Master thesis

Master’s Programme in Embedded and Intelligent Systems, 120 credits

Mining infrequent group of motifs from multidimensional time series: A case study at Alfa Laval AB

Computer Science and Engineering, 30 credits

Halmstad, 21-01-2023
Fabeena Thoufeeq and Reeba Susan Ashok
Fabeena Thoufeeq and Reeba Susan Ashok: *Mining infrequent group of motifs from multidimensional time series: A case study at Alfa Laval AB*, © Jan 2024

**SUPERVISOR:**
Hadi Fanaee Tork

**LOCATION:**
Halmstad, Sweden
In collaboration with an industrial partner, Alfa Laval AB, this thesis discusses a novel approach for identifying operational modes, specifically a cleaning mode, in separator machines without the benefit of labelled data and with very limited operating knowledge. Understanding the operational modes is crucial for comprehending the machine’s behaviour and ensuring its optimal performance.

Alfa Laval AB relies on a threshold-based fault detection system. The cleaning mode triggers vibrations that confuse the machine’s fault detection system, resulting in false alarms. The primary challenge revolves around the limited understanding of this infrequent cleaning mode, occurring periodically for 1-2 hours at intermittent intervals. To tackle this, we approach the problem as a data mining task. Matrix Profile (MP), a powerful tool in time series data analysis excels at identifying motifs and discords but struggles to distinguish between frequent and non-frequent motifs.

To address the drawback, we introduced an innovative approach capable of detecting frequent motifs and non-frequent motifs from the matrix profile output. The fundamental concept involves extracting the top-K motif matches using the Matrix Profile (MP) and systematically monitoring the evolution of structural similarity through pairwise similarity matrix calculation, progressing from pairs of two motifs to a group of K motifs.

This approach helps us to identify infrequent motifs that contain the most similar patterns which will be a good fit to address our challenge of identifying the cleaning mode.
ACKNOWLEDGEMENTS

We want to thank our supervisor, Hadi Fanaee Tork, for guiding us and offering valuable feedback throughout our work. His patience and dedication truly helped improve our project. We’re grateful for the time and insights he shared.

We also extend our thanks to Jens Lundstrom, our examiner, and Yuantao Fan, our opponent, for their constructive feedback and suggestions. A special mention to Slawomir Nowaczyk for his input during our presentations.

Lastly, we’re thankful to Halmstad University for providing this enriching course and learning opportunity.

Fabeena Thoufeeq & Reeba Susan Ashok
**CONTENTS**

1 INTRODUCTION 1  
1.1 Background 1  
1.2 Problem Formulation and Contribution 2  
  1.2.1 Objective 2  
  1.2.2 The Main Problem 2  
  1.2.3 Our Approach using Matrix Profile (MP) 3  
  1.2.4 Challenges Faced with Matrix Profile (MP) 4  
  1.2.5 Our Novel Contribution to Overcome the Challenge 5  
  1.2.6 How this methodology help Alfa Laval AB in the future? 7  
  1.2.7 Limitations 8  
1.3 Research Questions 9  
1.4 Novelty 9  

2 LITERATURE SURVEY 11  

3 DEFINITIONS, WORKING PRINCIPLE AND ALGORITHMS 15  
  3.1 Matrix Profile 15  
  3.2 Definitions and Notations 15  
  3.3 Working Principle of Matrix Profile 16  
  3.4 Mueens ultra-fast Algorithm for Similarity Search (MASS) 18  
  3.5 The STAMP Algorithm 18  
  3.6 STUMPY Python Library 19  

4 METHODOLOGY 21  
  4.1 Approach 22  
    4.1.1 Dataset Analysis: 22  
    4.1.2 Dimensionality Reduction using Principal Component Analysis 25  
    4.1.3 Window Size Selection: 26  
    4.1.4 Best Match Identification: 27  
    4.1.5 Identification of Frequent and Infrequent Patterns: 28  

5 RESULTS 31  
  5.1 Analysis of Pairwise similarity drift 31  
  5.2 Plot Analysis - No best matches 34  
  5.3 Plot Analysis - With best matches (Pairwise similarity drift analysis) 37  
    5.3.1 250628 37  
    5.3.2 250631 40  
    5.3.3 250625 43  
  5.4 Evaluation 45  
    5.4.1 Frequency Distribution 45  
    5.4.2 Client Confirmation 47
CONTENTS

6 DISCUSSION 49
   6.1 Answers to Research Questions 49
7 CONCLUSION 51
   7.1 Future Works 51

BIBLIOGRAPHY 53
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 1</td>
<td>Motif and discord in a time series data</td>
<td>4</td>
</tr>
<tr>
<td>Figure 2</td>
<td>Limitations of Matrix Profile (MP) and our novel method to overcome it</td>
<td>6</td>
</tr>
<tr>
<td>Figure 3</td>
<td>Proposed novel method</td>
<td>22</td>
</tr>
<tr>
<td>Figure 4</td>
<td>Multidimensional motif - Machine: 250631</td>
<td>24</td>
</tr>
<tr>
<td>Figure 5</td>
<td>Multidimensional motif - Machine: 250628</td>
<td>24</td>
</tr>
<tr>
<td>Figure 6</td>
<td>Principal Component Analysis - Machine: 250631</td>
<td>26</td>
</tr>
<tr>
<td>Figure 7</td>
<td>Principal Component Analysis - Machine: 250628</td>
<td>26</td>
</tr>
<tr>
<td>Figure 8</td>
<td>Plot Analysis</td>
<td>29</td>
</tr>
<tr>
<td>Figure 9</td>
<td>Drift in pairwise similarity: Best matches for the pattern with occurrence count of 12, 7, 4, and 0</td>
<td>33</td>
</tr>
<tr>
<td>Figure 10</td>
<td>Plot analysis without any best matches</td>
<td>34</td>
</tr>
<tr>
<td>Figure 11</td>
<td>Non-identical members</td>
<td>36</td>
</tr>
<tr>
<td>Figure 12</td>
<td>Pairwise similarity drift analysis - Machine: 250628</td>
<td>37</td>
</tr>
<tr>
<td>Figure 13</td>
<td>4 similar and remaining non-similar matches</td>
<td>39</td>
</tr>
<tr>
<td>Figure 14</td>
<td>Pairwise similarity drift analysis - Machine: 250631</td>
<td>40</td>
</tr>
<tr>
<td>Figure 15</td>
<td>4 similar and remaining non-similar matches</td>
<td>42</td>
</tr>
<tr>
<td>Figure 16</td>
<td>Pairwise similarity drift analysis - Machine: 250625</td>
<td>43</td>
</tr>
<tr>
<td>Figure 17</td>
<td>3 similar and remaining non-similar matches</td>
<td>45</td>
</tr>
<tr>
<td>Figure 18</td>
<td>Frequency distribution of the machines</td>
<td>47</td>
</tr>
</tbody>
</table>
LIST OF TABLES

