www.dergipark.gov.tr
ISSN:2148-3736
El-Cezerî Fen ve Mühendislik Dergisi
Cilt: 9, No: 3, 2022 (1085-1100)

El-Cezerî Journal of Science and Engineering
Vol: 9, No: 3, 2022 (1085-1100)
DOI :10.31202/ecjse.1075528

ECJSE

Research Paper / Makale

A Survey on Analysis of Data Mining Algorithms for High Utility Itemsets

Aditya NELLUTLA[1a*], N. SRINIVASAN[2b]

[1]Research Scholar, Sathyabama Institute of Science and Technology, 600119, Chennai
[2]Professor, Computer Science and Engineering, Rajalakshmi Engineering College, Chennai
[*]adityaresearch3@gmail.com

Received/Geliş: 16.03.2022 **Accepted/Kabul:** 22.08.2022

Abstract: High-Utility-Itemset Mining (HUIM) is meant to detect extremely important trends by considering the purchasing quantity and product benefits of items. For static databases, most of the measurements are expected. In real-time applications, such as the market basket review, company decision making, and web administration organization results, large quantities of datasets are slowly evolving with new knowledge incorporated. The usual mining calculations cannot handle such complex databases and retrieve useful data. The essential task of data collection in a quantifiable sequence dataset is to determine entirely high utility sequences. The number of sequences found is always extremely high, though useful. This article studies the issue of the mining of repeated high utility sequences that meet item restrictions to identify patents that are more suited to the needs of a customer. Also, this article introduces high-value element set mining, examines modern algorithms, their extensions, implementations, and explores research opportunities.

Keywords: Pattern mining, itemsets, Apriori algorithm, mean utilization.

Yüksek Faydalı Öğe Kümeleri için Veri Madenciliği Algoritmalarının Analizi Üzerine Bir Araştırma

Öz: Yüksek Faydalı Öğe Seti Madenciliği (HUIM), ürünlerin satın alma miktarını ve ürün faydalarını göz önünde bulundurarak son derece önemli eğilimleri tespit etmeyi amaçlar. Statik veritabanları için ölçümlerin çoğu beklenir. Pazar sepeti incelemesi, şirket karar verme ve web yönetimi organizasyon sonuçları gibi gerçek zamanlı uygulamalarda, büyük miktarlardaki veri kümeleri, dahil edilen yeni bilgilerle yavaş yavaş gelişmektedir. Olağan madencilik hesaplamaları bu kadar karmaşık veri tabanlarını işleyemez ve faydalı verileri alamaz. Ölçülebilir bir dizi veri setinde veri toplamanın temel görevi, tamamen yüksek faydalı dizileri belirlemektir. Bulunan dizilerin sayısı yararlı olsa da her zaman son derece yüksektir. Bu makale, bir müşterinin ihtiyaçlarına daha uygun patentleri belirlemek için madde kısıtlamalarını karşılayan tekrarlanan yüksek faydalı dizi madenciliği konusunu incelemektedir. Ayrıca, bu makale yüksek değerli eleman seti madenciliğini tanıtır, modern algoritmaları, bunların uzantılarını, uygulamalarını inceler ve araştırma fırsatlarını araştırır.

Anahtar Kelimeler: Örüntü madenciliği, öğe kümeleri, apriori algoritması, ortalama kullanım

1. Introduction

Data have been compiled and deposited in libraries in recent decades. Sensing these data has been difficult because the analysis by the hand of high data volumes is susceptible to errors and is time-consuming. Data mining is an essential activity as a solution. It involves the semi-automatic analysis of data using algorithms. Data mining algorithms are commonly considered to construct data predictive models to forecast the future or to identify fascinating trends for data description or past understanding. It is also called pattern mining to identify fascinating patterns in data. The aim

How to cite this article
Nellutla A., Srinivasan N., "A Survey on Analysis of Data Mining Algorithms for High Utility Itemsets", El-Cezerî Journal of Science and Engineering, 2022, 9 (3); 1085-1100.

Bu makaleye atıf yapmak için
Nellutla A., Srinivasan N., "Yüksek Faydalı Öğe Kümeleri için Veri Madenciliği Algoritmalarının Analizi Üzerine Bir İnceleme", El-Cezerî Fen ve Mühendislik Dergisi, 2022, 9 (3); 1085-1100.
ORCID: [a]0000-0003-3001-2056; [b]0000-0002-1650-7450

is to identify sets of values that exist in the data together and satisfy certain user requirements as specified in [1].

The main issue in the mining of patterns is called common itemset mining. The input is a transaction database (records) and a variable called the minimum supportive threshold. The performance is the popular itemsets, which are the value sets that are at least shown in input database records. In a database of purchases made by a customer, for instance, periodic itemset mining can be added to expose data such that customers frequently bought the articles together. There are two types of element set mining algorithms: one-step algorithms and two-phase algorithms. The two-step algorithms produce high utility itemsets for candidates during the first phase and again search the database to calculate candidate items for utility in the second phase.

The one-phase algorithms produce candidates and in one process compute their usefulness. The merger of frequent and high-value itemsets from a Database Table is offered with various data structures and algorithms. These data models are lower proportion performance time, the number of items examined, and the memory usage during the scan. For regular itemset mining, many algorithms have been suggested. These algorithms traverse the search area to locate regular objects, and many data models have been developed to enhance the productivity of time and space. Data mining methods are used in many real-world implementations to remove important correlations from datasets to support decisive decision-making. The frequently defined mining and association rule mining are two fundamental tasks to reveal interesting relationships among items in transactional databases.

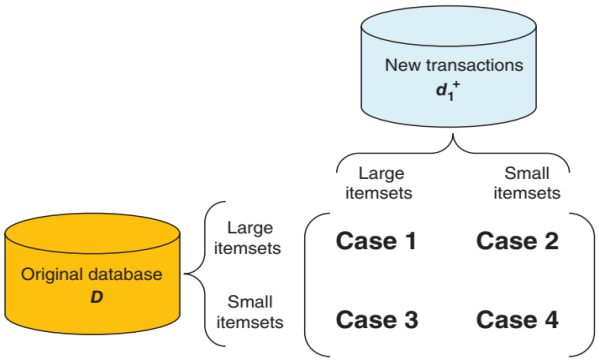

Figure 1. Overview of itemsets mining

Apriori algorithm is the most popular ARM algorithm. The method includes a generating and testing mechanism to identify frequent articles and the latter effectively degrades frequent articles with an approach of pattern growth. Several algorithms were proposed to mine association regulations effectively, usually based on levels of development or patterns of progress. Restriction pattern data mining techniques enable users to specify and then produce certain patterns, the requirements or restrictions that have to be fulfilled. Depending on these circumstances, the solution space can be decreased and obsolete or pointless patterns can be cut in the early stages to minimize the time it takes to locate patterns. Extensive scholars have conducted and categorized extensive studies on restrictive data mining algorithms discussed in [2].

