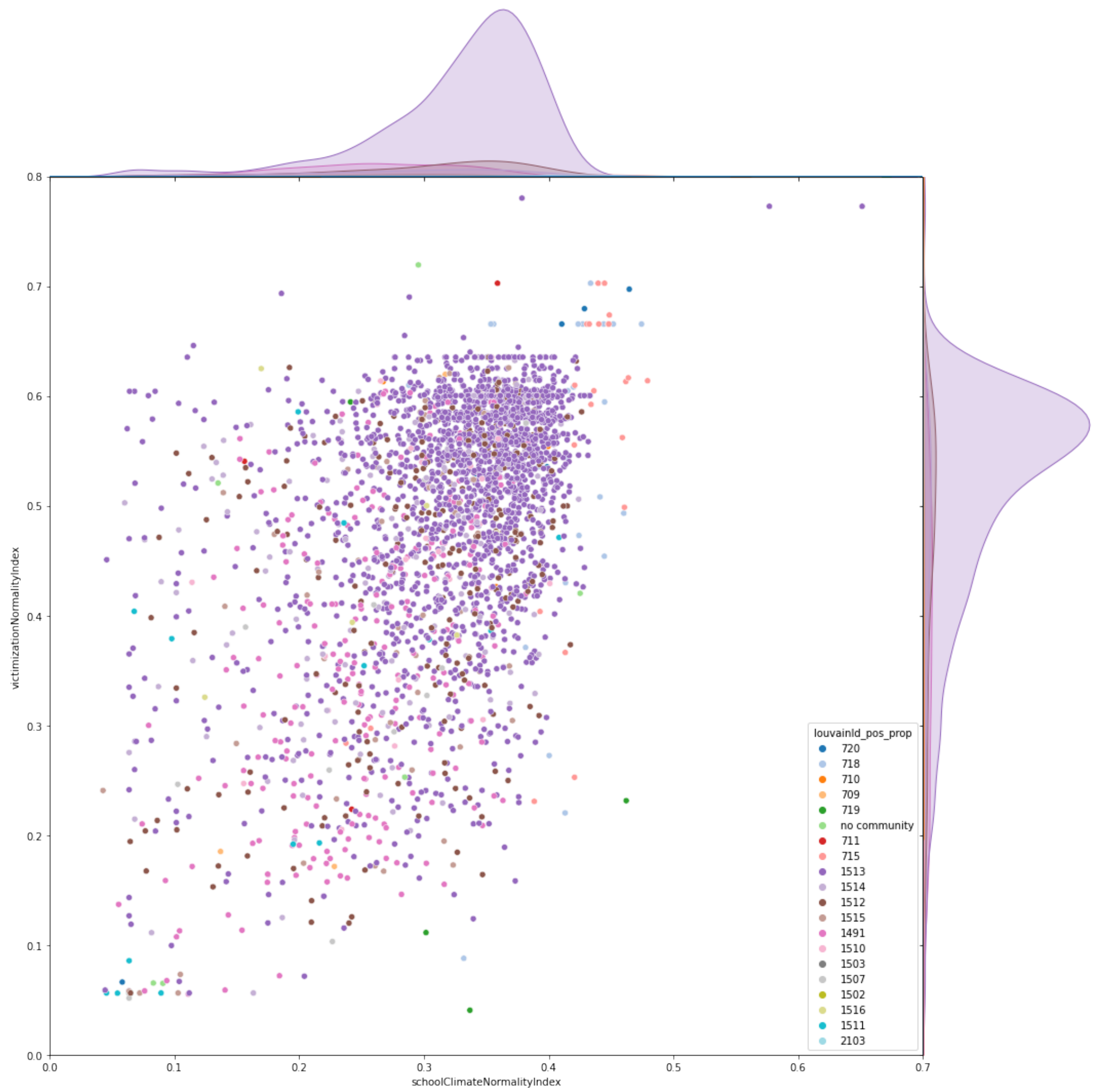

Figure 5.11: Clusters produced by unweighted Louvain

Although visual interpretation presents ambiguity, it is worth noting that the algorithm may still decipher meaningful structures within the data, even if they are not readily apparent. This notion aligns with the first part of our research question, examining how unsupervised GML techniques can be applied to the Friends dataset to identify patterns or clusters indicative of bullying behavior.

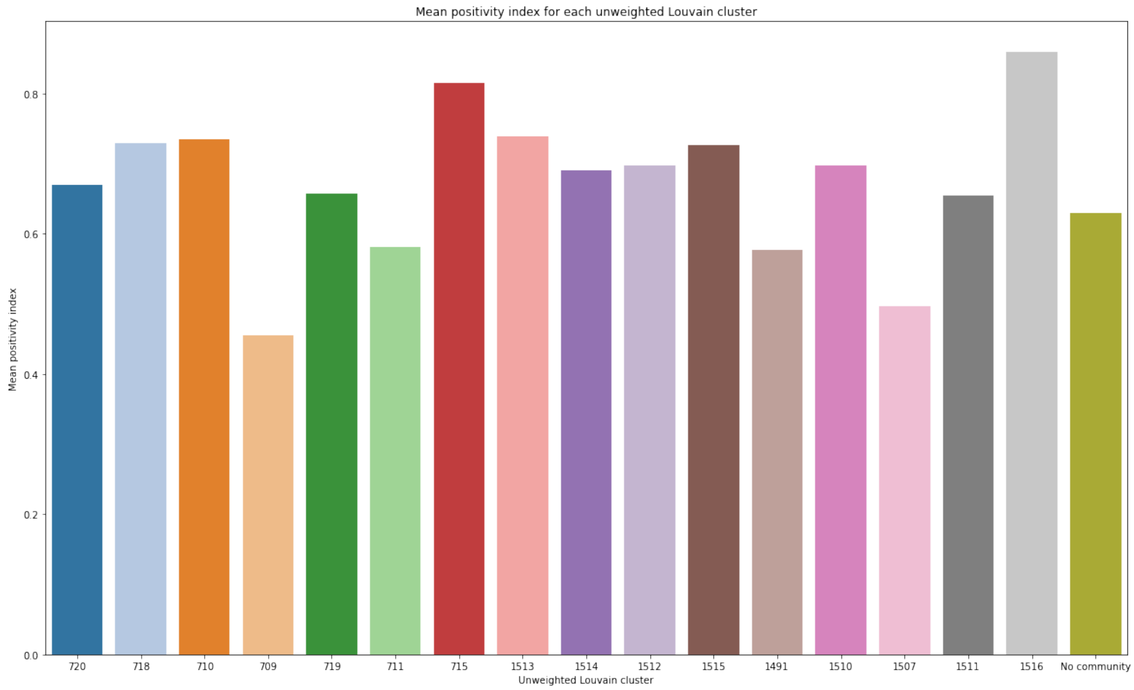

Figure 5.12: Mean Positivity Index for each unweighted Louvain cluster

The preliminary analysis suggests that the unweighted structure of the graph may not capture the data's intricacies, highlighting the potential drawbacks of relying solely on unweighted algorithms. The Louvain algorithm's ability to identify well-defined clusters significantly improves when incorporating the similarity score, underscoring the utility of weighted graph algorithms in revealing hidden patterns within the data. These findings and their implications on the second research question are further discussed in Section 5.3.

The results, depicted in Fig. 5.12 and 5.13, reveal clusters with a low mean Positivity Index. However, these clusters were small, accounting for less than 1% of the total students. This observation contrasts with behavioral science research which suggests that about 6-9% of students experience bullying [40]. This discrepancy provides evidence that unweighted Louvain may not accurately identify victimized students. It further reinforces the utility of incorporating weights or similarity scores into the Louvain algorithm, which the results suggest leads to improved detection of victimized students.

5.2.2 Label Propagation Algorithm

Applying the LPA to the unweighted graph, as shown in Fig 5.14, generates an alternative set of clusters. Though these clusters continue to appear less defined—indicating a less distinct separation among groups of students based on the school climate and victimization normality indices—there are subtle differences in how the algorithm classifies students.

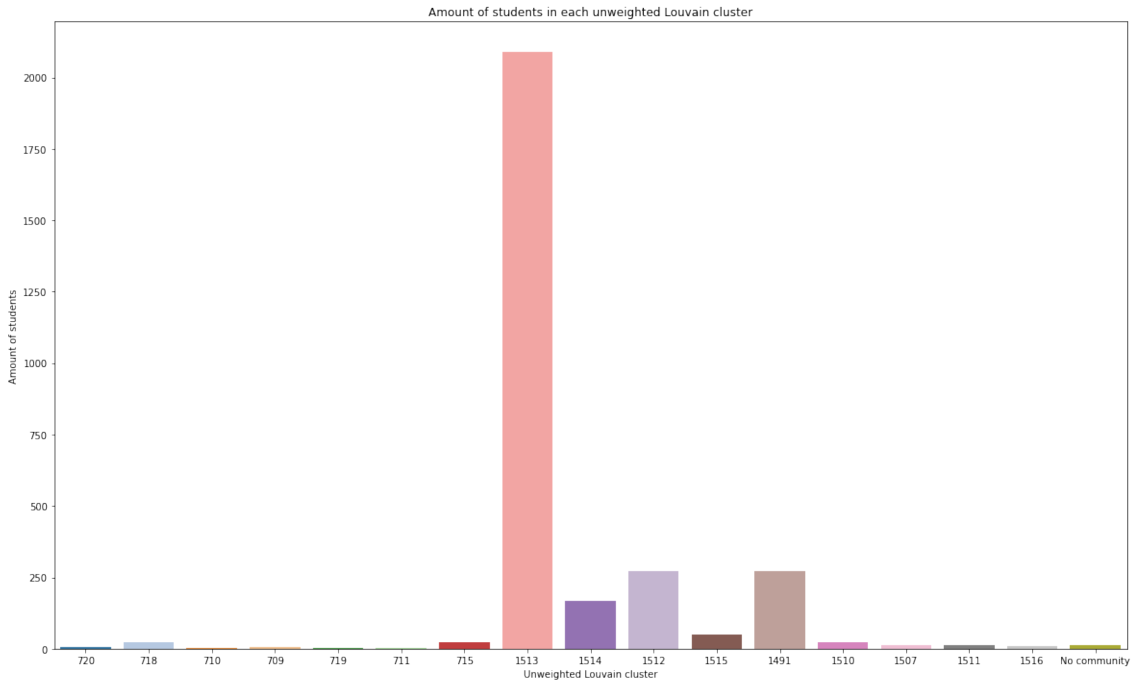

Figure 5.13: Number of students in each unweighted Louvain cluster

Despite the persistent visual ambiguity in the unweighted structure, it is vital to emphasize that the LPA might detect different structures within the data compared to the Louvain algorithm. This observation again highlights the research questions' first aspect, investigating how different unsupervised GML techniques applied to the Friends dataset can identify varied patterns or clusters potentially indicating bullying behavior.