Table 1    Toy time series   16
Table 2    Reference Subsequence  16
Table 3    Subsequence 1     17
Table 4    Subsequence 2     17
Table 5    Subsequence 3     17
Table 6    Subsequence 4     17
Table 7    Subsequence 5     17
Table 8    Distance Profile  17
Table 9    Matrix Profile    18
ACRONYMS

MP  Matrix Profile
API  Application Programming Interface
PCA  Principal Component Analysis
SAX  Symbolic Aggregate Approximation
DTW  Dynamic Time Warping
SOM  Self Organizing Map
FFT  Fast Fourier Transform
MASS Mueen’s ultra-fast Algorithm for Similarity Search
STOMP Scalable Time Series Outlier Mining Algorithm
STAMP Scalable Time series Anytime Matrix Profile
INTRODUCTION

1.1 BACKGROUND

Within industries [22], it is incredibly important to use time series data to observe how machines work over time. This allows us to predict when machines might have problems and fix them before they cause major issues. By studying how machines normally behave, we can quickly identify unusual patterns or variations that something might be going wrong. It’s as if we have a special warning system that alerts us to potential machine problems. This ensures that everything runs smoothly and that we can fix any faults before they become serious.

Alfa Laval AB has invested in robust fault detection systems, but occasionally these systems trigger false alarms due to our incomplete knowledge of the different operating modes of the machine. It is crucial to recognize and understand all the different modes of operation of these machines. By mastering operating modes, the company can refine the detection systems and ensure that alarms are only triggered when a real problem occurs. In this way, they can maintain smooth operations, minimize interruptions, and optimize response to real machine malfunctions [23].

Time series data, which track observations over time, can be divided into univariate and multivariate (multi-dimensional) data based on the number of variables recorded. Univariate data refer to a single variable over time, while multivariate data record several variables simultaneously. Analyzing multivariate time series data poses more challenges than analyzing single time series. The goal of time series analysis is to understand past behaviours, monitor current trends, and predict future values [5]. This analysis is essential for predictions, trend detection, and the detection of anomalies in the data [13].

The detection of motifs and discords is an important task in this area: motifs are recurring similar patterns, while discords are anomalies that do not repeat and stand out as outliers. The development of efficient algorithms for motif detection helps to summarize and visualize large data sets [19]. The detection of discords is also crucial as they signal serious problems such as system errors, defects, or suspicious activities that require immediate attention.
There are various ways to find patterns in time series data, e.g. dynamic time warping or Fourier transforms. However, we are enthusiastic about the Matrix Profile (MP) method because it can detect these motifs quickly and accurately. Other methods sometimes struggle with large data sets or take a long time to figure things out, but the Matrix Profile (MP) method does it all very well. It helps us find important patterns in the way machines work faster and more accurately over time, which gives us a better understanding of their behaviour for our research.

In the next sections, we will dive into the research problem of Alfa Laval AB separator machines. A major challenge for us is the precise identification of an operating mode that indicates the cleaning mode of the separator machines which is an infrequent mode of operation. We believe that the Matrix Profile (MP) method can be a valuable aid in this area. Our goal is to improve the ability of the Matrix Profile (MP) to recognize these infrequent motifs in the time series data of the separator machines. We will define it further in the following sections.

1.2 PROBLEM FORMULATION AND CONTRIBUTION

1.2.1 Objective

In collaboration with Alfa Laval AB, this thesis aims to solve a crucial problem: finding the infrequent mode “cleaning mode” in their separator machines. Alfa Laval AB has a large number of separator machines, but the operators have little knowledge about the behavior of the system. Our goal is to develop a novel method that uses the data from these machines to accurately detect this infrequent cleaning mode.

We want not only to identify the cleaning modes but also to develop a well-functioning method that can identify both the frequent and infrequent modes which will help Alfa Laval AB to better understand how the separator machines work.

1.2.2 The Main Problem

The biggest challenge is that there is no labeled data for these modes. Our knowledge of the cleaning mode is limited to its irregular occurrence, which motivates us to investigate the non-frequent patterns that are also evenly distributed in the time series data.
The challenge lies not only in the unavailability of existing solutions for identifying non-frequent motifs in multidimensional time series data but also in ensuring compatibility with the data from these separators. Another hurdle is to optimize the data to match these solutions. Once the motifs are identified, the next big challenge is to filter out the infrequent motifs from the results obtained.

1.2.3 **Our Approach using Matrix Profile (MP)**

Our approach is based on the use of the Matrix Profile (MP), a methodological innovation that has gained importance in time series analysis. The Matrix Profile (MP) represents a data structure that provides valuable insight into nearest neighbour information for all sub-series within a given time sequence [20]. This feature is invaluable in revealing various patterns such as motifs, discords, evolving patterns, and shapelets [21].

This methodology is based on the fundamental assumption that recurring patterns or motifs emerge over time in the workflows of separator machines. By carefully classifying these recurring patterns, we can systematically identify the modes of operation they are likely to indicate. The Matrix Profile (MP) described by authors Keogh and Mueen is an innovative and promising technique for motif recognition [20]. Its precision and simplicity distinguish it from conventional methods.

Our main goal is to develop an approach that efficiently identifies infrequent modes represented by similar patterns in the time series. This process relies on the detection of sparsely populated patterns that are evenly distributed across the time series. Matrix Profile’s ability to efficiently process large datasets, make optimal use of hardware, and overcome challenges solidifies its position as an advanced and comprehensive solution [22]. The combination of its recent emergence along with a diverse set of advantageous features, motivated us to explore and implement the Matrix Profile (MP) in our thesis for motif discovery.

To summarise, the use of Matrix Profile (MP) in motif discovery is a powerful tool that has proven its potential in our research to systematically identify motifs that can be further processed to understand infrequent modes of operation in separator machines.
1.2.4 Challenges Faced with Matrix Profile (MP)

While the Matrix Profile (MP) method excels in pinpointing the most accurate matches within a data set, its effectiveness in identifying the best matches of patterns is not been demonstrated. Existing libraries that are based on Matrix Profile (MP) computation enable the identification of the best matches/neighbors to a specified pattern, but they return a default set of motif neighbors and not the best matches. This requires manual efforts to segregate and identify best matches from the output.

The Matrix Profile (MP) based Stumpy library provides two primary APIs for finding similar patterns within multidimensional time series. One approach to finding these motifs is to use ‘stumpy.match,’ while another is to use ‘stumpy.mmotif,’ which is specifically designed to detect and cluster these motifs. Additionally, the ‘stumpy.match’ method can be used to retrieve all matches through the sliding window technique.

These types of APIs, however, return a default count of matches, rather than a set of the most optimal matches alone. Consequently, they do not autonomously determine the best matches naturally as the top k motif matches identified by these library functions may not be inherently similar to each other after a point.