The five main types of constraints, namely monotones, anti-monotone restrictions, clear and concise constraints, convertible anti-monotone restrictions, and convertible monotonous constraints, are being used in constraint-based data mining, according to [3]. The authors suggested that repeated pattern mining approaches dependent on constraints should be classified as item limitations, duration constraints, model-driven limitations, and aggregate limitations. The amount of data stored in databases grows progressively as a new business is inserted into real-world applications, such as pattern analysis in transactional databases and corporate decision-making. In batch mode,

traditional HUIM algorithms are run. Therefore, standard HUIM algorithms are used by the user to retrieve patterns from a modified database, but they do not know prior outcomes.

This is ineffectual since this knowledge should be found in a modified database to reduce the costs of finding trends. Traditional FIM and ARM methods for stage and trend development can manage only batch static databases. New trends may appear, and old patterns might become redundant if new transactions are added or modified in transactional data. Batch algorithms are modern database that is not suited for certain functional implementations with a modified database. In the 1990s, research on algorithms for pattern mining began with algorithms to find common shapes in records as specified in [4]. Apriori is the initial system aimed at recurrent model mining. It is intended to find common objects in client transaction databases.

A financial database is a collection of documents that show goods bought at various times by consumers. A common set of values is a collection of values commonly acquired by consumers in several transactions of a data structure. Human beings perceive those patterns clearly and could be used to promote decision-making. The pattern will, for example, be used for marketing decisions like spicy noodles. The identification of often used objects is a well-learned activity that applies in several fields. The analysis of a record to locate coinciding values in a series of records can be considered the overall job. While regular mining of patterns can be useful, it is supposed to be interesting frequent patterns. However, there are also applications of this assumption has been discussed in [5].

For instance, the pattern may be highly prevalent in a transaction data bank but may not be interesting, as it reflects a normal buying behavior, with low profitability. But on the other hand, some trends may not be common but may produce a greater benefit. Therefore, other factors such as benefit, or utility can be regarded to identify fascinating trends in statistics. A new field of study is the development of high utility patents in databases to overcome this restriction of frequent set mining. Utility research aims to extract patterns that are useful and of great significance to the consumer, in which the usefulness of a pattern is represented by a utility function.

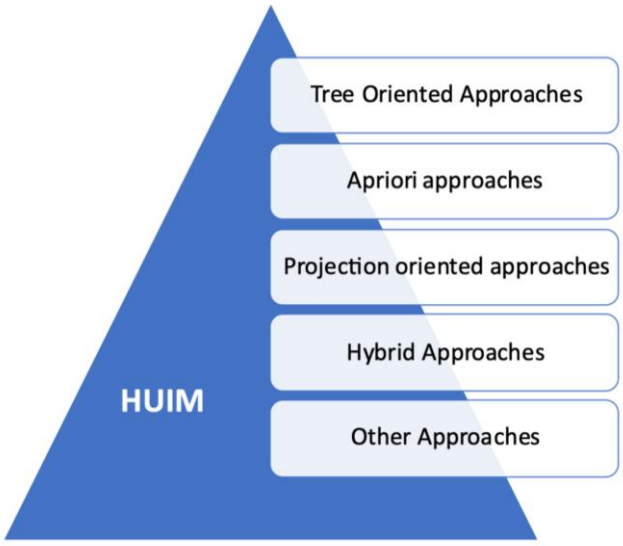

Figure 2. Classification of high utility Itemsets mining

An energy dissipation can be characterized by factors like benefits from the transaction or time spent on the network. Different forms of useful models were tested. This section examines the most common form of analysis, namely high utility papers. Mining high utility articles may be considered a general issue with regular mining of articles, where the entry is a database of transactions with a weight that reflects their value and where objects can be transacted in non-

binary amounts. This specific problem statement allows users to model different activities, such as the discovery of all items (sets of items) which make a high benefit from the transaction database, the search for web pages within a significant period, or the search for all common outlines, as in conventional regular pattern mining. The high efficacy approach can be a highly productive field of exploration discussed by [6].

This chapter offers an extensive examination of the topic, an introduction, and guidance on recent developments and prospects for study. Many methods in metaheuristics are in stochastic optimization, where outcomes are based on random variables. Metaheuristics may also discover efficient solutions with lower calculational work than optimization algorithms, iterative methods, or fundamental heuristics as they are searching for a wide range of potential solutions. Therefore, solutions to problems of optimization are useful. One of the major advantages of methods for optimizing metaheuristics is that strict limitations of termination can be implemented to restrict the calculation time such that an almost optimum solution can be found. Metaheuristic approaches to nature optimization are classified into evolutionary and smart computing groups.

Swarm Intelligence networks are typically a group of fundamental agents or entities that communicate with each other in their environment geographically. Nature, particularly biological processes, also inspires. Agents abide by very simple regulations and while there is a hierarchical control structure that decides how individual agents will behave, volatile and local interactions, to some extent, among these agents contribute to developing the "smart" global behavior that the agent does not understand. Natural development has inspired evolutionary computation. A container that mimics how the strategies look like and what the ingredients are is described in evolutionary computation.

The systems then create spontaneous, feasible but not inherently good solutions because, of course, they have been randomly assembled. These solutions won't be very successful, so we order them using some metric. Then delete the worse and keep the best a bit. We also build a few more candidate options for the next generation. Then mix and balance and borrow from strategies that in the last generations have been less evil. This method lasts for a while until a certain criterion is met and sometimes the time system will find almost optimum solutions.

2. Mean Utilization Measurement Algorithms

In PBAU, the authors suggested a tool to use to mine "high-average service products" based on prefix database prediction. To distinguish transactions from the database, an indexing system is used. The indexing process produces "High Average Services Products" from the database directly. That the original archive does not copy a lot of memory directly. To prune unsafe articles that result in less computational time, the cutting technique that overestimates the usefulness of any article is used by [7].

