Iztok Fister Jr. and Iztok Fister

Abstract Decisions made nowadays by Artificial Intelligence powered systems are usually hard for users to understand. One of the more important issues faced by developers is exposed as how to create more explainable Machine Learning models. In line with this, more explainable techniques need to be developed, where visual explanation also plays a more important role. This technique could also be applied successfully for explaining the results of Association Rule Mining.This Chapter focuses on two issues: (1) How to discover the relevant association rules, and (2) How to express relations between more attributes visually. For the solution of the first issue, the proposed method uses Differential Evolution, while Sankey diagrams are adopted to solve the second one. This method was applied to a transaction database containing data generated by an amateur cyclist in past seasons, using a mobile device worn during the realization of training sessions that is divided into four time periods. The results of visualization showed that a trend in improving performance of an athlete can be indicated by changing the attributes appearing in the selected association rules in different time periods.

Citation details: Fister Jr. I., Fister I. Association rules over time. In: Frontier advances of nature inspired industrial optimization, Springer Tracts in Nature-Inspired Computing (STNIC), 2021.

Iztok Fister Jr.

University of Maribor, Faculty of Electrical Engineering and Computer Science, Koroška cesta 46, Slovenia, e-mail: iztok.fister1@um.si

Iztok Fister

University of Maribor, Faculty of Electrical Engineering and Computer Science, Koroška cesta 46, Slovenia, e-mail: iztok.fister@um.si

1 Introduction

Nowadays, Artificial Intelligence (AI) powered systems are sophisticated to such an extent that they actually do not need human intervention for their design and deployment. On the other hand, their decisions ultimately start to affect human lives concerning medicine, law, economy, and defense [3]. As a result, these sophisticated decisions demand some understanding, on which basis these are furnished by AI methods [10].

The problem is not crucial considering older AI-powered systems, because these are easy to interpret. Actually, they act as black-boxes, and, as such, are easily understandable by the users. Obviously, the user wants to understand the mechanisms, by which the Machine Learning (ML) models work [22]. When the AI-powered system is transparent, this means that decisions made by such ML models are justifiable, legitimate, and allow obtaining detailed explanations of their behavior [11].

Explainable AI (EAI) [11] has emerged in order to avoid the limitations of the current AI systems. Primarily, two issues were placed before the new domain as follows [3]: (1) To create more explainable ML models by maintaining a high level of learning performance, and (2) To enable humans to understand, trust, and manage the emerging platform of AI partners effectively. In the sense of the first issue, more post-hoc explainable techniques have been developed, where visualization explanation plays an important role. Post-hoc means that the explainable techniques are applied after obtaining the results of specific ML methods, in order to discover any additional knowledge hidden in data, and, thus, help the user to understand the results properly.

Association Rule Mining (ARM) is a well known ML method that normally generates a huge amount of association rules, from which it is hard to make the proper decisions easily. An additional problem is presented by the complex form of the results, which are represented as an implication $X \Rightarrow Y$, meaning "if antecedent X is *true* then its consequent Y is also *true*". As a result, there we are confronted with two problems: (1) How to discover the relevant association rules in a huge archive, and (2) How to express the relations between attributes in those implication rules visually in a form that is understandable and interpretable by users.

Interestingly, the visualization of association rules has rarely been treated in literature. Indeed, the papers which referred to these topics can be summarized in the following review: The authors in [27] presented a design that is able to handle hundreds of multiple antecedent association rules in a three-dimensional display with minimum human interaction, low occlusion percentage, and no screen swapping. The authors in [16] show that the use of Mosaic plots and their variant, called Double Decker plots, can be applied for visualizing association rules. Ong et al. [23] prototyped the two visualizations, called grid view and tree view, for visualizing the association rules in their application called CrystalClear. Appice and Buono [2] presented a graph-based visualization that supports data miners in the analysis of multi-level spatial association rules, while Herawan et al. [15] proposed an approach for visualizing soft maximal association rules. A very interesting interactive visualization technique, which lets the user navigate through a hierarchy of association

rule groups is presented in paper [14]. The authors in paper [18] explored the Hasse diagrams for the visualization of Boolean association rules. Fister et al. [7] proposed a method for identifying dependencies among mined association rules based on population-based metaheuristics [8, 13, 12] and complex networks, while the paper of Fister Jr. et al. [19] looked for a visualization method capable of telling stories based on mined association rules. However, there are also generic tools, like the CloseViz [4] and the SPMF open-source data mining library Version 2 [9], specialized primarily in pattern mining, offering visual implementation of discovered data mined by ML algorithms that could also be used for visualization of ARM.