As a result of this limitation, we have developed a novel method to provide a more thorough analysis of the motifs that are obtained through the Matrix Profile (MP). The overview of this entire process is explained in the next section. Please refer to Figure 2 for a detailed analysis.
graphical representation and explanation.

1.2.5 *Our Novel Contribution to Overcome the Challenge*

Time series data often contains patterns with varying degrees of complexity and structure. The primary contribution of our research is to develop a method capable of identifying the natural cluster of infrequent motifs from top-K motifs generated by the Matrix Profile (MP). By natural cluster, we mean that the number of infrequent motifs is a part of the question and should be automatically determined by our solution. So, our method does not need the number of clusters as user input.

Our research introduces a novel approach to the identification of infrequent motifs. In our proposed method, we begin by identifying the top matches to the given input pattern using an existing ‘match’ function from the Stumpy library. Subsequently, we employ a pairwise similarity measurement within the found motifs.

We start by taking top-k motif neighbors (starting from k to K) and compute pairwise similarity. Finally, we look for the first drift in the pairwise similarity matrix. For instance, if observed drift at k=5, that means the number of most similar matches should be 5. The rationale behind this approach is that if we add a motif whose pattern is different it increases the norm of the similarity matrix. Because, if it was similar to previous patterns similarity matrix should not experience a shock or a sudden drift.

Simple distance measures to the seed pattern might not capture the subtle differences in shape, duration, or other characteristics among different patterns. Pairwise similarity considers the relationships between individual patterns, providing a more detailed view of their variability. While acknowledging the increased computational burden, the benefits of using pairwise similarity in terms of capturing temporal relationships and identifying inherent groups outweigh the costs. This trade-off allows a deeper, more insightful analysis of the time series data, making it a justified choice in scenarios where detailed knowledge of patterns is required. Since the number of cleaning modes is unknown and has to be determined by the proposed solution, we track the pairwise similarity between top-k motifs extracted from matrix profile. We use the drift in pairwise similarity as an indicator of a heterogeneous motif that does not belong to the cluster.
Figure 2: Limitations of Matrix Profile (MP) and our novel method to overcome it

[January 21, 2024 at 19:56 – classicthesis version 4.0]
Our goal in Figure 2 is to illustrate the challenges encountered by Matrix Profile (MP) functionalities as well as the innovative approach taken to overcome these limitations. It has been demonstrated that recognizing the matches of identified motifs, produces a mixture of both similar and dissimilar patterns, as we have mentioned earlier in this section.

In Figure 2, we can look at an example where we try to find the best matches of the pattern using ‘stumpy.match’. Since the result of the function returns a default count of neighbours/matches, the output contains the three optimal matches of the pattern and two non-matches of the pattern. The expected output should only contain the three patterns that are similar to the input pattern.

As a solution to this problem, we propose a novel method to verify the pairwise similarity between the obtained patterns. Through the similarity matrix calculation, we can identify the change point where the structural similarity measure of the pattern experiences a sudden increase. As a result, our method autonomously segregates the patterns helping us identify infrequent patterns easily.

1.2.6  How this methodology help Alfa Laval AB in the future?

The proposed novel method, which utilizes Matrix Profile (MP) features, offers Alfa Laval AB significant advantages in improving operational efficiency and fault detection mechanisms in its separator machines.

Improved Operational Awareness: By accurately identifying infrequent operating modes, such as the cleaning mode in separator machines, the methodology provides comprehensive insights into machine behaviour. This understanding enables Alfa Laval AB to gain a holistic view of machine operation, including duration, frequency, and dynamics of modes.

Reduced False Alarms: The identification and labelling of different operating modes contribute to a more robust fault detection system. Currently, Alfa Laval AB uses a threshold-based fault detection system that triggers alarms when measured values exceed threshold values. However, certain operating modes, such as cleaning mode, may have readings above these thresholds, resulting in false alarms. If the fault detection system is aware of the different operating modes, it can distinguish normal readings from actual anomalies and thus significantly reduce the false alarm rate.
Improved Monitoring Mechanism: By understanding the different operating modes, Alfa Laval AB can monitor and analyze machine performance more effectively. It facilitates the evaluation of mode-specific performance metrics, helping to optimize machine operation and ensure safe and efficient operation.

Future extensions: While the method initially targets the cleaning mode, its flexibility allows for future extensions to identify other modes of operation. Extending this approach to recognize other modes of operation further improves operational awareness and fault detection capabilities.

1.2.7 Limitations

Although the proposed methodology using the Matrix Profile (MP) technique offers considerable advantages, it also comes up against certain limitations and considerations:

Dependence on Human Expertise: Successful implementation of this methodology depends on human expertise, especially in determining the optimal window size for time series analysis. In addition, confirming the analyzed motifs to gain insights into the modes of operation requires expertise, which emphasizes the collaboration between computational methods and expertise.

Complexity in Mode Analysis: While the methodology is effective in pattern recognition, it can be complex in interpreting and deriving actionable insights from the identified patterns. Obtaining meaningful information on the individual operating modes represented by the motif groups requires careful analysis.

Scalability: Ensuring the expandability and transferability of this method across different separator machines or industrial facilities remains an area for consideration. The applicability and performance of the approach in different operating environments need to be further investigated.

Evaluation: The absence of ground truth can lead to a subjective interpretation of results. It becomes challenging to quantitatively evaluate and validate the discovered motifs. Although synthetic data is a potential solution to such scenarios, there are limitations to it. It doesn’t perfectly capture all the complexities of the actual system. In our case, we do not have any idea about the realistic pattern of the cleaning mode. Therefore, it may not be possible to draw conclusions based solely on synthetic data.
1.3 RESEARCH QUESTIONS

1. Is mining infrequent motifs equivalent to the detection of cleaning mode in separator machines?

2. How to mine infrequent motifs from multidimensional time series?

1.4 NOVELTY

The efficient operation of separator machines relies heavily on accurately identifying operational modes. However, these machines currently lack a robust mode identification system, which leads to problems such as false alarms. In our research, we propose an innovative method that combines Matrix Profile (MP) with multidimensional time series data and utilizes structural similarity computation through pairwise similarity matrices to identify infrequent modes of operation like cleaning mode.

Despite Matrix Profile’s precision in finding exact matches within time series data, its ability to reveal inherent groups of frequent and infrequent patterns remains an unexplored area. Existing Matrix Profile (MP) libraries provide exact matches but require manual effort to separate and recognize the best matches, limiting their autonomous detection of optimal patterns.

Our primary goal is to develop a methodology to intuitively classify frequent and infrequent modes in time series data. In particular, we focus on the identification of an infrequent and intermittent pattern associated with cleaning modes in separator machines.
Research on time series data mining has been gaining immense popularity due to its widespread application in various fields due to its potential to extract useful information and trends. The various real-life applications of time series analysis include various economic forecasting, intrusion detection, gene expression analysis, medical surveillance and hydrology [9]. The research on time series is mainly carried out in the fields of representation and indexing, similarity measure, segmentation, visualization and mining [8].