The variant of the PBAU solution is an optimized PAI algorithm. This method uses enhanced upper limits of my "High Average Utilities." The algorithm uses a projection technique to explicitly mines objects from the transaction database. Second, the nominee articles are not necessary, because the actual average utility articles are not mined. The entire initial database does not need to be copied due to the use of projection technology, which reduces memory use. The better upper bond reduces the calculation time significantly. The first article in the TPAU was the "Average Utility" metric as mentioned in [8].

The current test high average service items collection was suggested by Hong et al. to discover this article is a mining algorithm for "Two-Step." The "Downward Closure Property" algorithm is maintained in step one by searching out the upper limits of the element sets. This upper bound is

used to prune products whose utilities do not exceed the criterion of the "minimum utility." This method is carried out at a stage. For finding the actual "user" of itemsets a final database scan is needed in Phase 2. Since the benefits of the articles are overestimated, many candidate articles are cut out, saving a great amount of computing time [9].

The HAUI-Tree algorithm and a new data structure have been proposed for my "High Average Utility Itemsets." This optimization prunes itemset that overestimates the value of the item set without any promise. There is just one search of the site. Two datasets, namely BMS-POS and CHESS, are used to conduct experiments. This way candidates are produced even more quickly than in other approaches. The HAUI-Tree average minimum utility threshold takes 0.66min to use the BMSPOS data set at 0.8 percent as opposed to 157 minutes needed for PAIs. HAUI-Miner offers "High Average Utility Items Collections" to the "Utility List" structure to mine.

The "full utilities" in all transactions and the "average user interface" in each one of the element sets are needed for a database search. The database will again be scanned to delete High average objects, which have less than "Minimum Average Utility Terms." Database Each object in ascending order is changed to the index. All HAUUB databases are estimated and the item is pruned below the "Minimum Average Utility Threshold." In the context of each 1-HAUB item collection, HAUI-Miner uses the "Utility List" in the planned index. By using the depth-first search technique, the algorithm reveals "High Average Utility Products." Unpromising items are effectively pruned.

There are several algorithms built to extract patterns and rules frequently in immobile value databases for FIM and ARM. Since these value-based indexes are powerfully updated with new exchanges in genuine applications. The incremental fiber and ARM procedures developed, for example, FUP the pre-extensive concept, the FUFP-tree, and the PreFUFP-tree, for the incremental and intuitive mining in the territory of continuous mining example. However, these strategies are not always applicable to gradual, smart, useful mining examples. The typical HUIM estimates follow the model batch and do not suggest a response for gradual mining, which involves gradually inclusive exchanges.

Throughout 2008, the study examined IUM for the first time to consider entirely higher frequency utility objects occurring in an incremental database in a preset day and age. Both resource mining, known as IUM, and rapidly incremental utility mining two competent calculations are made (FIUM). In terms of the different calculations, the IUM calculation again depends on the FSM (ShFSM) calculation of the offers controlled [10]. However, because IUM is a tool that has been prioritized, multiple data sweeps may be used to isolate examples and witness the combinatorial explosions of the survey domain. FIUM is faster than IUM but has no distinguishable obstacles from IUM since it is a technique of Apriori aid.

The authors proposed the computation of the IHUP for the inclement and intuitive mining in HUIs with three tree systems, such as IHUPL-tree and IHUPTWU-tree for additional transactions. To gradually refresh the structure of HUIs using a FUP idea and show the 2-phase, the authors proposed calculating the HUI-INS calculation. It depends on the FUP concept, but the combinatorial blast in the investigating room is experienced. They suggested an upgraded calculation called pre-extended concept dependent measurement for extraction huis with an exchange extension to solve these impediments (PRE-HUI-INS).

An estimate of help to the inclusion of the exchange given the pre-substantial concept and TWU show. Late in the course of the interchange, an additional, memory-assisted approach was proposed to hold up to date and update collected HUIs. HUPIDGrowth calculation constructs a tree structure called a HUPID-tree with useful instances of a single database filter. Incremental databases were proposed for an approach called iCHUM with trees. The transaction data collection is packed into a

simpler decision tree known as the iCHUM-tree. They also designed the EIHI to discuss how incremental measurements are still outrageously high as far as the runtime is concerned and that the HUI-list-INS one-stage estimation still offers the potential to improve. A new list was developed to support the progressive approach, known as LIHUP, to draw up useful examples deprived of dynamic competition. It uses a list of the supported information structure for retention and creation of incremental information [11].

3. One-Phase and Two-Phase Algorithms

Two-phase algorithms are designed for computing high-end itemsets. The transaction-oriented weightage use model is described in stage 1 to estimate the top border on the usefulness of a group of items to satisfy the downward closing property of the transaction. Step 1 preparation is performed in a standard manner. Only high-speed operating articles together can be added to the candidate collection at each stage. While the overestimation of the transactional use of the itemset is an actual utility surplus, Two-Step scans the real high-level itemsets for the applicant set to discover in phase 2.

In [12], the researchers suggest a technique for isolated disqualification objects (IIDS) to enhance the HUIM methods at the stage. An item is referred to in the b^{th} pass as the isolated item if it does not appear in any high-profile candidate b-itemset or in some other candidate item set those extents are longer as of b. It can be used to decrease the number of applicants and to increase the efficiency of the HUIM procedures at the stage. The authors note that their prior conception would not take advantage of the existing HUIM algorithms. The effective structure of the tree incremental databases where IHUP construct until my many properties are proposed for mine high utility patterns.

In this way, the data structures and mining effects can be used to prevent unnecessary estimates. For the recently updated approaches, only the IHUP tree needs modification. The authors build UP-Tree built UP-Growth to discover high utility itemsets for multiple-pass scans in level algorithms that are efficiently produced over the Up-Tree with just dual permits over the search in the record. The UP tree is compact with the revamped transactions and an FP-Growth algorithm can be used to identify possible high utility objects (PHUIs) on the UP tree. Four methods (DGU, DGN, DLU, DLN), by discarding the utilities of promising products, are used to reduce the expected utility of candidates [13].