This Chapter proposes monitoring the performance of an athlete involved in sport training sessions during the past seasons (i.e., years). Thus, these seasons are divided into four time periods, where each of the periods captures a quarter of the data. The performances of the athlete are measured using real data obtained by mobile devices worn by the athlete during the training sessions. The performance data (e.g., duration, average heart rate, etc.) are stored as attributes into a transaction database, from which the knowledge hidden in these is discovered using the ARM methods. These methods normally produce a huge amount of data stored in the so-called ARM archives. Data stored in these archives are also distinguished according to specific time periods. These periods, indeed, present the flow of performance of the athlete, while transitions from one period to another highlight how the form of the particular athlete progressed.

In order to illustrate information from ARM archives in an understandable and interpretable way to users, Sankey diagrams are selected, that enable multivariate environment and historical data [21]. The method was applied to a transaction database created by an amateur athlete in the past seasons over the duration of four and a half years. The results of visualization allow a sport trainer to analyze how the performance of the athlete in sport training improved during the seasons, on the one hand, and which attributes in which of the time periods affected these the most, on the other.

The structure of the remainder of the chapter is as follows: Section 2 deals with the basic information needed for understanding the sections that follow. A description of a method for creating the Sankey diagrams from the ARM archive is the subject of Section 3. The experiments and the results are illustrated in Section 4. Chapter concludes with Section 5, that summarizes the performed and outlines the future work.

2 Basic information

This section is devoted to explaining the concepts necessary for a reader to understand the subjects that follow. In line with this, the following concepts are highlighted:

- ARM,
- DE,
- Sankey diagram,

formal definition of objectives.

In the remainder of the section, the mentioned subjects are discussed in detail.

2.1 Association Rule Mining

ARM can be defined formally as follows: Let us assume a set of objects $O = \{o_1, \ldots, o_M\}$ and transaction dataset $T_D = \{T\}$ are given, where each transaction T is a subset of objects $T \subseteq O$. Then, an association rule is defined as an implication:

$$X \Rightarrow Y,\tag{1}$$

where $X \subset O$, $Y \subset O$, and $X \cap Y = \emptyset$. In order to estimate the quality of the mined association rule, two measures are defined: support and confidence. The support is defined as:

$$supp(X \Rightarrow Y) = \frac{\#(X \cup Y)}{N},$$
 (2)

whereas confidence as:

$$conf(X \Rightarrow Y) = \frac{\#(X \cup Y)}{\#(X)},$$
(3)

where the function #(.) calculates the number of repetitions of a particular rule within T_D , and N is the total number of transactions in T_D . Let us emphasize that two additional variables are defined, i.e. minimum support S_{min} and minimum confidence C_{min} . These variables denote a threshold value limiting the particular association rule with lower confidence and support from being taken into consideration.

2.2 Differential Evolution

DE is appropriate for solving continuous, as well as, discrete optimization problems. Many DE variants have been proposed since the algorithm's birth in 1995. The original DE algorithm manages a population of real-valued vectors in the form:

$$\mathbf{x}^{(t)} = (x_{i,1}^{(t)}, \dots, x_{i,d}^{(t)}), \tag{4}$$

where d denotes the dimension of the problem and t is a generation counter. These vectors undergo operations of operators, such as mutation, crossover, and selection. In the basic mutation, two solutions are selected randomly and their scaled difference is added to the third solution, as follows:

$$\mathbf{u}_{i}^{(t)} = \mathbf{x}_{r0}^{(t)} + F \cdot (\mathbf{x}_{r1}^{(t)} - \mathbf{x}_{r2}^{(t)}), \quad \text{for } i = 1, \dots, NP,$$
(5)

where $F \in [0.1, 1.0]$ denotes the scaling factor that scales the rate of modification, *NP* represents the population size and *r*0, *r*1, *r*2 are randomly selected values in the interval 1,...,*NP*. Note that the proposed interval of values for parameter *F* was enforced in the DE community.