Motif discovery is an essential part of time series analysis as it can give useful information on hidden trends and patterns. Motifs are similar and conserved patterns that repeat over the time series that appear in univariate or multivariate time signals. Sahar Torkamani et al. [5] discuss the various state-of-the-art algorithms that currently exist for motif discovery. The authors discuss methods like random projection algorithms based on the principle of Symbolic Aggregate Approximation (SAX). But according to Butler M et al [6]. SAX approach has its limitations because when Piecewise Aggregate Approximation is used, the standard deviation is altered due to alteration of the data distribution.

The methods using Dynamic Time Warping have very high time complexity [5]. Dynamic Time Warping is prominent where clustering, classification and similarity search are the main criteria as meaningful motifs can be discovered. The algorithm SWAMP is an improved version of existing motif discovery algorithms which is an alternative to expensive DTW calculations. SWAMP algorithm consists of two steps that detect the DTW motifs paring using a "brute force" search algorithm and then remove the subsequence time series through an adaptive hierarchy of lower bounds[10].

Maria Ines Silva et. al [11] propose DTW-SOM as a method to discover the motifs. This represents a method that is the combination of Dynamic Time Warping (DTW) and Self Organizing Map (SOM). Self-Organizing Map (SOM) algorithm helps to discover the centres of motifs returned by any motif discovery algorithm. SOM maps high dimensional data to low dimensional data preserving the relations between the motifs. This paper explains how the motifs discovered using DTW are used in visualizing the data in a two-dimensional space using SOM.
Although there are many methods as discussed above available, a new novel algorithm based on the Matrix profile is gaining immense popularity for being a powerful tool in time series analysis. As per the researchers who developed it, their method is claimed to use a very fast similarity search algorithm based on \( z - z \)-normalised similarity search [1]. The algorithm generates two outputs which are the matrix profile and matrix profile index which store the distance of each sub-sequence to its nearest neighbours as well as the location of the neighbours respectively. They use a very fast algorithm Mueen’s ultra-fast Algorithm for Similarity Search (MASS) to calculate the distance profile by exploiting FFT which stores the distance of each subsequence to all other subsequences and Scalable Time series Anytime Matrix Profile to calculate the matrix profile, which is the only algorithm that does full, exact joins on time series subsequences [1].

Even though the STAMP algorithm can process very large data sets, they have upgraded the STAMP algorithm to a much faster STOMP algorithm to process even bigger and huge data sets and it has the capacity to perform the same tasks that take around 20 years with the STAMP in just 12 days [2]. In contrast to the STAMP algorithm which does the search in a random order, STOMP performs an ordered search which reduces the processing time.

Matrix profile can also be used to efficiently discover meaningful multidimensional motifs. There have been many other approaches in multidimensional motifs discovery which involved dimensionality reduction or conversion to one-dimensional time series before analysis, analysing each of the dimensions separately. However, these methods can be very computationally expensive and less accurate. Alireza Vahdatpour et al.[7] proposes multidimensional motif discovery using a two-phase algorithm where the first phase involves single-dimensional motif discovery, and the second phase involves motif clustering but focuses on motifs that have time and value irregularities.

Multidimensional motif discovery using matrix profile is claimed to be very fast and accurate [3]. Similar to univariate time series, the method uses a multidimensional matrix profile that stores the distance values which is calculated using an algorithm developed specifically for multidimensional time series called MSTAMP. The algorithm requires two inputs which are the time series and the size of the sliding window.

Window size is a major feature in discovering motifs in time series. Based on the type of discovery required, whole-series-based or subsequence-based is chosen. This paper does a comparison study between the methods of window size algorithms such as Dominant Fourier Frequency, Highest Autocorrelation, Multi-Window Finder, Autoperiod,
Robustperiod etc and found that autoperiod and FFT are the better ones compared to others [12]

Eamonn Keogh et al. [4] experimentally show the efficiency of matrix profile in labelling events based on the similarities between subsequences. We can also see that the method can automatically label events with just one input parameter which is the the window size. As per the authors, one of the main advantages of using matrix profile is its ability to handle weakly labelled data. The matrix profile-based method can also deal with scenarios where it is not possible to have any training data thus proving to be an effective choice in unsupervised labelling mechanisms. This framework is also claimed to be fast, scalable and accurate.

Principal component analysis (PCA) [14] is a statistical procedure that transforms a large number of possibly correlated variables into a smaller number of uncorrelated variables called principal components (PCs). PCA is often used to reduce the dimensionality of data, but it can also be used to find patterns in data or to visualize data. To perform PCA, the data is first standardized so that all of the variables have the same mean and standard deviation. Then, the covariance matrix of the data is calculated. The eigenvectors and eigenvalues of the covariance matrix are then calculated. The PCs are the eigenvectors of the covariance matrix, and the corresponding eigenvalues are the variances of the PCs. The PCs are ordered from highest to lowest variance.

The first few PCs typically capture most of the variance in the data. Therefore, by only considering the first few PCs, we can often reduce the dimensionality of the data while preserving most of the information. PCA can also be used to find patterns in data. For example, if the PCs are plotted in two-dimensional space, the data points will tend to cluster around the origin. This can help us to identify different groups in the data.

PCA is a powerful tool that can be used for a variety of data analysis tasks. It is a relatively simple technique that can be implemented in a variety of software packages.

Liao’s [15] comprehensive survey explores time series clustering by first categorizing time series data into different types, such as data value type, sampling characteristics, dimensionality, and sequence length, to lay a foundation for understanding the diversity of time series data and its impact on clustering algorithms. The survey then addresses the challenge of defining similarity measures for time series data, highlighting the limitations of traditional metrics like Euclidean distance and introducing dynamic time warping (DTW) as a flexible solution for comparing time series patterns. Furthermore, the survey covers various clustering algorithms tailored for time series data,
which can be grouped into three categories: distance-based methods that use DTW or similar metrics for similarity assessment, model-based techniques that assume specific data models for clustering, and hybrid approaches that combine aspects of both distance-based and model-based methods, offering a nuanced approach to time series clustering.