Mirroring the findings from the Louvain algorithm, the initial analysis using the LPA suggests that the unweighted graph structure may not fully encapsulate the data's intricacies. However, the ability of the algorithm to identify more defined clusters is expected to improve when employing the similarity index, underscoring the potential of weighted graph algorithms to reveal patterns within the data. Further exploration of these findings' implications for the second research question will be done in Section 5.3.

As seen in Fig. 5.15 and 5.16, the results are consistent with those from the unweighted Louvain analysis. There exist clusters with a low mean Positivity Index, but these clusters comprise a small fraction (less than 1%) of the total student count, which contradicts behavioral science research findings [40]. This further verifies the hypothesis that weighted algorithms perform more effectively in identifying victimized students than their unweighted counterparts.

5.2.3 Weakly Connected Components

The WCC algorithm identifies 11 communities within the dataset. However, the distribution of students across these communities is uneven, as displayed in Table 5.9. Two communities contain more than one student, while the remaining nine communities each consist of a single student.

The largest community includes 2,958 students, while the other community with multiple

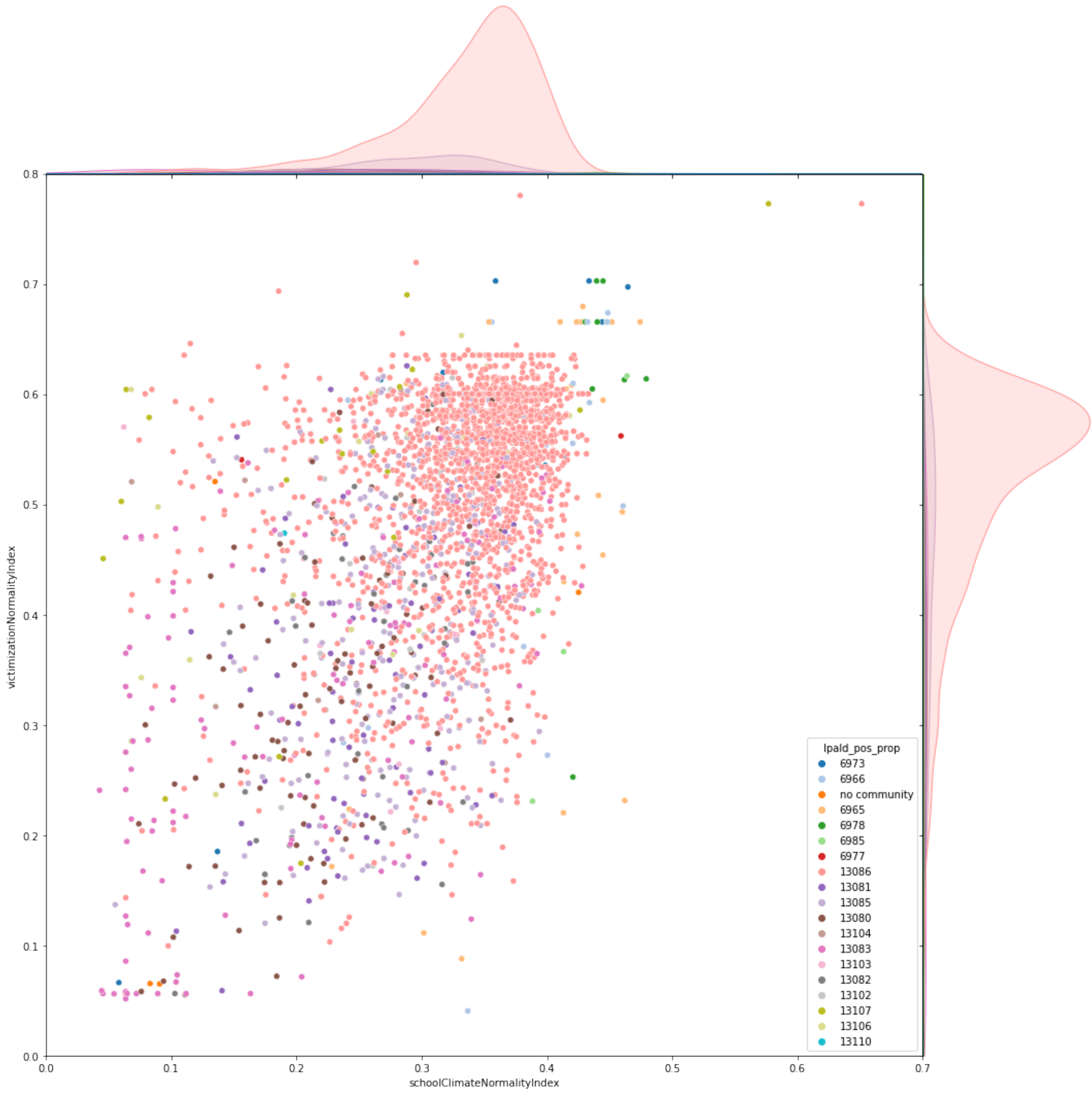

Figure 5.14: Clusters produced by unweighted Label Propagation

students comprises 72 members. This disparity in the distribution of students across communities could be interpreted in two ways. One perspective is that the WCC algorithm may have struggled to detect community structures within the dataset.

Alternatively, this could reflect actual disparities in student experiences and behaviors. Assuming that this is the case, even if we give that outcome perfect conditions where the community of 72 all represent victims of bullying, this group accounts for approximately 2.37% of the students which contradicts current research on the subject saying that 6-9% of adolescents are subject to bullying [40]. Thus, the WCC algorithm appears to underestimate the expected prevalence of bullying, suggesting its limited effectiveness in detecting distinct and meaningful community structures in this context. Consequently, these results are not further pursued. Instead, the focus shifts towards analyzing and evaluating other community detection methods to better understand the underlying patterns and relationships among students in the dataset.

Alternatively, these findings may reflect actual disparities in student experiences and behaviors. If this is the case, even when assuming ideal conditions where the entire com-