DE employs a binomial (denoted as 'bin') or exponential (denoted as 'exp') crossover. The trial vector is built from parameter values copied from either the mutant vector generated by Eq. (5) or parent at the same index position laid *i*-th vector. Mathematically, this crossover can be expressed as follows:

$$w_{i,j}^{(t)} = \begin{cases} u_{i,j}^{(t)}, & \text{rand}_{j}(0,1) \le CR \lor j = j_{rand}, \\ x_{i,j}^{(t)}, & \text{otherwise,} \end{cases}$$
(6)

where $CR \in [0.0, 1.0]$ controls the fraction of parameters that are copied to the trial solution. The condition $j = j_{rand}$ ensures that the trial vector differs from the original solution $\mathbf{x}_{i}^{(t)}$ in at least one element.

Mutation and crossover can be performed in several ways in DE. Consequently, in general, a specific notation was introduced to describe the varieties of these methods (also mutation strategies). For example, 'rand/1/bin' denotes that the base vector is selected randomly, 1 vector difference is added to it, and the number of modified parameters in the trial/offspring vector follows a binomial distribution.

Mathematically, the selection can be expressed as follows:

$$\mathbf{x}_{i}^{(t+1)} = \begin{cases} \mathbf{w}_{i}^{(t)}, & \text{if } f(\mathbf{w}_{i}^{(t)}) \leq f(\mathbf{x}_{i}^{(t)}), \\ \mathbf{x}_{i}^{(t)}, & \text{otherwise.} \end{cases}$$
(7)

The selection is usually called 'one-to-one', because the trial and the corresponding vector laid on the *i*-th position in the population compete for surviving in the next generation, where the better will survive according to the fitness function.

2.3 Sankey diagrams

Typically, Sankey diagrams are used for illustrating the quality and connectivity of flows between entities across time. Indeed, these diagrams present directed weighted graphs, where entities are represented as nodes connected by edges of different widths. The weights of the edges are referred to the quality of the flow.

The pioneer work of this kind of visualization was made by Charles Minard, who presented Napoleon's Russian Campaign of 1812 using a flow diagram, denoting the amount of French soldiers, overlaid onto a geographical map (Fig. 1). These kind of diagrams were named after Captain Matthew Sankey, who used this method for visualizing energy transfer in steam engines [24].

The main problem in creating Sankey diagrams is the placement of nodes within layers such that crossings are avoided between edges (flows). Compared to the classical layered graph drawing problems, the Sankey creation is harder, due to the fact



Fig. 1: Minard's classic diagram of Napoleon's invasion of Russia, a predecessor of the Sankey diagram from [26].

that each edge has a width connected to the value it encodes, and, therefore, crossings are of different importance. In general, this problem is defined as combinatorial and solved using various deterministic optimization algorithms, like the heuristic method proposed by Sugiyama et al. [25], Integer Programming by Zarate et al. [28], and even a visual approach to understand and analyze the flow of information in a ML system by Chaudhuri et al. [5].

2.4 Formal definition of the objectives

The aim of visualization using Sankey diagrams for ARM is to search for those association rules that are distinguished by their similarities (quality aspect), and expose the good fitness (quantity aspect) regarding more time periods (temporal aspect). The spatial aspect of the map is reflected in the topology of nodes (attributes) connected with edges (attribute flows). Indeed, there are two issues that need to be considered for construction of the Sankey diagrams for ARM:

- the map size,
- the similarity of the visualized association rules.

In summary, the Sankey diagram for ARM can be defined as a pair $\mathcal{F} = \langle M, R \rangle$, where *M* limits the map size as:

$$|R| \le M,\tag{8}$$

and the *R* denotes the set of the most similar rules, determined according to the following method: Let us suppose that the results of the algorithm for ARM is a huge archive of association rules in the form of $X \Rightarrow Y$, which can, in general, be

expressed as:

$$X_1 \wedge X_2 \wedge \ldots \wedge X_n \Longrightarrow Y_1 \wedge Y_2 \wedge \ldots \wedge Y_m, \tag{9}$$

where X_i for i = 1, ..., n denotes the attributes of antecedent, and Y_j for j = 1, ..., m the attributes of consequent sets, respectively.

In order to divide both parts of the particular association rule, the following functions are introduced:

$$Ante(X \Rightarrow Y) = \{X_1, X_2, \dots, X_n\},\$$

$$Cons(X \Rightarrow Y) = \{Y_1, Y_2, \dots, Y_m\},$$
(10)

where the $Ante(X \Rightarrow Y)$ function denotes the antecedent and the $Cons(X \Rightarrow Y)$ the consequent set of attributes contributed in the particular association rule $X \Rightarrow Y$.