The paper introduces a novel time series distance measure, MPdist, which is designed to overcome the limitations of existing measures like Euclidean distance and dynamic time warping (DTW) when dealing with time series data in data mining tasks [16]. MPdist is specifically engineered to be more robust and effective, addressing issues related to noise, outliers, and data irregularities that challenge Euclidean and DTW measures. The authors demonstrate through benchmark dataset evaluations that MPdist outperforms Euclidean and DTW in terms of accuracy and robustness while remaining computationally efficient. Additionally, MPdist offers advantages such as symmetry, making the comparison order of time series irrelevant, and improved interpretability compared to DTW, which allows for more straightforward subsequence matching. These qualities position MPdist as a valuable tool for various data mining tasks, including clustering, anomaly detection, classification, and similarity search in time series data, making it an essential asset for data scientists working with such data.
DEFINITIONS, WORKING PRINCIPLE AND ALGORITHMS

3.1 MATRIX PROFILE

The matrix profile, a newly developed algorithm, is considered far superior to the other existing methods when it comes to finding conserved repeated patterns as well as anomalies in time series. According to [1], the matrix profile algorithm presents us with a multitude of advantages as listed below:

- Simple and parameter-free
- Highly scalable and fast
- Incremental
- Parallelizable

3.2 DEFINITIONS AND NOTATIONS

Some of the definitions that are crucial in understanding the matrix profile are listed below as given in the paper [3].

**Time series motif**: A time series motif is the most similar subsequence pair of a time series

**Multidimensional time series**: A multidimensional time series is a set of co-evolving time series \( T \in \mathbb{R}^{d \times n} \) where \( d \) is the dimensionality of \( T \) and \( n \) is the length of \( T \)

**Multidimensional subsequence**: A multidimensional subsequence \( T_{i,m} \in \mathbb{R}^{d \times n} \) of a multidimensional time series \( T \) is a set of univariate subsequences from \( T \) of length \( m \) starting from position \( i \)

**Distance profile**: A distance profile \( D \in \mathbb{R}^{n-m+1} \) of a time series \( T \) and a subsequence \( T_{i,m} \) is a vector that stores \( \text{dist}(T_{i,m}, T_{j,m}) \) \( \forall j \in [1, 2, ..., n-m+1] \)

**Matrix profile**: A matrix profile \( P \in \mathbb{R}^{n-m+1} \) of a time series \( T \) is a meta time series that stores the z-normalized Euclidean distance between each subsequence and its nearest neighbour, where \( n \) is the length of \( T \), and \( m \) is the given subsequence length.
**k-Dimensional distance function**: The k-dimensional distance function computes the distance between two multidimensional subsequences by using only the best k out of d dimensions.

**k-dimensional matrix profile**: A k-dimensional matrix profile \( P \in \mathbb{R}^{n-m+1} \) of a multidimensional time series T is a meta time series that stores the z-normalized Euclidean distance between each subsequence and its nearest neighbour (the distance is computed using k-dimensional distance function), where n is the length of T, d is the dimensionality of T, k is the given number of dimension, and T is the given subsequence length.

### 3.3 Working Principle of Matrix Profile

The matrix profile stores the distance between a subsequence in a time series and its nearest neighbour. The only input parameters required for matrix profile calculation are the sliding window size and the time series. Its working can be better explained with an example. Consider a time series example given in Table 1. The sliding window size can be chosen as \( m = 2 \).

<table>
<thead>
<tr>
<th>5</th>
<th>7</th>
<th>10</th>
<th>2</th>
<th>12</th>
<th>3</th>
</tr>
</thead>
</table>

Table 1: Toy time series

The first step is calculating the pairwise Euclidean distance between each subsequence of size 2 with all other subsequences of the same size. We take one subsequence as the anchor and calculate the distance with all other subsequences by moving the sliding window of size 2.

The first subsequence is considered as the reference as shown in Table 2.

<table>
<thead>
<tr>
<th>5</th>
<th>7</th>
<th>10</th>
<th>2</th>
<th>12</th>
<th>3</th>
</tr>
</thead>
</table>

Table 2: Reference Subsequence

The Euclidean distance between the reference subsequence and all the other subsequences are calculated by moving the sliding window towards the right which can be seen in Table (3-7).
Table 3: Subsequence 1

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>10</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td></td>
<td>0.8</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4: Subsequence 2

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>10</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
<td>0.8</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 5: Subsequence 3

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>10</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.8</td>
<td>0.5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 6: Subsequence 4

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>10</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.8</td>
<td>0.5</td>
<td>0.9</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 7: Subsequence 5

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th>10</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.8</td>
<td>0.5</td>
<td>0.9</td>
<td>1.2</td>
<td></td>
</tr>
</tbody>
</table>

Table 8: Distance Profile

This process is repeated for all the subsequences, and they are placed on top of each other resulting in a distance matrix. Table 8 is an example of a distance matrix containing the distance values of five subsequences on top of one another.

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>0</td>
<td>0.8</td>
<td>0.5</td>
</tr>
<tr>
<td>S2</td>
<td>0.7</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>S3</td>
<td>0.3</td>
<td>0.9</td>
<td>0</td>
</tr>
<tr>
<td>S4</td>
<td>0.4</td>
<td>0.3</td>
<td>0.8</td>
</tr>
<tr>
<td>S5</td>
<td>0.6</td>
<td>0.7</td>
<td>0.9</td>
</tr>
</tbody>
</table>

The distance calculated between a subsequence to itself is irrelevant and ignored.
Now we take the smallest distance for each from subsequence from each row to find the nearest neighbour and store it as the matrix profile as seen in Table 9.

\[
\begin{array}{cccccc}
0.5 & 0.4 & 0.3 & 2 & 0.2 & 0.6 \\
\end{array}
\]

Table 9: Matrix Profile

3.4 **Mueens Ultra-Fast Algorithm for Similarity Search (MASS)**

The MASS algorithm is a new algorithm based on Euclidean distance and the algorithm is taken from [3].

Procedure MASS(Q, T)
Input: A query Q, and a user provided time series T
Output: A distance profile D of the query Q
1. QT ← SlidingDotProducts(Q, T)
2. \(\mu_Q, \sigma_Q, M_T, \Sigma_T \leftarrow \text{ComputeMeanStd}(Q, T)\)
3. D ← CalculateDistanceProfile(Q, T, QT, \(\mu_Q, \sigma_Q, M_T, \Sigma_T\))
4. return D

3.5 **The Stamp Algorithm**

The STAMP algorithm is used to calculate the matrix profile and the algorithm is taken from [3].

Procedure STAMP(TA, TB, m)
Input: Two user provided time series, TA and TB, interested subsequence length m
Output: A matrix profile PAB and associated matrix profile index
1. n_B ← Length(TB)
2. P_{AB} ← infs, I_{AB} ← zeros, idxes ← 1 : n_B−m+1
3. for each idx in idxes:
   4. D ← MASS(B[idx], T_A)
   5. P_{AB}, I_{AB} ← ElementWiseMin(P_{AB}, I_{AB}, D, idx)
4. endfor
6. return P_{AB}, I_{AB}

The MASS procedure serves to compute the distance profile, D, between a user-provided query, Q, and a given time series, T. Initially,
it calculates the sliding dot products between Q and T, resulting in QT. Subsequently, it computes statistical parameters such as the mean (\(\bar{Q}\)) and standard deviation (\(\sigma_Q\)) of Q, along with the mean (\(\bar{T}\)) and standard deviation (\(\sigma_T\)) of T. Utilizing these statistics and the sliding dot products, MASS derives the distance profile D, capturing the dissimilarity between Q and T. Ultimately, it outputs this distance profile, enabling users to assess the similarity between the query and the time series data.