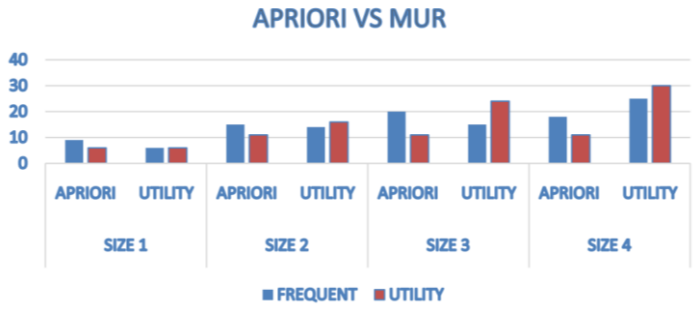

Figure 3. Comparison of algorithms

As the number of PHUIs frequently is considerably less than the number of candidates whose usefulness is estimated by transactions, step 2 of UP-Growth is much more efficacious than the evolutionary approaches. A two-phase algorithms design has been proposed by [14] to calculate high-value itemsets. The transaction-weighted use model is described in stage 1 to estimate the top border on the usefulness of a group of items to satisfy the downward closing property of the transaction. Step 1 preparation is performed in a standard manner. Only high-speed operating articles together can be added to the candidate collection at each stage. Since the weighted operation

of the object set is a surplus estimate of the real usefulness, Two-Step scans the true high-value itemsets in Phase 2 for the candidate set.

The authors are proposing an IDS to strengthen the HUIM approach on the ground. An item is referred to in the b^{th} pass as the isolated item if it does not appear in any high-profile candidate b-itemset or some other candidate itemsets are longer compared with b. It could be used to lessen the size of applicants and to increase the efficiency of the HUIM algorithms at the stage. The authors note their prior conception would not take advantage of the existing HUIM algorithms. The effective structure of the tree Incremental databases where IHUP "construct until my many properties' are proposed for mine high utility patterns has been discussed in [14].

In this way, the data structures and mining effects can be used to prevent unnecessary estimates. For the last updated transactions, only the IHUP tree needs modification. In [16], the authors develop an optimization to identify the high-use objects, be efficiently developed from the Up-Tree with only 2 passes in the database for multiple-passing scans in the level-specific algorithms. The UP tree is compact with the revamped transactions and an FP-Growth algorithm can be used to identify possible high utility objects (PHUIs) efficacies of unpromising objects to reduce the expected utility of candidates.

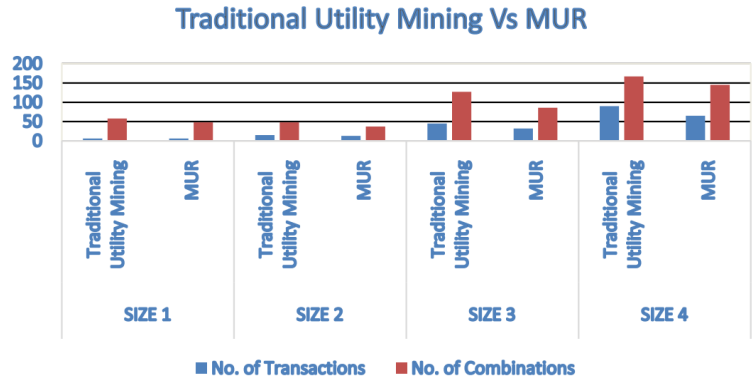

Figure 4. Utility mining analysis

As the quantity of PHUIs is always much lower than among candidates whose usefulness estimations are transaction weighted, Step is far much more effective than earlier procedures. Deprived of fixing the smallest efficacy brink, Wu, etc. proposes the TKU algorithm for the top-k HUI's. In Step 1, TKU initially builds UP-Tree in a transaction database by two scans to retain the transaction records. Moreover, during TKU, which has been fixed to 0 originally as well as step-by-schedule, a minimum service threshold is used during the generation of possible high-level products (PKHUIs). The quest process of UP-Growth will produce PKHUIs with a minimum utility threshold from the UP-Tree [15].

TKU calculates the exact PKHUI utility by another database search in step 2 and determines the current top-k HUI. In Step 1, the threshold is increased, and search space is pruned by four techniques as PE, NU, MD, and MC. In stage 2 of the Fifth Technique, SE, the utility threshold is increased and the number of applicants to be inspected is reduced. As TKU is to produce an enormous amount of candidate profiles, the researchers establish REPT HUIs with very low candidates for mining. The REPT's ultimate procedure is identical to TKU's. REPT initially builds a global tree with 2 database scans, creates a tree with top k candidate HUIs, and identifies candidates' results with another database scan [16].

To minimize applicant sizes substantially, the major improvements to REPT are to increase the minimum usefulness threshold efficiently. Two methods are used to raise the existing least efficacy

brink in the first global tree construction database search. Two more techniques, RSD and NU, are used in the routine checkup of the tree construction to further raise the threshold. In the building of a global tree, REPT precalculated the exact utility of 1 or 2 object sets that are used to lift the threshold. The future top-k HUIs could be produced using the UP-Growth technique from the global tree. In this step, the MC technique can be used to further raise the threshold. In Step 2, REPT will identify the top HUIs of contenders through additional SEP tactic, that recalls the approximate utilities fall.

Unfortunately, one-phase implementations do not produce applicants but measure the itemsets' utilities directly, unlike two-phased algorithms. This allows deciding the best k-HUIs in one step. Also, select the largest HUIs in one single step, the author proposes the algorithms. To store the utility information on itemsets, TKO uses the HUI-Miner search procedures and uses the dataset that is determined directly by its utility list when an object collection is created. Each object is initially correlated with a list of utilities, that is built over the skimming of the record twofold and reflects the item data in the transactions complicated.

A new approach to rapidly increase the minimum utility threshold process. Four techniques for improving TKO's efficiency are established by elevating the early value level and plummeting products' expected utility values. For high utility item collection discovery in a single step, Duong et al. offer a utility list-based kHMC algorithm. The search area is supplied with two new techniques, EUCPT and TEP. The EUCPT technique practices the element details accompaniment to exclude a significant quantity of accession processes. For minimizing the exploration domain, the TEP technique utilizes a new upper link to the utilities of item collection. In kHMC, an intersection is used to construct lowly complex efficacy lists. In addition, the early abandonment technique is built in the kHMC procedure, such that efficacy lists whose related products are not well-known HUIs are abandoned. Several methods are used in [17] to initialize and actively change the internal least efficacy brink to efficiently increase the inner least efficacy value.