Then, the similarity between two rules R_1 and R_2 can be calculated as follows:

$$Sim(R_1, R_2) = \frac{|Ante(R_1) \cap Ante(R_2)| + |Cons(R_1) \cap Cons(R_2)|}{|Ante(R_1) \cup Ante(R_2)| + |Cons(R_1) \cup Cons(R_2)|}.$$
 (11)

Let us mention that the value of $Sim(R_1, R_2) \in [0, 1]$, where 1 means the full similarity, and 0 the full dissimilarity. Interestingly, the similarity values of the rules within the ARM archive can be combined into an adjacency matrix Adj, as follows:

$$\operatorname{Adj} = \begin{bmatrix} a_{1,1} & \dots & a_{1,\tilde{N}} \\ \dots & \dots & \\ a_{\tilde{N},1} & \dots & a_{\tilde{N},\tilde{N}} \end{bmatrix},$$
(12)

where $a_{i,j} = sim(R_i, R_j)$ for $i = 1, ..., \tilde{N}$ and $j = 1, ..., \tilde{N}$ presents a similarity value between rules R_i and R_j , and $\tilde{N} \le N$ denotes the number of the observed best association rules in the archive.

Based on the introduced formalism, the problem of searching for the most similar set of association rules R can be defined as a Knapsack 0/1 [20] problem in the following sense: Let us suppose \tilde{N} objects and a knapsack of capacity M are given. The problem is how to put into a knapsack the maximum number of objects of different weights $W = (w_1, \ldots, w_{\tilde{N}})$ such that the obtained profit $Adj = \{p_{i,j}\}$ is maximum. If a binary vector $B = (b_1, \ldots, b_{\tilde{N}})$ is adopted for denoting the inclusion of the object into the knapsack, the problem can be defined formally as follows:

$$\max \sum_{i=1}^{\tilde{N}-1} a_{i,j} \cdot b_i, \quad \text{for } j = i+1, \dots, \tilde{N}, \quad (13)$$

subject to:

$$\max\sum_{i=1}^{\tilde{N}} w_i \cdot b_i, \le M.$$
(14)

Indeed, Eq. (13) determines a lot of association rules with equal maximum values of similarity.

3 Method for creating the Sankey diagrams from the ARM archive

A method for visualizing an archive of mined association rules using Sankey diagrams is divided into three steps (Fig. 2):

- preprocessing,
- optimization,
- visualization.

In the remainder of the section, the mentioned steps are illustrated in detail.



Fig. 2: Architecture of the method for visualization of ARM using Sankey diagrams.

3.1 Preprocessing

In the preprocessing step, a uniform transaction database is generally divided into K auxiliary databases, where each of these captures transactions of the K-th time period. Normally, the performance improves from period to period. The motivation behind this selection in sport, for instance, lies in the fact that the best condition of the athlete must be turned on in the last time period, where the most important competitions are expected. The fact demands that the athlete needs to be prepared optimally at this time. On the other hand, the first time periods are dedicated to rest or performing sport training sessions of less intensity.

However, the same method can also be applied for the other domains that are confronted with the association rules changing over time.

3.2 Optimization

Mining the association rules is performed in the proposed method using the DE algorithm for ARM, which demands modifying the following algorithm's components [6]:

- · representation of individuals,
- genotype-phenotype mapping,
- fitness function evaluation.

Each solution in the proposed DE represents a mined association rule that is encoded of d-discrete attributes, to which the value for determining the ordering of the particular feature in the definite association rule is assigned. In addition, the last element of the vector denotes the so-called cut point, separating the antecedent from the consequent part of the association rule. As a result, the individual is represented as follows:

$$\mathbf{x}_{i}^{(t)} = (\underbrace{\langle x_{i,0}^{(t)}, x_{i,1}^{(t)} \rangle}_{Feat_{1}}, \dots, \underbrace{\langle x_{i,2(d-1)}^{(t)}, x_{i,2d-1}^{(t)} \rangle}_{Feat_{d}}, \underbrace{x_{i,2d}^{(t)}}_{Cp}),$$
(15)

where each pair $Feat_j = \langle x_{i,2j}^{(t)}, x_{i,2j+1}^{(t)} \rangle$ for j = 0, ..., d-1 denotes the corresponding *j*-th feature with the first element designated decoded attribute, the second determining the ordering of this in a permutation $\Pi = (\pi_1, ..., \pi_d)$, and the *Cp* is a cut point. Obviously, the variable *d* indicates the maximum number of features, and the length of the individual is 2d + 1.