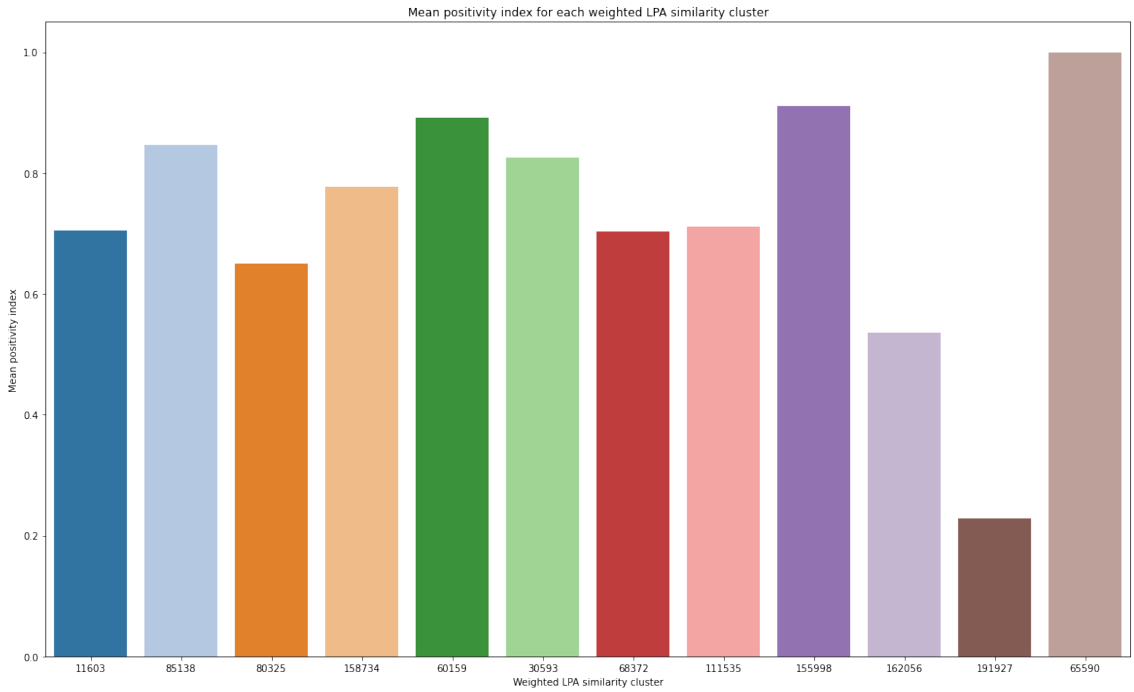

Figure 5.15: Mean Positivity Index for each unweighted Label Propagation cluster

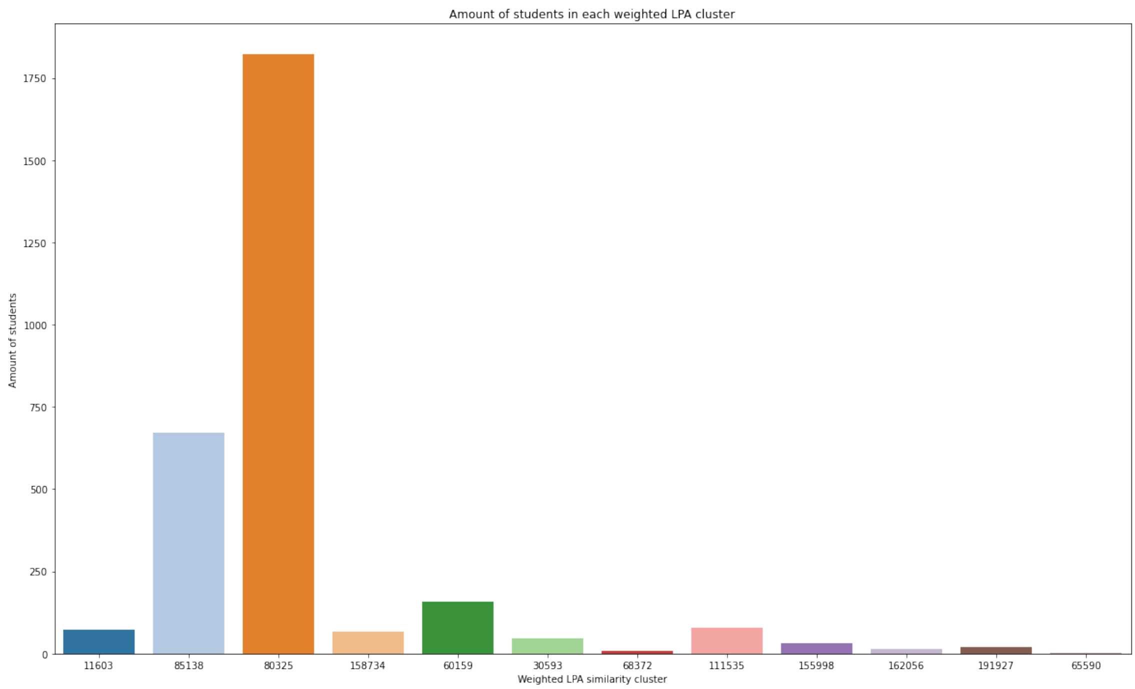

Figure 5.16: Number of students in each unweighted Label Propagation cluster

munity of 72 individuals represents victims of bullying, this group accounts for approximately 2.37% of the student population. This contradicts current research on the subject, which indicates that 6-9% of adolescents experience bullying [40].

Consequently, it appears that the WCC algorithm underestimates the expected prevalence of bullying, suggesting limited effectiveness in detecting community structures in this particular context. As a result, these findings will not be pursued further. Instead, the focus will shift towards analyzing and evaluating alternative community detection methods to gain a better understanding of the patterns and relationships among students in the dataset.

Table 5.9: Clusters produced by unweighted WCC.

WCC Community ID	Number of students
0	2958
700	72
1471	1
1472	1
2568	1
28	1
307	1
594	1
3063	1
3420	1
3712	1

5.3 Weighted Graph Algorithms

The addition of weights to the Louvain algorithm enables the recognition of distinct groups that align well with the school climate and victimization normality indices. Although the LPA shows an improvement with defined clusters over its unweighted variant, it still lacks the clarity produced by the Louvain algorithm. On the other hand, the Weighted WCC algorithm manages to identify communities, yet fails to find patterns supported by the field of behavioral science. These algorithms' performance, implications, and limitations are discussed in the following sections.

5.3.1 Louvain Algorithm

The Louvain algorithm, when applied to the weighted graph, unveils a more structured pattern of clusters, as shown in Figure 5.17. Unlike the results from the unweighted graph, the clusters in the weighted graph are more distinct. This finding points towards a clearer separation among groups of students based on the school climate and victimization normality indices.

The clusters are more distinctly grouped along the x-axis (school climate), rather than the y-axis (victimization). This could suggest a stronger influence of school climate on the formation of the clusters. Furthermore, the clusters are fewer in number compared to the results from the unweighted graph, which may indicate a higher level of aggregation in the weighted graph.

The added value of incorporating the similarity index into the graph structure is underscored by these findings. The weighted structure of the graph, in combination with the Louvain algorithm, seems to better capture the complexities of the data, potentially offering a more refined representation of student groupings as it relates to bullying indicators. This aligns with the first research question on how the identified patterns or clusters may indicate bullying behavior.

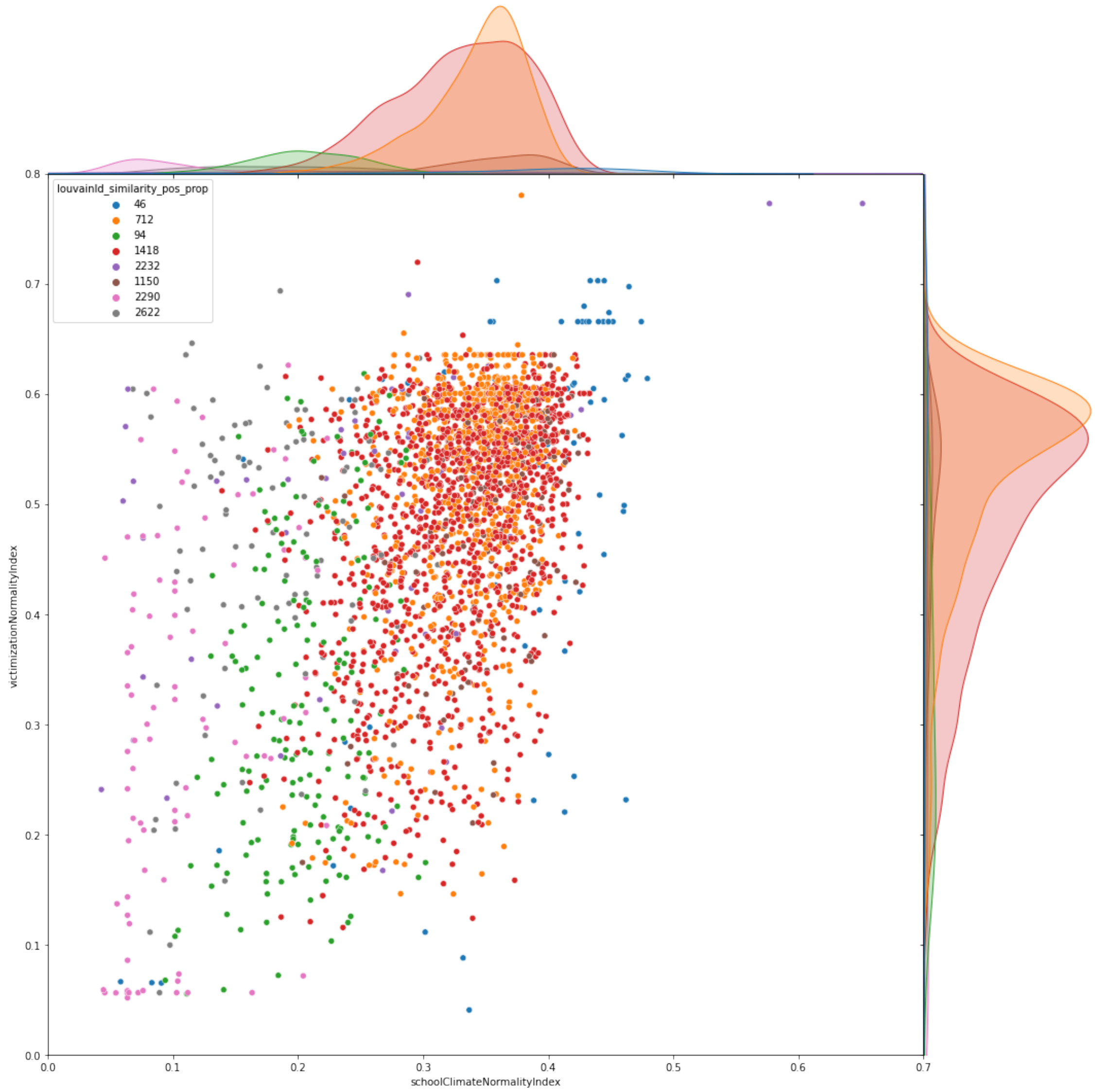

Figure 5.17: Clusters produced by weighted Louvain

Observing Figures 5.18 and 5.19, one cluster, denoted as `94`, stands out. This cluster comprises around 6-7% of the total students and displays a low positivity score. Examining the students' responses within this cluster reveals a pattern that aligns with known behavioral markers of victimization, suggesting a likelihood of these students being at risk of bullying. This pattern becomes apparent in Figure 5.20, where students from cluster `94` provide the most negative responses to a question on school climate, out of all the clusters. A similar pattern is observed in Figure 5.21, where students from cluster `94` display the most negative responses to a victimization question. These are merely two examples; more instances demonstrating this pattern can be found in the project's GitHub repository [49].