The authors of [18] argue that for dense databases, especially at the beginning of the mining process, strategies to increase minimum utility limit values in modern algorithms are inefficient. A new THUI to efficiently mine the top-known HUI is proposed in one step. The THUI proposes four threshold increasing methods to increase the minimal utility threshold efficiently. The two last methods are the LIU structure newly implemented in THUI. The LIU structure is a triangular matrix given the specified commanding of items to maintain the utility details of items in a compact manner comprising contiguous items, which it claims to be more efficient in increasing the threshold values of the previous structures [14].

Another DynaDescend approach rearranges the elements in the decreasing direction including its higher limit so that the limit is rapidly elevated, and the pruning function is improved. More opportunistic methods are also suggested with the maximum power of TONUP, such as ExactBorder, SuffixTree, and OppoShift. For raising the frontier threshold, the ExactBorder approach customs the particular utility of the mentioned designs. The SuffixTree technique uses the tree to maintain the designs mentioned above, which are much more powerful than the prior information edifices. Unscrupulous approaches to dealing with the very long trends mentioned are opportunistically changed by the Oppo Shift technique [19].

4. High Utility Itemset Mining Algorithms

Many authors like [12], [13], [15] have suggested the issue of FIM to locate collections of objects (products) that appear in a database at least a few minimum times. The incidence intensity of a pattern is called support which has the cool property of being anti-monotonic. This is because a set cannot have a superset that has higher support. FIM has very broad search spaces. In general, there

are 2n - 1 itemsets available if the index contains n distinct items. N can be more than 1 million for certain applications such as business basket research on online stores. Associated with anti of the make reports, the search area is grouped with regular products, which eliminates most of the search space. This has resulted in many accurate algorithms being designed that are reasonably effective.

However, FIM's input data format remains quite plain despite its many uses. It can be interpreted as a record table of binary attributes. Therefore, data in many fields cannot be modeled well. In addition, common trends are not always fascinating and other parameters should be considered. The issue of HUIM was suggested in generalizing FIM for transaction databases in which the number of items and the weights of items reflects the unit profit of products is the basis of each transaction. HUIM's objective is to locate objects with a value greater than or equal to a minimum utility threshold. The problem with the HUIM is generally much more serious than with FIM since the utility function is neither anti-monotonic nor monotonous [20].

High utility itemsets should therefore be distributed in the search area and the utility cannot be used to decrease the search area directly. For HUIM, it is an active field of study, several exact algorithms had been suggested. To minimize search space effectively, various top limits on the use of anti-monotonic products, including the TWU upper bound, have been implemented by the above-mentioned accurate algorithms. These upper limits may therefore be very loose, and many low-utility articles are therefore also tested to identify genuine HUIs that impair results. Although the same FIM, as well as HUIM algorithms, ensure full results, run times can become very long. Especially if an algorithm will last for several hours or longer, especially when the user sets the minimum limit too short, it is common to interrupt the algorithm until it terminates. In addition, with certain transactions, lengthy transactions or many distinct objects the search area appears to get very broad. The development of evolutionary and heuristic algorithms was a fruitful solution to these problems. This same idea is to strike an outstanding balance between acceleration and integrity [21].

In truth, in a shorter time, an exact algorithm will find any of this algorithmics. In addition, evolutionary and heuristic algorithms usually develop the current solution iteratively and it is therefore simple to avoid them from getting results at all times. These algorithms can also be considered more realistic. FIM and ARM were the first to work on evolutionary, heuristic, and model mining algorithms. The experiments suggested, for example, the FIM and ARM GA. The two GAs suggested for HUIM were to iteratively identify the typical operators such as selection, intersection, and mutation. But the chromosomes 1-HTWUIs can initially not be readily found and so they require a large calculation to set the correct chromosomes for accurate mining of HUIs [22].

Furthermore, it is a non-trivial job to set the necessary values for certain particular parameters. HUIM-GA production was later improved with a cutting structure OR/NOR-tree. A bio-inspired structure for implementing GA for HUIM has been suggested. The HUI discovery process was expedited by effective database representation techniques and a pruning process. Furthermore, an improved GA that employed many new techniques to mine HUIs efficiently has been proposed. In addition to GA, PSO and BA were used in vast databases as a bio-inspired system to manage the HUI. In addition, the Proposed algorithm to resolve the HUIM problem was used in this framework [23].

Two PSO algorithms proposed recently are based on the traditional PSO and the second on a biologically influenced HUIM frame to resolve the issue of the high average useful element set mining (HAUIM). The thesis used the HUIM problem using a Boolean-based Gray Wolf algorithm known as BGWA-HUI. Furthermore, an OR/NOR-tree framework for productive mine HUIs has been adopted with a binary PSO. HUIs were also found using an Ant Colony System (ACS). The HUIM-ACS proposal mapped the whole space of the solution to the routing graph and used two

new pruning techniques to accelerate the convergence of the algorithm. FSM (Frequent Sequence Mining) is a well-studied data mining topic, which involves finding the set of FS of any sequence in a sequence database [24].

A series is also said to be common if there is no justification under a minimum user-defined support threshold for its incidence (support). For that dilemma, different data structures and research space exploration techniques have been planned with many powerful algorithms. These algorithms are based on the downward locking feature of the help function to reduce the search area. He says that it is not possible to endorse a sequence more than all of its subsequences. The PrefixSpan algorithm Pei et al. suggested, uses the trend development method and planned databases to decrease the costs of searching the database for repeated successions. It considers only trends in the database. However, the algorithm has to construct and archive several predicted databases, so runtime and memory use are inefficient [25].

Two other algorithms, SPADE and SPAM, which use a candidate strategy, have been designed. We use a hierarchical database format to easily compute sequence support without depending on the existing dataset. These algorithms produce several candidates however and it consumes a lot of effort to verify that the pattern of a candidate is common. The CM-SPADE, as well as CMSPAM algorithms, have been proposed to deal with this issue. They are essential features of SPADE and SPAM which use pairs of consecutive objects to reduce search space with coincidence information. Many of the above protocols use a conventional method that explicitly extracts sequences from a database [29].

Recently, a new way of generating frequent sequences from the frequent sequences of closed and generators has been suggested. Even if FSM is common, there is a significant drawback that every item is equally important and only binary amounts can be included in each input set. This assumption does not apply to applications like business basket analysis. For example, it is less profitable to sell bread than to sell diamonds and non-binary amounts of products such as five milk bottles can be purchased. The topic of FSM has been extended to high utility sequence mining to overcome that constraint. This latter issue is designed to manage sequences of internal and external utility values annotated in objects as mentioned [26].