Genotype-phenotype mapping is used to map a representation of individuals in a genotype space to the variables in the problem space [6]. In our case, the first variables in a pair $x_{i,2j}^{(t)}$, for j = 0, ..., d-1, are decoded to the categorical attributes *Attr_j* according to the equation:

$$Attr_{\pi_j} = \left[\frac{x_{2j}^{(t)}}{|Feat_{\pi_j}| + 1} \right], \quad \text{for } j = 1, \dots, d.$$
(16)

Let us mention that $Attr_{\pi_j} = 0$ has a special meaning, because it determines that the corresponding feature is not presented in the association rule.

A permutation of attributes in the association rule is calculated as follows: Each value $x_{2j+1}^{(t)}$ for j = 0, ..., d - 1 are assembled into an auxiliary vector $X' = (x_2^{(t)}, ..., x_{2d-1}^{(t)})$. After ordering the descending vector, the permutation $\Pi = (\pi_1, ..., \pi_d)$ is constructed, such that the following relation is valid:

$$x_{i,\pi_1}^{(t)} \le x_{i,\pi_3}^{(t)} \le \ldots \le x_{i,\pi_{2d-1}}^{(t)}.$$
(17)

The cut point is calculated according to the following equation:

$$Cp_i = \lfloor x_{2d}^{(t)} \cdot (d-2) \rfloor + 1, \quad \text{for } i = 1, \dots, Np,$$
 (18)

and delimits the antecedent from the consequent part of the particular association rule.

Fitness function is expressed as follows:

$$f(\mathbf{x}_{i}^{(t)}) = \frac{\alpha \cdot conf(\mathbf{x}_{i}^{(t)}) + \beta \cdot supp(\mathbf{x}_{i}^{(t)})}{\alpha + \beta},$$
(19)

where α and β denote weights, while $conf(\mathbf{x}_i^{(t)})$ and $supp(\mathbf{x}_i^{(t)})$ are the confidence and the support of the mined association rule, respectively.

3.3 Visualization

As can be seen from Eq. (9), mined association rules in the form $X \Rightarrow Y$ express explicitly the flow of a conjunction of antecedent attributes to a conjunction of consequent rules. Therefore, it is natural to represent them in a weighted graph, where the antecedent attributes denote the source and the consequent ones the sink nodes in the graph. Thus, the source nodes are joined with edges into a chain flowing to the specific sink nodes. The quality aspect of the graph is reflected by representing in the graph only the best *n*-association rules according to the similarity measure.

When a weight is assigned to each edge, connecting source and drain nodes, that assesses the strength of the selected association rule in the sense of its fitness value, and the best mined association rules (i.e., flows) are compared in different time periods, also the temporary aspect of realizing the sport training sessions is obtained during the season. Thus, all preconditions for creating the Sankey diagram are satisfied.

Finally, a dynamic Knapsack 0/1 algorithm [17] is applied for searching the M the most similar association rules appropriate for creating the Sankey diagrams. In order to prefer the more important ones, these rules are additionally distinguished according to their fitness function values.

4 Experiments and results

The goal of the experimental work was to show that the proposed method for association rules over time can be explained successfully by visualizing with Sankey diagrams. In line with this, improving the performance by an amateur cyclist during past seasons was analysed based on data obtained by realizing sport sessions using a wearable device.

10

The proposed method employs two algorithms: (1) The DE for ARM, and (2) The Dynamic KNAPsack 0/1 (DKNAP 0/1). The former is responsible for mining association rules, while the latter for explaining the content of huge ARM archives to the user by constructing the Sankey diagrams. The parameters of the mentioned algorithms used during the experimental work are illustrated in Table 1.

Algorithm	Parameter	Abbreviation	Value
DE	Scale factor	F	0.5
	Crossover rate	CR	0.9
	Population size	Np	100
DKNAP 0/1	Map size	M	4
	Number of observed rules	N	100

Table 1: Parameter settings of the algorithms in tests.

Let us mention that the best results of the DE algorithm, obtained after 25 independent runs, were considered for visualization. Obviously, these results were assessed according to the fitness function value expressed by Eq. (19). On the other hand, the result of the DKNAP 0/1 was obtained after one run.