The STAMP procedure aims to identify the matrix profile \(P_{AB}\) and its associated index \(I_{AB}\) for two distinct time series, \(T_A\) and \(T_B\), considering a specific subsequence length denoted as \(m\). Initially, it initializes \(P_{AB}\) as infinity, \(I_{AB}\) as zeros, and a sequence of indexes. Iterating through each index, STAMP calculates the MASS between subsequences of length \(m\) from \(T_B\) and the entire \(T_A\) using the MASS procedure. It then compares these computed distances to the existing matrix profile \(P_{AB}\), updating \(P_{AB}\) and \(I_{AB}\) by selecting element-wise minimum values. In conclusion, STAMP provides users with the matrix profile \(P_{AB}\) and its associated index matrix \(I_{AB}\), facilitating the identification of similar patterns and subsequences between the two-time series datasets.

### 3.6 STUMPY PYTHON LIBRARY

STUMPY is a very powerful library that has been developed for computing the matrix profile by carefully applying the concepts published in the paper \([1,2,3]\). In our thesis, we have used two functions from the library

**Match:** The `stumpy.match` function in the STUMPY library is a powerful tool for efficiently searching for a specific pattern of interest within a time series. By leveraging the capabilities of STUMPY, this function automates much of the intricate work involved in pattern matching. Users can input their target pattern, and `stumpy.match` efficiently scans through another time series to identify occurrences of the specified pattern.

**Matrix profile Dist (Mpdist):** The `stumpy.mpdist` simplifies the task of measuring similarity between subsequences in distinct time series. This function is particularly useful in time series analysis, enabling users to measure the similarity between subsequence patterns across different time series datasets.
METHODOLOGY

In our study, we focused on finding and grouping recurring patterns in the time series data of separator machines. We assume that the processes in these machines are repeated over time. By finding these recurring patterns, we can systematically analyze them to find out the modes of operation, especially the cleaning mode. This helps us to understand the different behaviors of the separator machines and gain insights into their operation. This method is at the core of our analysis and provides a clear and effective way to understand the time series data we are working with.

Our analysis centers on identifying groups of similar motifs within time series data that consist of a small number of members. In particular, we are interested in isolating non-frequent modes, such as a cleaning mode. In this context, larger groups within the time series can be interpreted as indicative of a frequent or normal operating mode of the machine. Essentially, the number of members in a group of motifs serves as a key measure for distinguishing between operating modes, with smaller numbers highlighting less frequent states, such as a specific cleaning mode, while larger numbers indicate more typical and recurring operating patterns in the machine.

For motif identification, we have used a recently developed method known as Matrix Profile (MP), which is claimed to be very fast, accurate, and requires much fewer parameters. While the Matrix Profile (MP) method excels at accurately identifying specific matches to a pattern, its effectiveness in categorizing the patterns based on frequency is yet to be proven. Existing libraries based on the calculation of matrix profiles allow the identification of the best matches/neighbors to a given pattern, but they provide a standard set of motif neighbors and not the most optimal matches.

Current methods that use Matrix Profile (MP) output a default set of motif neighbors, so manual work is required to sort and find the best matches out of them. This manual process limits their ability to independently generate the best group members that fit the data. Our novel method aims to automatically detect and isolate the most similar patterns within these matches, reducing the need for extensive manual work.
4.1 Approach

Figure 3: Proposed novel method

Our approach can be broken down into the following steps:

4.1.1 Dataset Analysis:

The data we received is from separator machines located in five different locations. Each location has several separator machines.
The dataset from all the separator machines are similar and it consists of seven parameters:

- SiteID
- DeviceID
- Timestamp
- Temperature
- X-RMS
- Y-RMS
- Z-RMS

The time series data in this scenario is multivariate/ multidimensional and we must consider four parameters to analyze the data. The time interval of the data collection (sampling rate) is 5 mins. Keeping the actual temporal order is crucial for accurate analysis because it holds the relationships and patterns within the data. Disregarding this order might lead to incorrect conclusions or a misreading of the data. The matrix profile methods in ‘stumpy’ preserve the temporal order of the original time series. We have evaluated the data of separator machines to perform data cleaning and since no null values have been traced, this process is omitted.

The parameters are Temperature, X-RMS, Y-RMS and Z-RMS are the multidimensional parameters and below is the time series representation separately and together for the location BLCAM with DeviceID 250628 and 250631.
Figure 4: Multidimensional motif - Machine: 250631

Figure 5: Multidimensional motif - Machine: 250628
4.1.2 Dimensionality Reduction using Principal Component Analysis

The time series data from the separator machines comprises of four dimensions where the vibrations recorded from the machine are correlated. There can be various approaches for handling multidimensional time series like treating data separately or utilizing dimensionality reduction. Treating data separately is computationally intensive, while dimensionality reduction proves to be more time-efficient. In the context of correlated data, various motif discovery methods excel when sensor values are independent, but their efficacy diminishes when dealing with highly correlated multidimensional data. Employing dimensionality reduction techniques enables efficient information capture with minimal data loss, ensuring the effectiveness of motif discovery algorithms even after the reduction in dimensions[24]. This backs our idea to mix dimensionality reduction with matrix profiles.

We chose Principal Component Analysis (PCA) because it analyzes multidimensional machine behavior including temperature and vibrations along x, y, and z axes within our dataset and generates interpretable principal components without significant loss of information. The data was first standardized to ensure that PCA accurately captures the variance without being influenced by varying scales. Subsequently, PCA was applied to the standardized data, resulting in a set of principal components that represent the most critical information in the data set. When examining the cumulative explained variance plot, it became evident that the first principal component alone accounted for 75 - 80 percent of the total data variance.

Even though the variance retention is satisfactory there is a remaining 20 percent variance that is not included in the PCA and there might be an infrequent mode that can be missing. Given that our emphasis is on detecting the cleaning mode, an infrequent mode, successfully identifying it would overcome the limitation of PCA in this scenario.
4.1.3 Window Size Selection:

A crucial step in our approach involves the selection of an appropriate window size, guided by domain knowledge. This window size serves as the basis for subsequent operations, allowing us to focus on specific sub-sequences of the time series. There are several ways to choose the right window size.

We first tried trial and error by choosing several window sizes to generate the motifs and their sub-sequences. We understood that a smaller window size is better for identifying the motifs when the sampling rate is small and a large window size is for motifs that
have longer time intervals. Since the data is collected every five minutes, we have considered several window sizes ranging from 10 to 40. Based on domain knowledge, we consider window size 12 to be the optimal value.