The relatively important products and their proportions can be modeled respectively. Seen in unclear quantitative datasets, researchers have considered several HUSM extensions such as the discovery of top-kilometer HU sequences, HU sequences from evolved data sources, and gradual QSDBs, HU sequence laws, periodical HU sequences, and HU likelihood sequences. Most HUSM papers use the umax calculation to quantify sequence usefulness in a quantitative sequence for each input. These researches, therefore, take a positive view. Since umax does not comply with the downward shut-off properly, for example, if a utility sequence is not high, all its super sequences are also not high utility sequences, several researchers have suggested upper limits for umax, which value DCP to minimize search space. The first such UB is called SWU and is used in various algorithms. The SWU UB has been shown to meet the DCP property and can thus be used to decrease the search area. However, as the UB of the umax measure is not tight, algorithms that use SWU also test several trends of candidates. Tighter UBs have been suggested to more effectively minimize the search space as specified in [27].

5. Other HUIM Approaches

The standard high-value mining algorithms are limited by the assumption that all utility values are positive. Databases also have unfavorable utility values in real-life implementations. Take a quantity order database of consumer purchases in a department store, for example. In this database, products with negative unit income known as adverse external utilities are commonly included. The

explanation is that chosen pieces are mostly sold for attraction in retail stores. It has been shown that typical high-utility itemset mining algorithms search missing high-utility itemsets if negative benefit unit values exist in a database. It is because the higher limits such as the TWU no longer have upper limits for the use of objects when unfavorable use values are considered [28].

High-use itemsets will then be improperly pruned. Algorithms were suggested with novel upper limits to solve this issue. HUINIV-Mine is the first algorithm for mining high-value objects, which extends two-phase with negative utility values. Then it was proposed the FHN algorithms. It is an algorithm built on the one-step utility list that expands the FHM algorithm. More than two orders of size were found to be quicker than HUINIV-Mine [19]. High utility object range mining with Discount Methods is another extension to be more realizable to analyze consumer transactions.

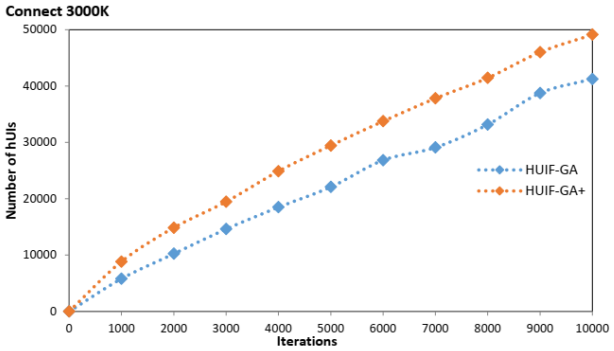

Figure 5. Training approaches comparison

This expansion is based on three kinds of discount policies that may be sold: an item can be soldered with a discount of 0 to 100%, if a customer buys n goods, receives m of free items and, when the customer buys *n* items of the same, he gets a percent discount on each item he buys [29].

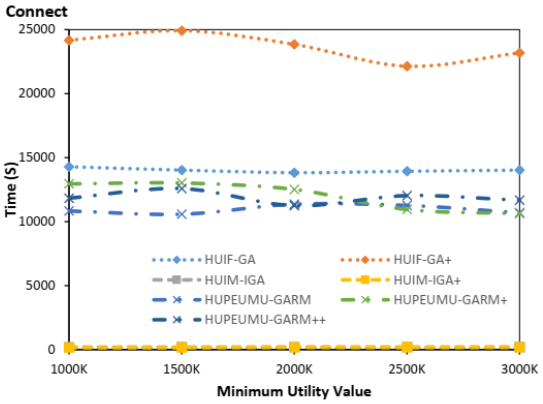

Figure 6. Runtime comparison analysis

An expanded statistical transaction repository is considered in a deflation strategies table that allows users to show if any for each object, the discount strategy. In addition, a table showing the expense and price tag for each commodity replaces the unit benefit table used for conventional high utility mining products. This helps the usefulness of each item to be calculated by taking the discount strategies into account. The first proposal was to mine high utility articles in a three-phase algorithm to understand discount strategy. It was subsequently proposed to extend two phases, Huiminer and FHM, respectively by three faster algorithms, called HUIDTP, HUIS-miner, and HUI-Deminer. Allows calculating the usefulness of each object collection by taking discount strategies into account [30].

A three-phase algorithm to mine high utility products was first suggested when discount strategies were considered. Therefore, three more fast algorithms were suggested, namely HUIDTP, HUI-DMiner, and HUI-DEMiner, which respectively extended the algorithms. Another extension of the useful itemset mining system is the finding of high usefulness objects that contain no more than a limited number of products maxLength defined by users. The reason for this expansion is that standard high-value mining algorithms will locate articles containing several articles. However, these objects are uncommon and could therefore be less interesting for consumers than smaller items [15].

Therefore, the full number of products that can be contained by high utility components is always desirable. One naive approach to doing this is first to explore all high utility articles using a standard high-utility article set mining algorithm. While this solution results in the right outcome, it is expensive since the length limit does not minimize search space. This approach is not effective. To increase the efficiency of the mining work, we should also drive the limitations to the deepest possible degree [21]. Length restrictions such as the overall length restriction were used during repeated pattern mining. The main concept for algorithms with a limit on maximum duration is to avoid an article collection containing the maximum number of objects because items are created by a remedial add-on to itemsets. While this technique will reduce the search space with longitudinal restrictions, there are new approaches to reduce search space with longitudinal restrictions, to further enhance algorithm accuracy has been explained in [25].

The FHM+ algorithm has been proposed by extending the FHM algorithm to tackle this problem. To eliminate upper limitations on the efficiency of the array using length restrictions and thereby reduce the search area, the new principle called "Length Upper-bound Reduction" (LUR) was proposed. The algorithm suggested could be much quicker than the FHM algorithm, and the number of patterns introduced to the user was significantly reduced [34]. Another constraint of conventional high-value itemset mining algorithms is that they often locate objects with high benefits yet poor correlations. These products are confusing or unnecessary to make marketing choices. Consider, for instance, a department store transaction database. Current algorithms will find it to be very useful to purchase a 50-inch plasma TV and pen since both products have produced high profits globally when sold together [28].