4.1 Transaction database

The mobile data obtained by wearable devices were saved into a transaction database that was divided into K = 4 auxiliary databases according to the different time periods. In our case, the first time period captures transactions realized from March 2013 to May 2014, the second time period from May 2014 to July 2015, the third time period from July 2015 to May 2016, and the last time period from October 2016 to October 2017 (Table 2). Interestingly, the sizes of the auxiliary databases can also be seen from the Table.

Part	Time period	Number of transactions
1	March 2013 to May 2014	126
2	May 2014 to July 2015	134
3	July 2015 to May 2016	139
4	October 2016 to October 2017	136
Total	The whole dataset	535

Table 2: Characteristics of auxiliary databases.

As can be seen from Table 3, each transaction is described by seven discrete features indicating the performance of the athlete during realization of a particular training session. To each feature in the table, the corresponding domain of values is attached that is represented as a discrete set of feasible attributes.

Nr.	Feature	Abbreviation	Attributes
1	Duration	DURATION	{SHORT,MEDIUM,LONG}
2	Distance	DISTANCE	{SHORT,MEDIUM,LONG}
4	Calories	CALORIES	{SMALL,MEDIUM,HIGH}
3	Average Heart Rate	HR	{MEDIUM,HIGH}
5	Altitude	ALTITUDE	{LOW,MEDIUM,HIGH}
6	Ascent	ASCENT	{LOW,MEDIUM,HIGH}
7	Descent	DESCENT	{LOW,MEDIUM,HIGH}

Table 3: Observed discrete features with their corresponding attributes.

Let us mention that attributes in the mined association rules are represented as a concatenation of a feature name by an attribute name using character '_' for joining both. Obviously, the DE for ARM is applied on each of the auxiliary databases independently. As a result, four independent archives of mined association rules are obtained that further enters into the visualization process.

4.2 The results

The results of the proposed method for ARM changing over time need to be analyzed after two steps: The first step highlights the results of an optimization, while the second a visualization of the obtained data using Sankey diagrams. The results of the optimization are illustrated in Table 4 that presents the four best association rules according to the fitness function value. As can be seen from the Table, relatively sim-

Table 4: The best association rules selected for visualization.

Part	Rule nr.	Association rule	Fitness
1	1	CALORIES_SMALL \Rightarrow ASCENT_LOW	0.7262
	2	DURATION_SHORT \Rightarrow ASCENT_LOW	0.6661
	3	CALORIES_SMALL \land DURATION_SHORT \Rightarrow ASCENT_LOW	0.6439
	4	CALORIES_SMALL \land DISTANCE_SHORT \Rightarrow ASCENT_LOW	0.6293
2	1	$CALORIES_SMALL \Rightarrow ASCENT_LOW$	0.7612
	2	DURATION_SHORT \Rightarrow ASCENT_LOW	0.7015
	3	CALORIES_SMALL \land DURATION_SHORT \Rightarrow ASCENT_LOW	0.6977
	4	$DISTANCE_SHORT \Rightarrow ASCENT_LOW$	0.6903
3	1	$CALORIES_SMALL \Rightarrow ASCENT_LOW$	0.7446
	2	DISTANCE_SHORT \Rightarrow ASCENT_LOW	0.7122
	3	DURATION_SHORT \Rightarrow ASCENT_LOW	0.7086
	4	CALORIES_SMALL \land DURATION_SHORT \Rightarrow ASCENT_LOW	0.7014
4	1	DURATION_SHORT \Rightarrow ASCENT_LOW	0.7500
	2	$DISTANCE_SHORT \Rightarrow ASCENT_LOW$	0.6875
	3	$HR_HIGH \Rightarrow ASCENT_LOW$	0.6744
	4	DURATION_SHORT \land DISTANCE_SHORT \Rightarrow ASCENT_LOW	0.6691

ple association rules are preferred by the DE for ARM, i.e., one consequent follows from one antecedent. On the other hand, all consequents in the Table converge to the same attribute, ASCENT_LOW. As antecedents in compound association rules, typically, a permutation of two attributes are taken from a set {CALORIES_SMALL, DURATION_SHORT,DISTANCE_SHORT}.