This observation corresponds well with the Friends paper which suggests a bullying prevalence rate of approximately 6-9% [40]. It appears that the weighted Louvain algorithm has been able to uncover a cluster signifying bullying patterns within the student population. This finding informs the second research question on how these patterns, combined with domain knowledge from the field of behavioral science, can be used to

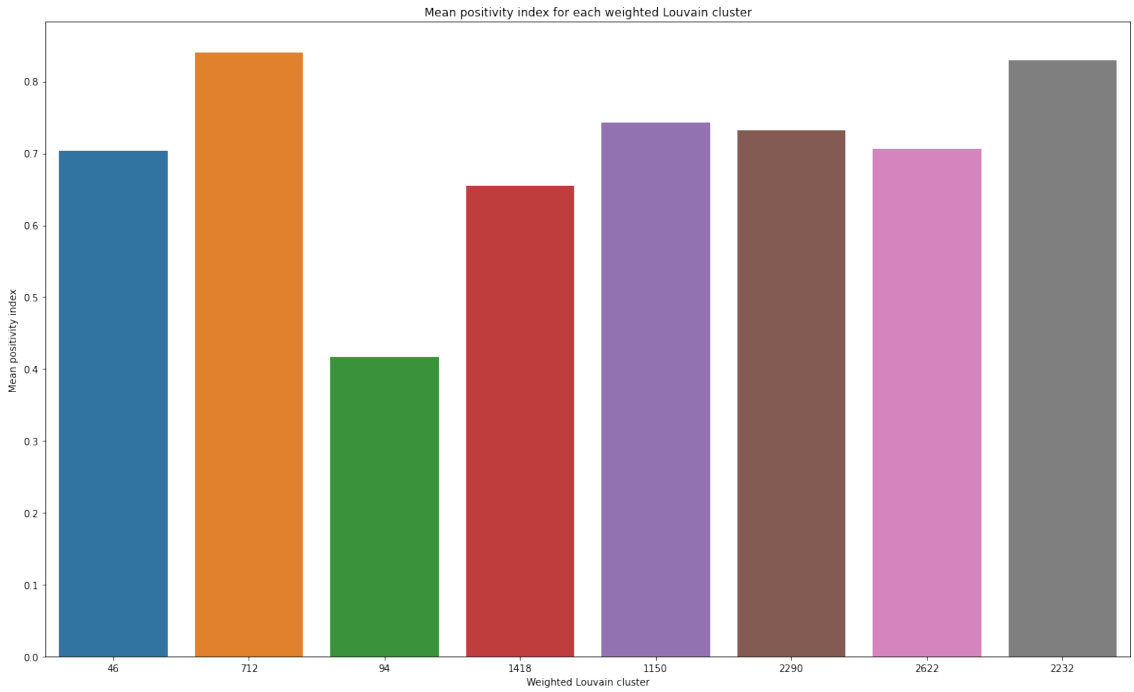

Figure 5.18: Mean Positivity Index for each weighted Louvain cluster

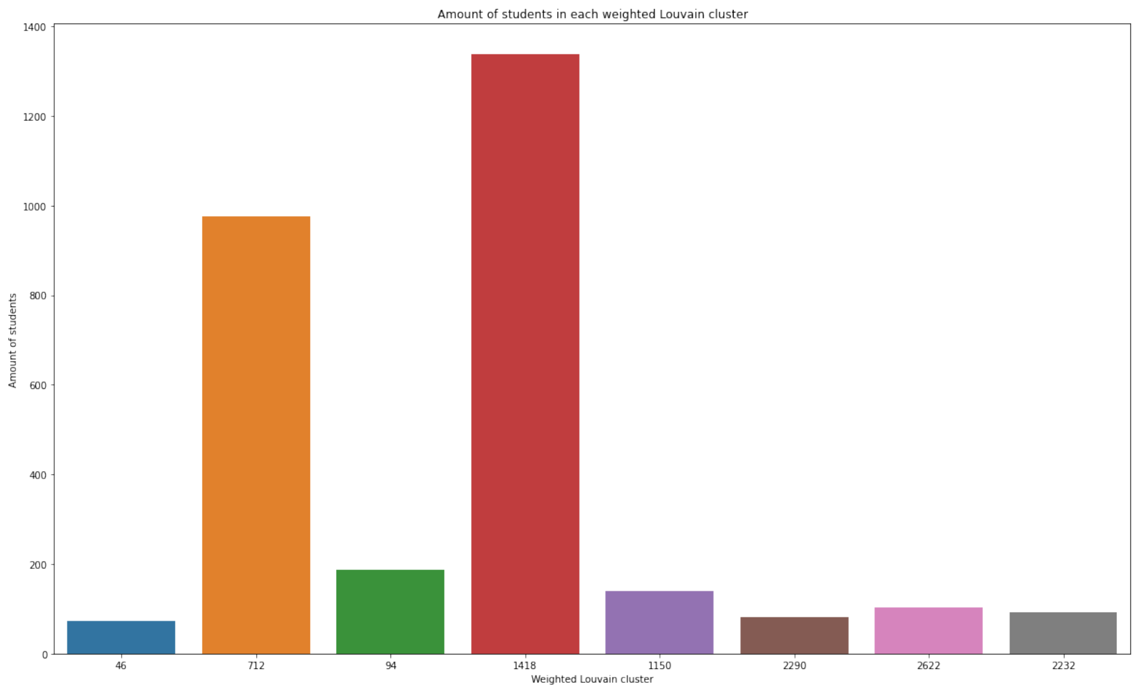

Figure 5.19: Amount of students in each weighted Louvain cluster

reveal relationships that can indicate a likelihood of bullying.

The improvements observed with the application of the similarity index in the Louvain algorithm suggest the potential of weighted graph algorithms in revealing patterns within the data. This finding also informs the second research question.

The implications of these findings will be further explored and compared with the results

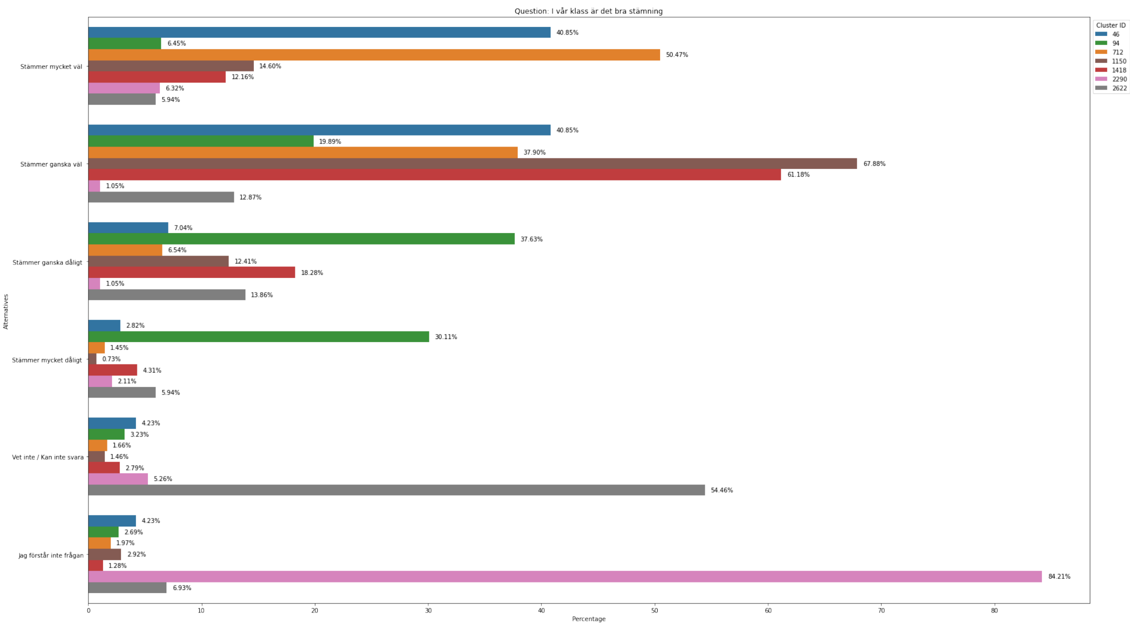

Figure 5.20: Weighted Louvain clusters response to a school climate question

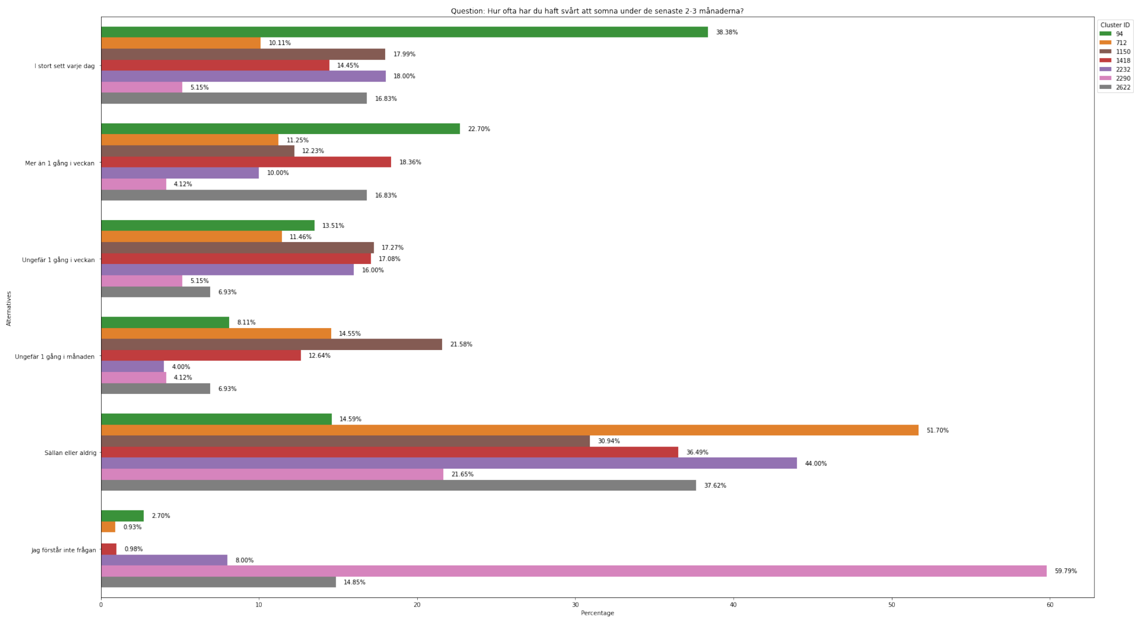

Figure 5.21: Weighted Louvain clusters response to a victimization question

from the LPA on the weighted graph in the following section. Specifically, it will be assessed how the differences in the clustering approach of the LPA might yield different insights into the patterns of bullying behavior within the student population.