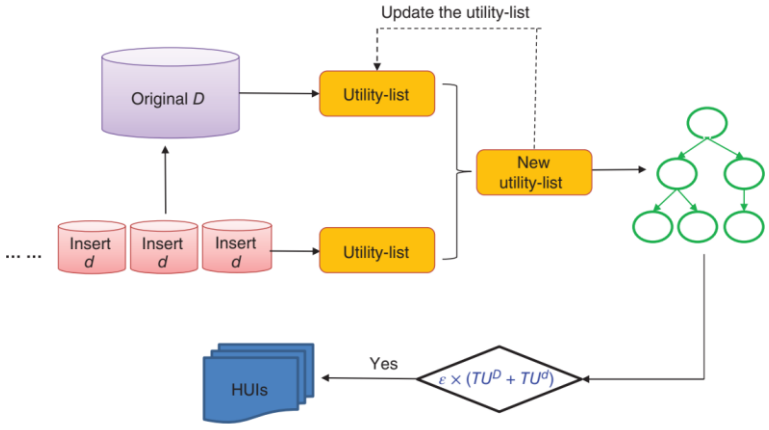

Figure 7. Flow model of a HUIM approach

But using this trend to advertise plasma TV for those who purchase plumbing would be an error, as these two products are seldom sold together if you take a close look. Although a very small connection exists between pens and plasma televisions, this pattern can be a highly useful element, and therefore almost all things paired with a plasma television can be a HUI. This is because plasma television is highly costly This is critical for restricting standard high utility mining algorithms. In

an experimental sample, less than 1 percent of patterns observed in conventional mining algorithms for high utility components also have elements that are highly correlated [27].

6. Research Opportunities and Challenges

This section discusses different possibilities of study with high average utility models. HAUIM research is open to designing better-performing algorithms than previous algorithms. In the areas of research that are using relatively high utility pattern mining, analysis is also available. Better algorithms build a great deal was done to build algorithms for my "high average utility products." Some algorithms take a lot of time, and some algorithms use a lot of memory. The tighter upper limits can be established to allow the pruning of unpromising papers. Better techniques of pruning can be created to reduce the search space.

Research in the field of time consumption, necessary memory, the number of applicants created, and scalability is open in HAUIM. Research-oriented towards application; the reliability of the algorithms is a major part of research carried out in the field of HAUIM. The implementation of HAUIM is no job completed. You can create applications to extract High Utility patterns from various social networks such as Facebook, Reddit, Twitter, etc. Research-oriented towards application Most algorithms in pattern mining use information that does not shift over time. Various algorithms that can handle complex data like spatiotemporal data, text, and stream data are created [7], [9], [12], [13]. Static data is the most often used data. Data is complex in real-life systems. For instance, stock information is often complex.

To mine "High Average Utility Itemsets" dynamic data, efficient algorithms can be created. Development of different data structures was used like "High Average Utility Item Sets" listing based on a tree. Different new data structures including graph data structures and updated list structures may be used to make the algorithms more powerful. Use of a distributive method Mining high-average utility items from transaction databases has been carried out in enormous amounts [21]. A parallel solution to mine HAUIs can be used to maximize the runtime of the HAUI without jeopardizing the mining of the HAUI. While for more than a decade the issue of high utility component mining was studied and numerous papers on the subject have been written, numerous research opportunities exist.

- The foundational opportunities for study are for the use of current pattern mining algorithms in new ways. Because algorithms for pattern mining are very common, they can be used in several fields. In particular, in developing fields like data visualization, the Internet of Things, and sensor networks the use of pattern-mining techniques offers several new applications.
- Pattern mining can take quite a while to develop more effective algorithms, particularly on thick datasets, massive databases, and databases containing numerous longer transactions. This is particularly important with new extensions to the issue of high-value mining, such as on-stage high-value mining or intermittent, less studied useful item-set mining. Many possibilities often lie in the distributed creation of algorithms, GPU, multi-core, or parallel to improve algorithms' speed and scalability.
- Another potential area for study is the development of high utility model mining algorithms for complex data forms. Various extensions have been suggested, as discussed in this article. However, the management of more complex data forms such as spatial data remains a challenge.
- Another relevant question relating to this research area is to find more complicated pattern forms. Furthermore, another chance for the study is to assess trends with new measures, for example, since it is often important to ensure the most fascinating or helpful patterns are detected.

7. Conclusions

Conventional HUIM architectures are built-in static databases to discover HUIs. However, the highly advantageous HUP effectively reduces the full runtime and is suitable for continuous information handling, as however, in many functional fields more critical HUPs are necessary for fast exploration. A detailed study with a measured average utility was discussed in this paper on all high-value itemset mining algorithms. As these algorithms usually face substantial performance problems as complex data is processed, incremental data mining has become a significant subject of study in recent decades. However, no research offered a systematic survey and description of the UIM algorithms and suggested that these algorithms should be taxonomically general. We explored and classified fundamental approaches to HUIM in this article. Highly useful itemset mining is a multifaceted area of science. The key techniques used to explore the search area of articles employed by useful element set mining algorithms were described in this chapter. The database is much lower in storage and scanning time than the initial transaction database capacity. It simplifies the operating procedure to a great degree. The conceptual confirmation and test results show that our approach is preferable to the standard Apriori Algorithm both in time and space while facing three thick datasets. However, the method of choosing the optimum cluster number is sluggish when the data volume is sparse. The paper then addressed extensions of the standard high usefulness mining algorithm to solve some of its shortcomings, for instance, handling complex databases and using different restrictions. Finally, the paper addressed various possibilities for future study.

Authors' Contributions

Aditya and Srinivasan collected the literature from various sources. The analysis part was shared by both the authors. Research directions were derived by Srinivasan. Both authors read and approved the final manuscript.

Competing Interests

The authors declare that they have no competing interests.

References

[1]. Fournier-Viger, P., Chun-Wei Lin, J., Truong-Chi, T., and Nkambou, R., "A survey of high utility itemset mining", In High-utility pattern mining, 2019: 1-45. Springer, Cham.

[2]. Duong, H., Truong, T., Tran, A. and Le, B., "Fast generation of sequential patterns with item constraints from concise representations", Knowledge and Information Systems, 2020, 62(6): 2191-2223.

[3]. Pazhaniraja, N. and Sountharrajan, S., "High utility itemset mining using dolphin echolocation optimization", Journal of Ambient Intelligence and Humanized Computing, 2021, 12(8): 8413-8426.