Visualization of the results presented in Table 4 are depicted in Fig. 3 that is divided into four diagrams reflecting the performance of an athlete in four different time periods. Diagrams were visualized using the network3D package [1]. As



Fig. 3: Visualization of the most similar four association rules using Sankey diagrams.

can be seen from the Figure, the CALORIES_SMALL attribute is the most important antecedent in the first time period, due to producing the widest flow in the corresponding Sankey diagrams. Indeed, this flow diminished till the end of the observed data, where it disappeared. In place of this attribute, an antecedent DU-RATION_SHORT and DISTANCE_SHORT prevailed. Interestingly, the antecedent HR_HIGH emerged in the last time period, denoting that the feature HR is a more important factor for determining the performance of an athlete than the CALORIES.

4.3 Discussion

Visualization of association rules changing over time with Sankey diagrams showed many advantages that can be summarized as follows: These diagrams are appropriate for representing the flow (topology) of a set of antecedents to different consequents. Actually, the antecedents are a sequence of source nodes, while the consequents represent sink nodes in the Sankey diagrams.

A temporary aspect of the presentation is accomplished by maintaining four auxiliary databases, within which searching for performances of the athlete is conducted temporally, and the results are depicted in Sankey diagrams. From the sequence of these diagrams, a sport trainer can conclude, which relations of attributes are more important for an athlete in a specific time period.

5 Conclusion

Sophisticated AI powered systems operate nowadays without any human intervention for design and deployment on the one hand, but they have become hard to understand for the users on the other. Therefore, the new domain of EAI has emerged that tries to avoid these limitations of the current AI systems. The EAI searches for answers to the following two issues: (1) How to create more explainable ML models, and (2) How to enable users to understand, trust, and manage the emerging AI platforms effectively. More post-hoc techniques have emerged for dealing with the first issue, where a visual explanation is one of the more prominent ones.

The aim of the chapter is to propose the new explainable method for visualization mined association rules over time. The method consists of three steps: preprocessing, optimization, and visualization. The first step is devoted to dividing the uniform transaction database into more auxiliary databases corresponding to definite time periods. In the second step, the archives of association rules mined by DE are discovered from each auxiliary database. These are then visualized using Sankey diagrams, which are capable of representing multivariate data from different aspects, i.e., quality, quantity, topological, and temporal. In these diagrams, each attribute in association rules denotes a node in the corresponding direct graph, while edges indicate the relations between these. In addition, the edges are of different widths that are proportional to the fitness of the association rule.

The proposed method was applied to the transaction database created by an amateur cyclist during the past seasons. Each training session was monitored by a mobile device worn by an athlete during the realization, and saved as a transaction into a transaction database. The transaction is characterized by discrete attributes. The transaction database was divided into four auxiliary databases each denoting one fourth of the past seasons. DE for ARM was employed to each of the auxiliary databases, producing four archives of the association rules. Four of the best association rules according to similarity and fitness are entered into the visualization using Sankey diagrams. The result of visualization showed that a trend in improving performance of an athlete can be indicated by changing the attributes appearing in the selected association rules in different time periods.

The preliminary study offers many directions for the future development: At first, the method could be applied to other transaction databases (e.g., UCI Machine Learning Repository). Then, DE could be replaced by another stochastic nature-inspired population based algorithms. Finally, another kind of calculating the similarity between association rules could be developed.