5.3.2 Label Propagation Algorithm

When the LPA is applied to the weighted graph, the resulting clusters, shown in Figure 5.22, are more defined compared to those produced by the unweighted version of the algorithm. Despite this improvement, the clusters are not as distinct as those generated by the weighted Louvain algorithm. This result underscores the significance of the weighting process in enhancing the performance of unsupervised machine learning algorithms on this dataset.

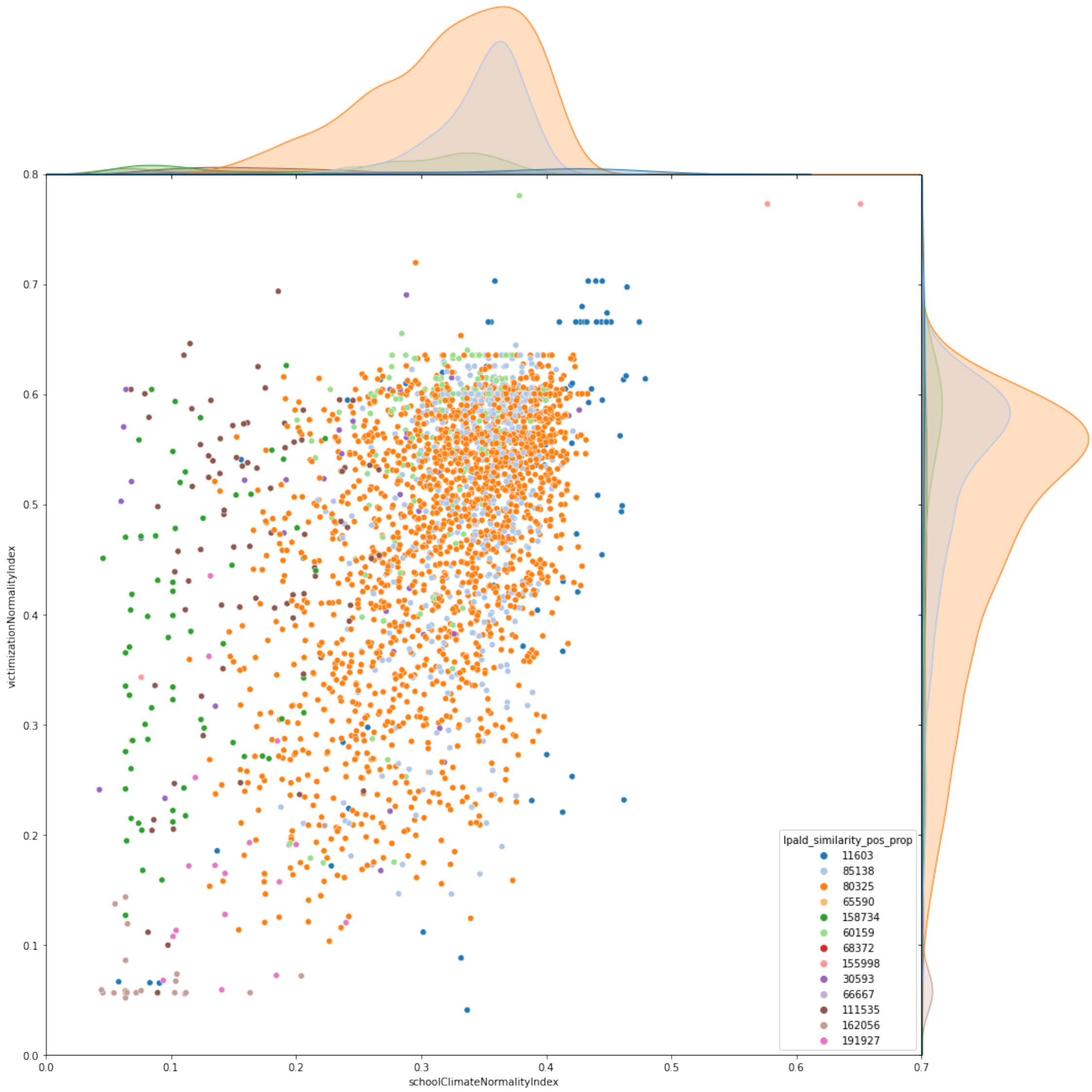

Figure 5.22: Clusters produced by weighted Label Propagation

Similar to the Louvain algorithm's output, the clusters in the weighted LPA are more distinctly grouped along the x-axis (school climate), although not as clearly. This reaffirms the earlier observation of a stronger influence of school climate on the formation of clusters. The spread along the y-axis (victimization) is also similar to the pattern observed in the Louvain algorithm's output, suggesting a consistent pattern across different algorithms.

These findings further strengthen the understanding of how patterns or communities may indicate bullying behavior, addressing the first research question. Additionally, they provide more insights into the value of incorporating the similarity index into our graph structure, informing the second research question on how these patterns, combined with domain knowledge from behavior science, can reveal relationships indicative of bullying.

Looking at Fig. 5.23 and 5.24, the clusters with a low Positivity Index, specifically clusters `162056` and `191927`, are smaller compared to other clusters. These smaller clusters represent less than 1% of the total student population, suggesting that the number