There are several other methods available to detect the window size like Dominant Fourier Frequency, Multi-Window-Finder, Summary Statistics Sub sequence etc [12].

4.1.3.1 Top Matching Sub sequences:

Our approach involves identifying and analyzing the best matching neighbors for each of the sub-sequences within the time series data. To initiate this process, we locate the top 50 best matches or neighbors for all sub-sequences of the selected window size using the "stumpy.match" function. We recognize that the top 50 matches might not all be genuine matches, but this represents the 50 best candidates the match function could find. The approach identifies potential matches within the data to facilitate further analysis and improvement.

4.1.4 Best Match Identification:

Now we need to identify the best matches to the pattern from within the top 50 matches found in the above step. For this, we calculate the pairwise similarity between the first two matches as the initial similarity score, followed by the addition of the third match and the calculation of the pairwise similarity among the three. This process continues as we add each subsequent member and calculate the corresponding similarity measures.

To find the best matches, we use a similarity threshold. This threshold helps us decide which matches are good enough to include in the results. The rationale behind this is that for the initial similar patterns, the similarity score remains relatively constant but when a non-similar match is added to the calculation, there is a sudden increase in the similarity measure that goes beyond the threshold value. The threshold, set above 0.1, is determined through thorough analysis and visual inspection of diverse patterns across different machines. This increase in similarity measure indicates the drift in the structural similarity when a non-similar member is added to the group. Only matches that are below the threshold are considered. Finally, we return all the matches that meet the criteria up to the threshold point.
4.1.5  Identification of Frequent and Infrequent Patterns:

The best matches for a sub-sequence found in the above step give us the total number of occurrences of that sub-sequence across the time series. If the total count of occurrences is a large number then it is a frequent motif and if the number of occurrences is small it is infrequent or rare motif. We repeat this process for all the sub-sequences to identify all the frequent and infrequent motifs across the time series. Our primary objective is to find infrequent modes that exhibit minimal repetition across the time series. This approach is guided by our pursuit to find infrequent patterns with characteristics likely to represent the cleaning mode which is the key focus of our research.

We have implemented a Python code in such a way that it picks those patterns that follow a consistently high similarity for the first few members and then the plot shows a noticeable drift, indicating a significant change exhibiting a decrease in the similarity.

For a better understanding, we can look at an example. Figure 8 is a plot among the many generated which is the pattern we are looking for and is particularly of interest to us. The plot reveals the evolving similarity when we incorporated each of the top-matched members of a subsequence generated by the 'stumpy.match' function from the matrix profile into our similarity measures method. The initial plot shows how the similarity measure evolves with the introduction of each member of the group of matches identified using stumpy.match. It is important to note that a lower similarity measure value implies a higher degree of similarity.

In the plot shown in Figure 8, we observe that for the initial four members, the average similarity remains relatively constant. However, when the fifth member is added to the group, there is a noticeable increase in the measure, signifying a decrease in similarity. This observation leads us to conclude that among the top 50 identified matches, the top four are the true matches, and the rest can be safely disregarded.

The members section in Figure 8 provides a visual representation of the members involved in the plot presented in the figure. A close examination of these members reveals a clear distinction: the first four members exhibit a notable degree of similarity among themselves, while the fifth and sixth members tend to be different and do not belong to the same group.
Figure 8: Plot Analysis
RESULTS

In our study, we focused on the identification of infrequent motifs occurring at regular intervals within time series data across various machines. Through the utilization of the novel methodology developed, we successfully detected similar patterns of the same window length in multiple machines. This methodology as explained involved several key steps that enabled us to uncover various patterns within the time series data.

As discussed in previous sections we find the top 50 matches or neighbors of each sub-sequences of our time series identified using the "stumpy.match" method from the matrix profile and when we applied our novel methodology using pairwise similarity drift, we were able to distinguish between the frequent and non-frequent motifs. Utilizing this novel methodology, we successfully detected similar patterns of the same window length in multiple machines. Thus, our analysis revealed that, within the top 50 matches, distinct patterns became evident.

5.1 ANALYSIS OF PAIRWISE SIMILARITY DRIFT

It is evident that the results obtained by analyzing the drift in the pairwise similarity process yield different outputs that represent the number of members which represents the the most identical patterns. It is important to note that a lower similarity measure value implies a higher degree of similarity.

Figure 9 displays a few of the examples of drift /deviation in the pairwise similarity plot that we were able to observe, with the x-axis representing the number of matches and the y-axis indicating the pairwise similarity values. The plots (a),(b), and (c) are the plots that we are interested in as they clearly show the set of best matches as explained in detail in the previous sections. These plots correspond to different motifs, each having occurrences of the best matches of that pattern of count 12, 7, and 4 respectively. However, plot (d) represents a pattern that does not have any close matches.

The initial point is not a user-defined parameter. The first number always represents the similarity between the top two motifs. In this context, a score of ‘0’ signifies a high degree of similarity, while a score of 0.2 suggests subtle and negligible differences in the patterns. We increased the number of motifs from two to three, observed the average
pairwise similarity among the three motifs, and repeated the process until we detected a change in the average similarity. The matches that appeared before the drift belong to the first core of clusters. These counts indicate the number of occurrences for the best matches of that pattern within the data set. Higher indices in the plots signify more common patterns found across the machines. We selected patterns that share the most similarities, specifically focusing on those with a total occurrence count below five.
Figure 9: Drift in pairwise similarity: Best matches for the pattern with occurrence count of 12, 7, 4, and 0.
5.2 Plot Analysis - No Best Matches

While analyzing the graphs, we observed different types of plots, few without any best matches for the pattern as shown in Figure 10. This can be classified as patterns with a low level of similarity.

Figure 10: Plot analysis without any best matches

After examining the plot in Figure 10 and the members shown in Figure 11, it’s obvious that the matches we found don’t fit into the category of best matches. The sub-figures within Figure 11, are clearly different from each other. Each pattern looks unique, so it’s hard to group them together.
5.2 PLOT ANALYSIS - NO BEST MATCHES

(a)

(b)

(c)
Figure 11: Non-identical members
5.3 PLOT ANALYSIS - WITH BEST MATCHES (PAIRWISE SIMILARITY DRIFT ANALYSIS)

Our main aim was to identify the infrequent patterns, so we paid closer attention to the value of drift lower than five which will help us to identify the best matches with fewer than five members. While studying patterns in different machines, we detected one particular pattern that rarely appears in one of the machines and was observed in other machines too. So it is obvious that it’s not just random noise in the time series data. Now, we can thoroughly examine this pattern across different machines in the below sections.

5.3.1 250628

A close examination of Figure 12 and Figure 13 reveals a clear distinction: In Figure 13, the first four members exhibit a notable degree of similarity among themselves, while the fifth and sixth members tend to be different and do not belong to the same group. This observation leads us to conclude that among the top 50 identified matches, the top four are the true matches, and the rest can be safely disregarded.