References

- J. Allaire, C. Gandrud, K. Russell, and C. Yetman. networkd3: D3 javascript network graphs from r. *R package version 0.4*, 2017.
- A. Appice and P. Buono. Analyzing multi-level spatial association rules through a graph-based visualization. In *International Conference on Industrial, Engineering and Other Applications* of *Applied Intelligent Systems*, pages 448–458. Springer, 2005.
- A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera. Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information Fusion*, 58:82–115, 2020.
- C. L. Carmichael and C. K.-S. Leung. Closeviz: Visualizing useful patterns. In *Proceedings* of the ACM SIGKDD Workshop on Useful Patterns, UP '10, page 17–26, New York, NY, USA, 2010. Association for Computing Machinery.
- A. Chaudhuri. A visual technique to analyze flow of information in a machine learning system. In T. Wischgoll, S. Zhang, D. L. Kao, and Y. Chiang, editors, *Visualization and Data Analysis* 2018, Burlingame, CA, USA, 28 January 2018 - 1 February 2018. Ingenta, 2018.
- A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computing*. Springer Publishing Company, Incorporated, 2nd edition, 2015.
- I. Fister Jr., A. Galvez, E. Osaba, J. D. Ser, A. Iglesias, and I. Fister. Discovering dependencies among mined association rules with population-based metaheuristics. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, pages 1668–1674, 2019.
- I. Fister Jr., X.-S. Yang, I. Fister, J. Brest, and D. Fister. A brief review of nature-inspired algorithms for optimization. *Elektrotehniški vestnik*, 80(3):116–122, 2013.
- P. Fournier-Viger, J. C.-W. Lin, A. Gomariz, T. Gueniche, A. Soltani, Z. Deng, and H. T. Lam. The spmf open-source data mining library version 2. In B. Berendt, B. Bringmann, É. Fromont, G. Garriga, P. Miettinen, N. Tatti, and V. Tresp, editors, *Machine Learning and Knowledge Discovery in Databases*, pages 36–40, Cham, 2016. Springer International Publishing.
- B. Goodman and S. Flaxman. Eu regulations on algorithmic decision-making and a "right to explanation". 2016. cite arXiv:1606.08813Comment: presented at 2016 ICML Workshop on Human Interpretability in Machine Learning (WHI 2016), New York, NY.
- 11. D. Gunning. Explainable artificial intelligence (xai). Defense Advanced Research Projects Agency (DARPA), nd Web, 2:2, 2017.
- N. Gupta, M. Khosravy, O. P. Mahela, and N. Patel. Plant biology-inspired genetic algorithm: superior efficiency to firefly optimizer. In *Applications of Firefly Algorithm and its Variants*, pages 193–219. Springer, 2020.
- 13. N. Gupta, M. Khosravy, N. Patel, and I. Sethi. Evolutionary optimization based on biological evolution in plants. *Procedia Computer Science*, 126:146–155, 2018.
- M. Hahsler and S. Chelluboina. Visualizing association rules in hierarchical groups. In 42nd Symposium on the Interface: Statistical, Machine Learning, and Visualization Algorithms (Interface 2011). The Interface Foundation of North America. Citeseer, 2011.
- T. Herawan, I. T. R. Yanto, and M. M. Deris. Smarviz: Soft maximal association rules visualization. In *International Visual Informatics Conference*, pages 664–674. Springer, 2009.

- H. Hofmann, A. P. Siebes, and A. F. Wilhelm. Visualizing association rules with interactive mosaic plots. In *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 227–235, 2000.
- E. Horowitz and S. Sahni. Fundamentals of Computer Algorithms. W. H. Freeman & Co., USA, 1978.
- B. Jiang, C. Han, and X. Hu. A finite ranked poset and its application in visualization of association rules. In 2008 IEEE International Conference on Granular Computing, pages 322–325. IEEE, 2008.
- 19. I. F. Jr. and I. Fister. Information cartography in association rule mining. 2020.
- 20. H. Kellerer, U. Pferschy, and D. Pisinger. Knapsack problems. Springer, 2004.
- 21. B. Lehrman. Visualizing water infrastructure with sankey maps: a case study of mapping the los angeles aqueduct, california. *Journal of Maps*, 14(1):52–64, 2018.
- 22. Z. C. Lipton. The mythos of model interpretability. Queue, 16(3):31-57, June 2018.
- H.-H. Ong, K.-L. Ong, W.-K. Ng, and E. P. Lim. Crystalclear: Active visualization of association rules. In *International Workshop on Active Mining AM 2002, in conjunction with the IEEE International Conference on Data Mining ICDM*, 2002.
- M. Sankey. Introductory note on the thermal efficiency of steam-engines. In *Minutes of* Proceedings of the Institution of Civil Engineers, volume 134, pages 278–283. ICE Publishing, 1898.
- K. Sugiyama, S. Tagawa, and M. Toda. Methods for visual understanding of hierarchical system structures. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(2):109–125, 1981.
- 26. Wikipedia, the free encyclopedia. Classic diagram of napoleon's invasion of russia by minard, 1869. [Online; accessed August 15, 2020].
- P. C. Wong, P. Whitney, and J. Thomas. Visualizing association rules for text mining. In Proceedings 1999 IEEE Symposium on Information Visualization (InfoVis' 99), pages 120– 123. IEEE, 1999.
- D. C. Zarate, P. L. Bodic, T. Dwyer, G. Gange, and P. Stuckey. Optimal sankey diagrams via integer programming. In 2018 IEEE Pacific Visualization Symposium (PacificVis), pages 135–139, 2018.