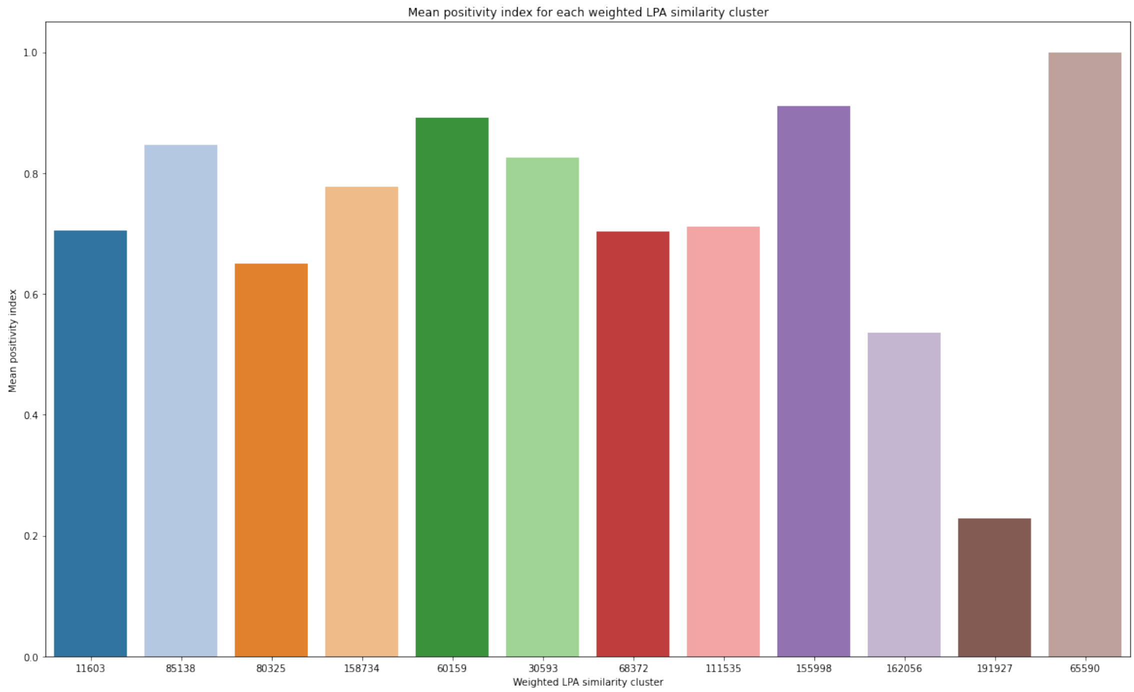

Figure 5.23: Mean Positivity Index for each weighted LPA cluster

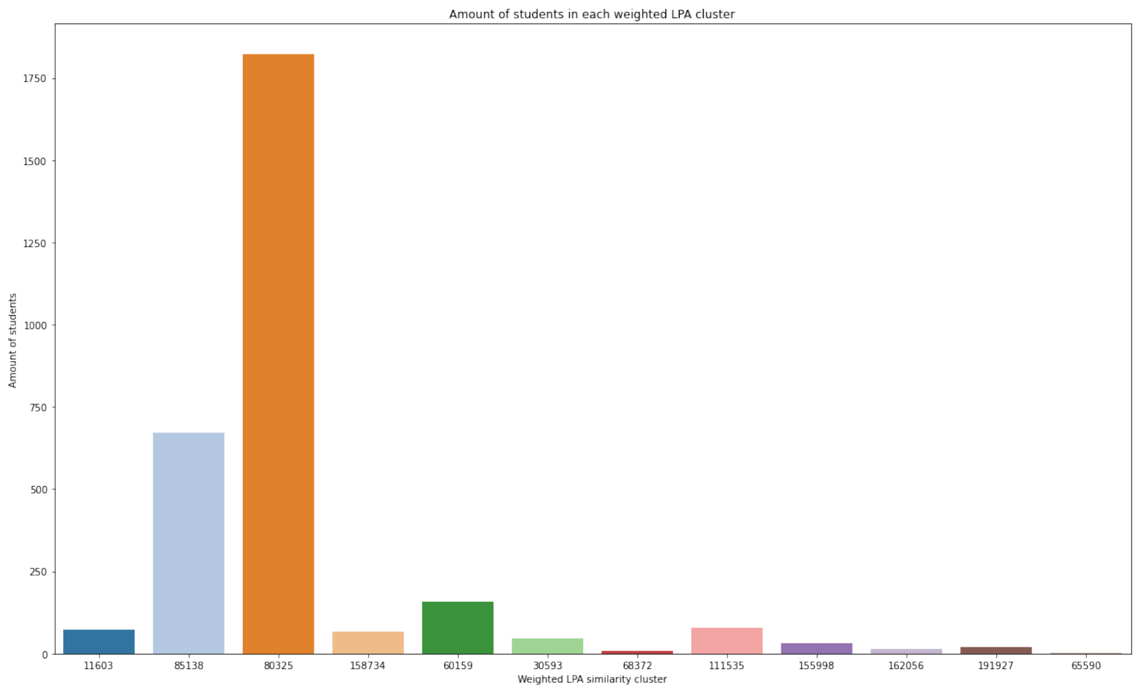

Figure 5.24: Amount of students in each weighted LPA cluster

of students potentially experiencing bullying, as indicated by this analysis, is lower than 6-9%, contradicting the findings in the field of behavioral science [40]. This discrepancy indicates that the LPA, in its current implementation, has not been able to identify communities that would be indicative of bullying patterns within the student population.

The implications of these findings and their comparison with the Louvain algorithm will be further explored in Section 6. This will include an assessment of how the different clustering approaches of the Label Propagation and Louvain algorithms might yield insights into the patterns of bullying behavior within the student population.

5.3.3 Weakly Connected Components

The Weighted WCC algorithm also identifies a total of 11 communities, with a distribution of students across communities that is similar to the unweighted WCC algorithm. As shown in Table 5.10, two communities contained more than one student, while the remaining nine communities consists of a single student each.

The larger community includes 2,958 students, whereas the other community with multiple students contains 72 members. This skew in the distribution of students across communities suggests that the Weighted WCC algorithm, like its unweighted counterpart, did not detect patterns supported by the field of behavioral science within the dataset. As a result, the outcomes of this algorithm are not examined further.

Table 5.10: Clusters produced by weighted WCC.

WCC Cluster ID	Number of students
74	2958
0	72
70	1
71	1
333	1
657	1
936	1
1223	1
1528	1
1885	1
2177	1

5.4 Validation with Domain Expert

The Positivity Index is utilized to further validate the machine learning results and provide an understanding of the students' perceptions regarding school climate and victimization. This index is devised in collaboration with behavioral scientists and mirrors the students' feelings towards school climate and victimization on a scale of 0.0 to 1.0. It incorporates expert advice by excluding non-answers, ensuring a higher accuracy in understanding the student's perspectives. A subset of the data generated using the Positivity Index algorithm can be observed in Table 5.11.

This added dimension of analysis verifies the machine learning algorithms' findings and highlights the data's patterns. A graphical representation, depicted in Figure 5.25, clusters the students based on the Positivity Index, allowing a comparison of the VNI and SCNI. The graph contains four subplots, each incrementing by one additional cluster from the preceding one, initiating with three clusters in the first subplot. Across all subplots, the bottom 8% of students, sorted by the lowest Positivity Index first, are preserved, as this is approximately the number of students who are bullied according to domain experts [40].

Comparing the communities generated by the LPA and Louvain algorithms to the clusters derived via the Positivity Index, differences are observed. The LPA algorithm matched the

Table 5.11: Sample of data produced by the Positivity Index algorithm.

Student	Positivity Index
7483	0.858156
7771	0.738095
7957	0.055556
...	...
232103	0.509259
232167	0.891473
232484	0.515152

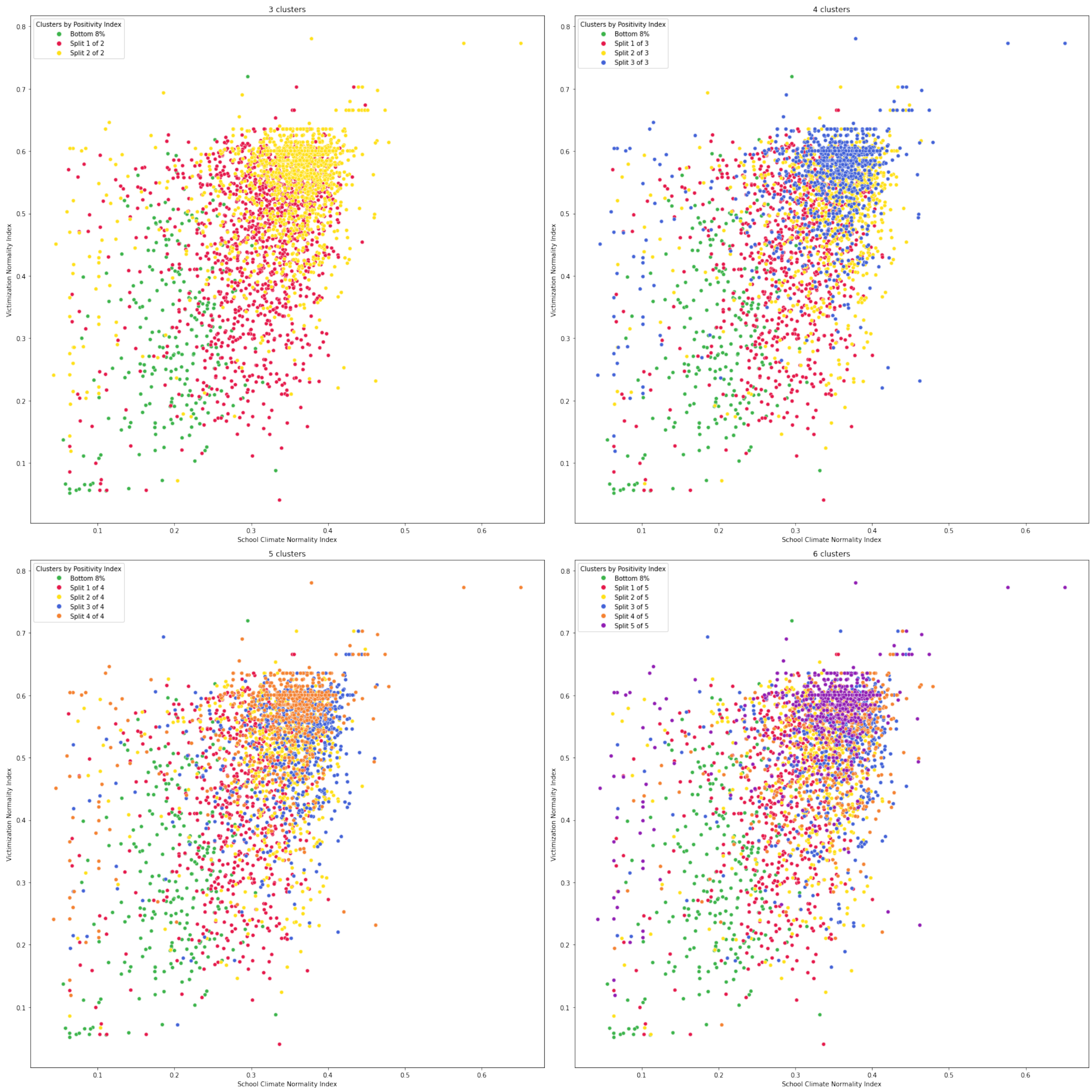

Figure 5.25: Four different splits of clusters based on the Positivity Index.

patterns reflected in the Positivity Index-driven clusters worse than the Louvain algorithm. There is a 78.3% overlap between students identified in cluster 94 by the Louvain algorithm and those in the bottom 8% cluster of the Positivity Index graph, signifying a strong correlation in terms of the students included and their respective cluster assignments.

This coherence suggests that the Weighted Louvain algorithm is more competent, compared to the other algorithms, at discerning patterns consistent with expert-driven insights. It also underscores the capability of the algorithm in categorizing students based on their attitudes toward school climate and victimization. This alignment, with the expert knowledge, indicates that the Weighted Louvain algorithm is a fitting method for uncovering patterns in the data.

6 Discussion

The discussion section presents an evaluation of the research outcomes, where graph algorithms are employed to distinguish bullying patterns in a school environment. The introduction of weighting to the Louvain and Label Propagation algorithms proves crucial for community detection. Nevertheless, algorithm performance exhibits variation, underscoring the necessity of algorithm selection specific to the dataset in question. Furthermore, the study reveals a pronounced influence of school climate on the formation of communities, aligning with prior research findings. Despite certain limitations, the research promotes subsequent studies to further examine bullying patterns, using machine learning techniques.

6.1 Summary of Key Findings and Research Questions Addressed

The research set out with the aim of exploring how graph algorithms could help identify patterns of bullying in school. The use of weighting in graph algorithms (Louvain and Label Propagation) proved to be vital in identifying meaningful communities, thereby addressing our first research question. More specifically, the weighted Louvain algorithm was successful in recognizing a collection of students at risk consistent with bullying prevalence rates found in other studies, answering our second research question.

6.2 Strengths of the Study

The main strength of the study lies in its innovative approach to detecting bullying patterns using graph algorithms. The weighting feature was effectively incorporated, improving the capacity of the algorithms to detect communities. Collaborating with domain experts for the development of the Positivity Index is crucial for the validation of the results.