Figure 12: Pairwise similarity drift analysis - Machine: 250628
5.3 PLOT ANALYSIS - WITH BEST MATCHES (PAIRWISE SIMILARITY DRIFT ANALYSIS)

Figure 13: 4 similar and remaining non-similar matches
5.3.2 250631

Subsequently, we examined patterns on a different machine labeled 250631 and focused on the drift of less than five which focuses on fewer than five members of the best possible match. During this analysis, we noticed the presence of a pattern from machine 250628 within machine 250631 as well. To thoroughly understand this match sequence, we conducted a detailed study, illustrated in Figure 14 and Figure 15.

Figure 14: Pairwise similarity drift analysis - Machine: 250631
5.3 Plot Analysis - with Best Matches (Pairwise Similarity Drift Analysis)

(a) 

(b) 

(c) 

[January 21, 2024 at 19:36 – classicthesis version 4.0]
Figure 15: 4 similar and remaining non-similar matches
5.3.3 250625

We aimed to locate similar matches for the patterns we found in earlier machines. Employing stumpy.match, we successfully identified two similar patterns shown in Figure 16. However, in the subsequent two figures, we observed a shift to dissimilar patterns.

Figure 16: Pairwise similarity drift analysis - Machine: 250625
RESULTS

(a) Motif, T=5630

(b) Motif, T=17306

(c) Motif, T=26012
5.4 Evaluation

To assess the accuracy of this novel method, we employed the following approaches.

5.4.1 Frequency Distribution

Our primary focus in this study is to identify the less frequent modes which are not anomalies. The possibility of occurrence of the mode can be in intermittent intervals which cannot be so close to each other. So we used frequency distribution as a useful way to assess our find-

Figure 17: 3 similar and remaining non-similar matches
ings. Hence we have plotted the occurrence of the modes for illustration and better understanding.

Machine: 250631, Distribution: 4956, 17000, 29552, 38661

Machine: 250628, Distribution: 12286, 24796, 31780, 35680
Figure 18: Frequency distribution of the machines

5.4.2 Client Confirmation

We’ve documented our findings and shared the outcomes of our work with the client for evaluation. We’re currently awaiting their feedback to confirm the cleaning mode we have discovered.
DISCUSSION

6.1 ANSWERS TO RESEARCH QUESTIONS

1. Is mining infrequent motifs equivalent to the detection of cleaning mode in separator machines?

A crucial aspect of this research is detecting the ‘cleaning mode’ present within separator machines by identifying the infrequent motifs. Our understanding of the cleaning mode is rather limited. It is an infrequent operational mode of the separator machine that occurs periodically, and it typically lasts one to two hours.

Our study suggests that the integration of Matrix Profile into our methodology is promising to uncover both frequent and infrequent multidimensional motifs in the time series data of separators. Using this approach, we were able to successfully uncover elusive, non-frequent motifs and, in particular, identify the characteristic ‘cleaning mode’ within the separator machines.

2. How to mine infrequent motifs from multidimensional time series?

Our research focuses on utilizing the capabilities of the matrix profile method to identify recurring patterns within multidimensional time series data obtained from separator machines. This involves employing the matrix profile to detect initial patterns and subsequently examining the results, integrating supplementary methods to refine the precision of pattern detection.

The matrix profile, known for its efficiency and accuracy, becomes a key tool in our pursuit of discovering meaningful multidimensional motifs within the operational data. As part of our methodology, we begin by effectively modelling the multidimensional time series data using a principal component analysis (PCA) to reduce the dimensionality of multidimensional data. In order to maintain the integrity of the data while reducing complexity, it is necessary to perform this step.

Following dimensionality reduction, the matrix profile is applied to identify recurring patterns of motifs that signify specific operational behaviors of the separator machines. The matrix profile, particularly when coupled with innovative techniques, proves effective in extracting frequent and infrequent patterns from the time series data. Our
approach involves identifying top-matching sub-sequences and generating the best matches by utilizing a pairwise similarity and a plotting technique to identify the number of occurrences of these matches.

By employing the matrix profile in conjunction with our novel approach, we addressed the existing limitation of the matrix profile in identifying frequent and infrequent motifs from time series data and uncovering motifs that carry operational significance. These motifs provide a representation of repeated patterns, allowing a more focused analysis of the separator machine’s behavior. In our study, similar patterns are detected and categorized within time series data, assuming machines exhibit repetitive behaviors over time. This method enables a systematic analysis of these recurring patterns and the total count of occurrences, which enables us to distinguish different modes of operation in the time series data.

The total count of best matches identified size serves as a key indicator of how operations are performed. Smaller numbers represent non-frequent modes of operation, such as cleaning operations. A larger number is the indication, on the other hand, that the machine is in a frequent or normal mode of operation. The emphasis on identifying patterns with a small number of members allows us to mine infrequent motifs from multidimensional time series data.
CONCLUSION

In conclusion, our research has successfully introduced and implemented a method for identifying and analyzing infrequent motifs within time series data. The methodology, which combines dimensionality reduction through PCA, window size selection, motif discovery using Matrix Profile (MP), and the discovery of drift in pairwise similarity, has proven to be a valuable approach for gaining insights from multidimensional datasets.

By applying this approach to our separator machines dataset, we were able to uncover patterns that exhibited remarkable similarity until a certain point, after which a sudden decrease in similarity signified a shift to a different pattern. This methodology allowed us to extract valuable insights into the non-frequent cleaning mode.

Furthermore, our approach is not limited to a specific domain or dataset, making it versatile and applicable to a wide range of time series analysis tasks. It provides a valuable framework for researchers seeking to better understand and work with high dimensional time series data, with applications in areas such as anomaly detection, trend analysis, and pattern recognition.

7.1 FUTURE WORKS

In future studies, there’s a lot to explore using this method. We noticed various patterns with best matches when we analyzed drift in pairwise similarity, and these matches had total occurrence counts ranging from 5 to 30. Understanding these patterns with varying counts of best matches better could help us link certain patterns to specific machine operations, giving us insights into how the separator machines work.

We mainly focused on a window size of 12 (equal to 1 hour) in our calculations. But by trying out different window sizes, we believe we can uncover more frequent and infrequent patterns in the machine data. This exploration of various window sizes could give us a better understanding of how things change over time, helping us predict and detect unusual events in different situations. Exploring these ideas further could make our method more useful in different industries,
making it better at analyzing data for various purposes.

In summary, our research has showcased the effectiveness of our approach in identifying frequent and infrequent motifs within time series data. As a result, we were able to identify the operational mode, in particular, the cleaning mode of the separator machines. Further analysis of the drift detection at higher values observed may identify the frequent modes of the machines which may help the operators have a better understanding of the machine’s behavior. The novel method can be applied in various other domains to contribute to the broader data analysis field and facilitate labeling processes.