[4]. Masseglia, F., Poncelet, P. and Teisseire, M., "Efficient mining of sequential patterns with time constraints: Reducing the combinations", Expert Systems with Applications, 2009, 36(2): 2677-2690.

[5]. Nouioua, M., Fournier-Viger, P., Wu, C.W., Lin, J.C.W. and Gan, W., "FHUQI-Miner: Fast high utility quantitative itemset mining", Applied Intelligence, 2021, 51(10): 6785-6809.

[6]. Dawar, S., Goyal, V. and Bera, D., 2021. "Mining high-utility itemsets from a transaction database" (Doctoral dissertation, IIIT-Delhi).

[7]. Pham, T.T., Do, T., Nguyen, A., Vo, B. and Hong, T.P., "An efficient method for mining top-K closed sequential patterns", IEEE Access, 2020, 8: 118156-118163.

[8]. Nguyen, L.T., Nguyen, P., Nguyen, T.D., Vo, B., Fournier-Viger, P. and Tseng, V.S., "Mining high-utility itemsets in dynamic profit databases", Knowledge-Based Systems, 2019, 175: 130-144.
[9]. Gan, W., Lin, J.C.W., Fournier-Viger, P., Chao, H.C., Hong, T.P. and Fujita, H., "A survey of incremental high-utility itemset mining", Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 2018, 8(2): 1242.
[10]. Sun, R., Han, M., Zhang, C., Shen, M. and Du, S., "Mining of top-k high utility itemsets with negative utility", Journal of Intelligent & Fuzzy Systems, 2021, 40(3): 5637-5652.
[11]. Logeswaran, K., Andal, R.K.S., Ezhilmathi, S.T., Khan, A.H., Suresh, P. and Kumar, K.P., "A survey on metaheuristic nature inspired computations used for mining of association rule, frequent itemset and high utility itemset", In IOP Conference Series: Materials Science and Engineering, 2021, 1055(1): 012103. IOP Publishing.
[12]. Niu, K., Jiao, H., Gao, Z., Chen, C. and Zhang, H., "A developed apriori algorithm based on frequent matrix", In Proceedings of the 5th international conference on bioinformatics and computational biology, 2017: 55-58.
[13]. Dinh, D.T., Le, B., Fournier-Viger, P. and Huynh, V.N., "An efficient algorithm for mining periodic high-utility sequential patterns", Applied Intelligence, 2018, 48(12): 4694-4714.
[14]. Leleu, M., Rigotti, C., Boulicaut, J.F. and Euvrard, G., "Constraint-based mining of sequential patterns over datasets with consecutive repetitions", In European Conference on Principles of Data Mining and Knowledge Discovery, 2003: 303-314. Springer, Berlin, Heidelberg.
[15]. Liao, J., Wu, S. and Liu, A., "High utility itemsets mining based on divide-and-conquer strategy", Wireless Personal Communications, 2021, 116(3): 1639-1657.
[16]. Dong, X., Qiu, P., Lü, J., Cao, L. and Xu, T., "Mining Top-k Useful Negative Sequential Patterns via Learning", IEEE Transactions on Neural Networks and Learning Systems, 2019, 30(9): 2764-2778.
[17]. Truong, T., Duong, H., Le, B., Fournier-Viger, P., Yun, U. and Fujita, H., "Efficient algorithms for mining frequent high utility sequences with constraints", Information Sciences, 2021, 568: 239-264.
[18]. Srilatha, G. and Chandra, N.S., "Robust frequency affinity-based high utility itemset mining approach using multiple minimum utility", 2021, Materials Today: Proceedings.
[19]. Dam, T.L., Li, K., Fournier-Viger, P. and Duong, Q.H., "An efficient algorithm for mining top-k on-shelf high utility itemsets", Knowledge and Information Systems, 2017, 52(3): 621-655.
[20]. Nawaz, M.S., Fournier-Viger, P., Yun, U., Wu, Y. and Song, W., "Mining high utility itemsets with Hill climbing and simulated annealing", ACM Transactions on Management Information System (TMIS), 2021, 13(1): 1-22.
[21]. Vivekanandan, S.J., Ammu, S.P., Sripriyadharshini, R. and Preetha, T.R., "Computation of high utility item sets by using range of utility technique", J Univ Shanghai Sci Technol, 2021, 23(4): 94-101.
[22]. Kenny Kumar, M.J. and Rana, D., "High Average Utility Itemset Mining: A Survey", In Proceedings of International Conference on Computational Intelligence and Data Engineering, 2021: 347-374. Springer, Singapore.
[23]. Nawaz, M.S., Fournier-Viger, P., Song, W., Lin, J.C.W. and Noack, B., "Investigating crossover operators in genetic algorithms for high-utility itemset mining", In Asian Conference on Intelligent Information and Database Systems, 2021: 16-28. Springer, Cham.
[24]. Singh, K., Singh, S.S., Kumar, A. and Biswas, B., "TKEH: an efficient algorithm for mining top-k high utility itemsets", Applied Intelligence, 2019, 49(3): 1078-1097.
[25]. Chu, C.J., Tseng, V.S. and Liang, T., "An efficient algorithm for mining high utility itemsets with negative item values in large databases", Applied Mathematics and Computation, 2009, 215(2): 767-778.

[26]. Fournier-Viger, P., Wu, Y., Dinh, D.T., Song, W. and Lin, J.C.W., "Discovering periodic high utility itemsets in a discrete sequence", In Periodic Pattern Mining, 2021: 133-151. Springer, Singapore.
[27]. Han, X., Liu, X., Li, J. and Gao, H., "Efficient top-k high utility itemset mining on massive data", Information Sciences, 2021, 557: 382-406.
[28]. Duan, Y., Fu, X., Luo, B., Wang, Z., Shi, J. and Du, X., "Detective: Automatically identify and analyze malware processes in forensic scenarios via DLLs", In 2015 IEEE International Conference on Communications (ICC), 2015: 5691-5696. IEEE.
[29]. Lin, J.C.W., Ren, S., Fournier-Viger, P. and Hong, T.P., "EHAUPM: Efficient high average-utility pattern mining with tighter upper bounds", IEEE Access, 2017, 5: 12927-12940.
[30]. Yun, U., Nam, H., Lee, G. and Yoon, E., "Efficient approach for incremental high utility pattern mining with indexed list structure", Future Generation Computer Systems, 2019, 95: 221-239.