6.3 Comparison of Clustering Algorithms

The Louvain algorithm's superior performance compared to the other methodologies in this study points towards the importance of considering the specific features and requirements of the dataset when choosing an algorithm. In particular, the Louvain algorithm's ability to identify a constellation of students at a risk similar to the bullying prevalence rates and in high comparability with the domain expert simulated results is a significant point in its favor.

6.4 Influence of School Climate on Cluster Formation

Across both algorithms, the clusters were more distinctly grouped along the x-axis (school climate), suggesting a stronger influence of school climate on the formation of the communities. This finding aligns with previous research suggesting that school climate plays a crucial role in student experiences of bullying.

6.5 Validation with Domain Expert

The Positivity Index, a new metric established in collaboration with domain experts, serves as a significant validation factor in this study. Of note is the Louvain algorithm's detection of cluster 94, which demonstrates a substantial overlap with the bottom 8% cluster in the Positivity Index graph. This result suggests the algorithm's efficacy in accurately assorting students based on their perceptions of school climate and victimization.

However, it's also important to consider that the interpretation and validation of results are dependent on the expertise and perspectives of the domain experts involved. Enlarging the pool of domain experts could lead to additional interpretations or observations. Different experts, especially those with diverse backgrounds and specialties, might spot other patterns, relationships, or insights within the data. Hence, future studies could benefit from a broader engagement with a diverse group of domain experts to further validate and enrich the analysis

6.6 Weaknesses and Challenges of the Study

While this study holds merit, it also faces multiple challenges. The LPA does not align with the behavioral science findings on bullying prevalence, suggesting a limitation in its suitability for this type of data. The WCC algorithm poses challenges as well, as it fails to generate meaningful community structures. These issues underscore the importance of deliberate algorithm selection and tuning. Revised two paragraphs version:

The scope of data used in this study imposes certain limitations. The analysis primarily focuses on students' responses, thereby overlooking potentially insightful data like teachers' survey responses or specific school attendance, a choice influenced by the study's particular focus, data processing constraints, and time limitations. These unexplored data within the current dataset could provide further insights and potentially influence the results, suggesting an area of potential expansion for future research if time and resource constraints are mitigated.

In addition to this, there exist other untapped data sources that could enhance the research. For instance, Friends' older records encompass more data related to student experiences, stored in a relational database. However, the significant task of transferring and adapting this data for use with the Neo4j graph database was not done due to the required effort and time considerations. Similarly, the integration of publicly accessible data about schools, such as the percentage of students eligible for upper high school or the number of certified teachers, could enrich the study. Although such integration could improve the understanding of bullying patterns, it was deemed outside of the present study's scope due to time constraints. Future studies may consider these avenues, given the availability of more resources and time.

These challenges underline the potential for more comprehensive insights into bullying patterns through a more diverse dataset and careful algorithm selection.

6.7 Potential for Further Analysis and Validation

This study provides findings while simultaneously highlighting additional approaches for analysis and validation of the metrics and methods used. The research primarily utilizes the Normality Index for data visualization and the Positivity Index for data analysis, opening an interesting line of exploration into the potential influence of interchanging these applications or exclusively employing either index. Moreover, future studies present opportunities for an examination of the validity and effectiveness of the Positivity and Normality Indices, possibly through statistical analyses or comparisons with other established metrics in bullying research.

An intriguing group of students identified in this study are those found within Louvain community `94`, a cluster associated with bullying due to its lower overall Positivity Index. However, some students within this community deviate from the trend, exhibiting unexpectedly high Positivity scores. These students, despite being in the *bullied* cluster, do not exhibit the typical negative indicators often seen in bullied students. Two main scenarios might account for this observation. One possibility is a misclassification by the algorithm, mistakenly placing a few *well-being* students into the *bullied* cluster, highlighting the potential for model refinement. Alternatively, the algorithm may have successfully identified students experiencing bullying that might have been missed by domain experts, as their signs of victimization are less evident. Regardless of the underlying reason, these students form a subject for further exploration, potentially accentuating insights into school climate perception, victimization, and the qualities of machine learning approaches in bullying research. Future studies should consider a more detailed investigation into these students and the factors influencing their positioning within the data.

The data preprocessing steps employed in this study could also be revisited and validated. Given the essential role of data preprocessing in shaping the dataset's suitability for GML and the resulting algorithm performance, a thorough examination of these steps and their optimization for GML is warranted. This could include exploring alternative preprocessing techniques or tuning the existing ones to better cater to GML.

Collectively, these potential areas for further analysis and validation could help refine the research methodology, increase confidence in the findings, and potentially uncover additional insights about bullying patterns within the dataset.

7 Conclusion

In the conclusion of our study, we encapsulate the findings and implications of our utilization of Neo4j's unsupervised GML methodology applied to the Friends dataset. We were able to discern patterns in bullying behavior and victimization. These insights are instrumental for educators and policymakers, enabling the development of targeted strategies for mitigating bullying. We suggest future avenues of research, such as examining other unsupervised learning methodologies, fostering interdisciplinary collaboration, and expanding the data pool for more robust findings. Our project underscores the significance of converging data science tools with domain-specific knowledge for addressing multifaceted social problems, such as bullying.

7.1 Main Findings

Our research employed the capabilities of Neo4j's unsupervised graph learning techniques to shed light on the complex patterns of bullying behavior and victimization. By applying these methods to the Friends dataset, we unlocked valuable insights that wouldn't have been otherwise identifiable. The success of this innovative approach has significant implications for the fields of data science and educational policy alike, illuminating the potential for these techniques to guide the creation of more effective anti-bullying strategies.

7.2 Future Work

Building upon the insights gained from this study, future research could delve into examining the performance of a wider array of unsupervised GML techniques in Neo4j, or even explore other machine learning tools or platforms to analyze bullying behavior and victimization. Such a multi-tool approach might enable the application of specialized algorithms or techniques not available in Neo4j, thereby contributing to a more diverse understanding of bullying dynamics within schools.

Further work could also explore the use of semi-supervised machine learning techniques. While this study has primarily focused on unsupervised techniques, incorporating elements of supervision could potentially enhance the precision and interpretability of our findings. Domain experts could be engaged to classify a subset of students, thereby creating *seed* labels. These labels could then serve as input for semi-supervised versions of algorithms like LPA. The addition of such a supervised element might allow for a more nuanced understanding of bullying patterns, as the algorithms can make use of both the label-free structure of the graph and the domain knowledge encapsulated in the seed labels. Enhancing interdisciplinary collaboration, similar to the engagement with behavioral scientists in this study, could further extend the interpretability of the results. This could potentially involve a broader group of domain experts, bringing in varied perspectives and knowledge.

Additionally, future work could shift focus towards evaluating interventions based on the insights obtained from this study, with the ultimate aim of mitigating bullying among students. A comprehensive analysis involving more advanced machine learning techniques or a fusion of different algorithms could further enrich these insights. This process might also necessitate further tuning and experimentation with algorithm parameters, including potentially modifying the projection for kNN to remove the relationship to question alternatives. Doing so would allow the kNN to only use properties for comparison and prevent the creation of similarity between students and question alternatives. Such an attempt could uncover more insights or enhance performance. It is important to note, however, that any changes to parameters or projections should be made with caution, as they could also introduce new challenges or complexities.

Furthermore, augmenting the volume of data used to train the unsupervised GML algorithms presents an interesting direction. This could involve integrating a broader range of data, such as older datasets or openly available school-related data, and could provide an assessment of how the volume of data influences the results and the performance of the algorithms. This approach could offer valuable insights into the scalability and robustness of the GML techniques, as well as provide a more nuanced view of bullying behaviors.

Collectively, these potential directions for future research — including more machine learning techniques, expanding the dataset, interdisciplinary collaboration, further investigation of the results, and evaluation of interventions — carry the potential to contribute to the body of knowledge surrounding bullying dynamics within schools.

7.3 Final Thoughts

Our work underscores the transformational potential of unsupervised GML techniques in Neo4j for analyzing intricate social phenomena such as bullying. The success of our research suggests that the combination of data science techniques with domain expertise, as demonstrated by our collaboration with behavioral scientists, offers a powerful approach to understanding and addressing complex issues. This project has made a valuable contribution to the field and laid a strong foundation for future explorations in this domain.

References

[1] B. Johansson, E. Flygare, and K. Hellfeldt, *Pass or fail?: A study concerning how experiences of bullying, truancy and social relations influence pupils' academic performance*, 2018. [Online]. Available: http://urn.kb.se/resolve?urn=urn:nbn:se:oru:diva-64640 [Accessed: 2023-04-27]

[2] B. Johansson, *DROPPING OUT OF SCHOOL: a systematic and integrative research review on risk factors an interventions*, ser. Working Papers and Reports, 2019. [Online]. Available: http://urn.kb.se/resolve?urn=urn:nbn:se:oru:diva-77853 [Accessed: 2023-04-27]

[3] Friends, "Zero tolerance against bullying." [Online]. Available: https://friends.se/en/what-friends-does/ [Accessed: 2023-05-23]

[4] M. Warner, "The impact of bullying on mental health," March 2021. [Online]. Available: https://med.uth.edu/psychiatry/2021/03/12/the-impact-of-bullying-on-mental-health/ [Accessed: 2023-04-03]

[5] R. Armitage, "Bullying in children: Impact on child health," *BMJ paediatrics open*, vol. 5, no. 1, 2021.

[6] B. C. Molokwu, S. B. Shuvo, N. C. Kar, and Z. Kobti, "Node classification in complex social graphs via knowledge-graph embeddings and convolutional neural network," *Computational Science – ICCS 2020*, vol. 12142, pp. 183 – 198, 2020.

[7] Neo4j, "The neo4j graph data science library manual v2.3," 2023. [Online]. Available: https://neo4j.com/docs/graph-data-science/current/ [Accessed: 2023-04-10]

[8] K. J. Wardenaar, H. M. van Loo, T. Cai, M. Fava, M. J. Gruber, J. Li, P. de Jonge, A. A. Nierenberg, M. V. Petukhova, S. Rose, and et al., "The effects of co-morbidity in defining major depression subtypes associated with long-term course and severity," *Psychological Medicine*, vol. 44, no. 15, p. 3289–3302, 2014.

[9] N. Zhang, C. Liu, Z. Chen, L. An, D. Ren, F. Yuan, R. Yuan, L. Ji, Y. Bi, Z. Guo, G. Ma, F. Xu, F. Yang, L. Zhu, G. Robert, Y. Xu, L. He, B. Bai, T. Yu, and G. He, "Prediction of adolescent subjective well-being: A machine learning approach," *General Psychiatry*, vol. 32, no. 5, 2019. [Online]. Available: https://gpsych.bmj.com/content/32/5/e100096 [Accessed: 2023-04-18]

[10] A. Madurkar, "Graph machine learning with python part 1: Basics, metrics, and algorithms," Towardsdatascience, Nov 2021. [Online]. Available: https://towardsdatascience.com/graph-machine-learning-with-python-pt-1-basics-metrics-and-algorithms-cc40972de113 [Accessed: 2023-04-17]

[11] H. Källmén and M. Hallgren, "Bullying at school and mental health problems among adolescents: a repeated cross-sectional study," *Child and Adolescent Psychiatry and Mental Health*, vol. 15, no. 1, p. 74, 2021.

[12] B. Johansson, "Associate professor in social work," April 2023, personal communication.

[13] F. Harary, *Graph Theory*. Reading, MA, USA: Addison-Wesley, 1969.

[14] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2666651021000012 [Accessed: 2023-04-10]

[15] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *International Conference on Learning Representations (ICLR)*, 2017.

[16] Y. Ioannidis, R. Ramakrishnan, and L. Winger, "Transitive closure algorithms based on graph traversal," *ACM Trans. Database Syst.*, vol. 18, no. 3, p. 512–576, sep 1993. [Online]. Available: https://doi.org/10.1145/155271.155273 [Accessed: 2023-04-12]

[17] Neo4j, "Graph database concepts." [Online]. Available: https://neo4j.com/docs/getting-started/current/appendix/graphdb-concepts/ [Accessed: 2023-04-10]

[18] ——, "Introduction to cypher." [Online]. Available: https://neo4j.com/docs/getting-started/current/cypher-intro/ [Accessed: 2023-04-10]

[19] E. D. Claudio Stamile, Aldo Marzullo, *Graph Machine Learning: Take graph data to the next level by applying machine learning techniques and algorithms*. Packt Publishing, 2021.

[20] W. Hu, M. Fey, H. Ren, M. Nakata, Y. Dong, and J. Leskovec, "Ogb-lsc: A large-scale challenge for machine learning on graphs," 2021.

[21] Z. Ghahramani, "Unsupervised learning," *Advanced Lectures on Machine Learning: ML Summer Schools 2003, Canberra, Australia, February 2-14, 2003, Tübingen, Germany, August 4-16, 2003, Revised Lectures*, pp. 72–112, 2004.

[22] Neo4j, "Modularity metric." [Online]. Available: https://neo4j.com/docs/graph-data-science/current/algorithms/alpha/modularity/ [Accessed: 2023-04-17]

[23] ——, "Conductance metric." [Online]. Available: https://neo4j.com/docs/graph-data-science/current/algorithms/alpha/conductance/ [Accessed: 2023-04-17]

[24] J. M. Santos and M. Embrechts, "On the use of the adjusted rand index as a metric for evaluating supervised classification," in *Artificial Neural Networks–ICANN 2009: 19th International Conference, Limassol, Cyprus, September 14-17, 2009, Proceedings, Part II 19*. Springer, 2009, pp. 175–184.

[25] A. Amelio and C. Pizzuti, "Is normalized mutual information a fair measure for comparing community detection methods?" in *Proceedings of the 2015 IEEE/ACM international conference on advances in social networks analysis and mining 2015*, 2015, pp. 1584–1585.

[26] W. Dong, C. Moses, and K. Li, "Efficient k-nearest neighbor graph construction for generic similarity measures," in *Proceedings of the 20th International Conference on World Wide Web*, ser. WWW '11. New York, NY, USA: Association for Computing Machinery, 2011, p. 577–586. [Online]. Available: https://doi.org/10.1145/1963405.1963487 [Accessed: 2023-05-09]

[27] Neo4j, "K-nearest neighbors." [Online]. Available: https://neo4j.com/docs/graph-data-science/current/algorithms/knn/ [Accessed: 2023-05-09]

[28] ——, "Node similarity." [Online]. Available: https://neo4j.com/docs/graph-data-science/current/algorithms/node-similarity/ [Accessed: 2023-05-17]

[29] S. Fortunato, "Community detection in graphs," *Physics reports*, vol. 486, no. 3-5, pp. 75–174, 2010.

[30] F. Nielsen, *Hierarchical Clustering*, 02 2016, pp. 195–211.

[31] H. Lu, M. Halappanavar, and A. Kalyanaraman, "Parallel heuristics for scalable community detection," *Parallel Computing*, vol. 47, pp. 19–37, 2015, graph analysis for scientific discovery. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0167819115000472 [Accessed: 2023-04-17]

[32] Neo4j, "Louvain." [Online]. Available: https://neo4j.com/docs/graph-data-science/current/algorithms/louvain/ [Accessed: 2023-04-17]

[33] U. N. Raghavan, R. Albert, and S. Kumara, "Near linear time algorithm to detect community structures in large-scale networks," *Phys. Rev. E*, vol. 76, p. 036106, Sep 2007. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevE.76.036106 [Accessed: 2023-04-17]

[34] Neo4j, "Label propagation." [Online]. Available: https://neo4j.com/docs/graph-data-science/current/algorithms/label-propagation/ [Accessed: 2023-04-17]

[35] ——, "Weakly connected components." [Online]. Available: https://neo4j.com/docs/graph-data-science/current/algorithms/wcc/ [Accessed: 2023-05-18]

[36] H. Cai, V. W. Zheng, and K. C.-C. Chang, "A comprehensive survey of graph embedding: Problems, techniques, and applications," *IEEE Transactions on Knowledge and Data Engineering*, vol. 30, no. 9, pp. 1616–1637, 2018.

[37] H. Chen, S. F. Sultan, Y. Tian, M. Chen, and S. Skiena, "Fast and accurate network embeddings via very sparse random projection," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, ser. CIKM '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 399–408. [Online]. Available: https://doi.org/10.1145/3357384.3357879 [Accessed: 2023-04-05]

[38] Neo4j, "Node embeddings." [Online]. Available: https://neo4j.com/docs/graph-data-science/current/machine-learning/node-embeddings/ [Accessed: 2023-04-05]

[39] ——, "Fast random projection." [Online]. Available: https://neo4j.com/docs/graph-data-science/current/machine-learning/node-embeddings/fastrp/ [Accessed: 2023-04-05]

[40] M. Loodberg, N. Abera, Y. Christiansson, M. Loftsson, F. Warg, and B. Johansson, *Friendsrapporten 2022*, 2022nd ed., ser. Friendsrapporten, 2022.

[41] M. Zhang and Y. Chen, "Link prediction based on graph neural networks," *Advances in neural information processing systems*, vol. 31, 2018.

[42] Neo4j, N. Dewhurst, N. Ivakina, and D. Oliver, "Transition from relational to graph database - getting started," Mar 2023. [Online]. Available: https://neo4j.com/docs/getting-started/current/appendix/graphdb-concepts/graphdb-vs-rdbms/ [Accessed: 2023-04-19]

[43] E. Flygare, G.-M. Frånberg, P. Gill, B. Johansson, O. Lindberg, C. Osbeck, and Å. Söderström, *Evaluation of anti-bullying methods*. Skolverket, 2011. [Online]. Available: http://urn.kb.se/resolve?urn=urn:nbn:se:oru:diva-24377 [Accessed: 2023-04-19]

[44] J. Brock, M. Lange, J. A. Tratalos, S. J. More, D. A. Graham, M. Guelbenzu-Gonzalo, and H.-H. Thulke, "Combining expert knowledge and machine-learning to classify herd types in livestock systems," *Scientific Reports*, vol. 11, no. 1, p. 2989, 2021. [Online]. Available: https://doi.org/10.1038/s41598-021-82373-3 [Accessed: 2023-04-20]

[45] P. Z. Lappas and A. N. Yannacopoulos, "A machine learning approach combining expert knowledge with genetic algorithms in feature selection for credit risk assessment," *Applied Soft Computing*, vol. 107, p. 107391, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1568494621003148 [Accessed: 2023-04-20]

[46] Åsa Gustafsson, "Expert at friends," April 2023, personal communication.

[47] K. P. Allen, "Classroom management, bullying, and teacher practices," *Professional Educator*, vol. 34, no. 1, p. 15, 2010.

[48] D. Olweus, *Mobbning i skolan : vad vi vet och vad vi kan göra*, 2nd ed. Stockholm: Almqvist & Wiksell, 1992.

[49] C. Eid and O. Enström, "Bachelor_thesis_oe222fh_ce223af," https://github.com/oenstrom/Bachelor_thesis_OE222FH_CE223AF, 2023.